Article

Chicken Tracking and Individual Bird Activity Monitoring Using the BoT-SORT Algorithm

Allan Lincoln Rodrigues Siriani [1], Isabelly Beatriz de Carvalho Miranda [2], Saman Abdanan Mehdizadeh [3] and Danilo Florentino Pereira [4],*

1 Graduate Program in Agribusiness and Development, School of Sciences and Engineering, São Paulo State University, Tupã 17602-496, SP, Brazil; allan.siriani@unesp.br
2 Undergraduate Program in Big Data, Sao Paulo College of Technology, Pompeia 17586-050, SP, Brazil
3 Department of Mechanics of Biosystems Engineering, Faculty of Agricultural Engineering and Rural Development, Agricultural Sciences and Natural Resources University of Khuzestan, Ahvaz 63417-73637, Iran; s.abdanan@asnrukh.ac.ir
4 Department of Management, Development, and Technology, School of Sciences and Engineering, São Paulo State University, Tupã 17602-496, SP, Brazil
* Correspondence: danilo.florentino@unesp.br

Abstract: The analysis of chicken movement on the farm has several applications in evaluating the well-being and health of birds. Low locomotion may be associated with locomotor problems, and undesirable bird movement patterns may be related to environmental discomfort or fear. Our objective was to test the BoT-SORT object tracking architecture embedded in Yolo v8 to monitor the movement of cage-free chickens and extract measures to classify running, exploring, and resting behaviors, the latter of which includes all other behaviors that do not involve displacement. We trained a new model with a dataset of 3623 images obtained with a camera installed on the ceiling (top images) from an experiment with layers raised cage-free in small-scale aviaries and housed in groups of 20 individuals. The model presented a mAP of 98.5%, being efficient in detecting and tracking the chickens in the video. From the tracking, it was possible to record the movements and directions of individual birds, and we later classified the movement. The results obtained for a group of 20 chickens demonstrated that approximately 84% of the time, the birds remained resting, 10% of the time exploring, and 6% of the time running. The BoT-SORT algorithm was efficient in maintaining the identification of the chickens, and our tracking algorithm was efficient in classifying the movement, allowing us to quantify the time of each movement class. Our algorithm and the measurements we extract to classify bird movements can be used to assess the welfare and health of chickens and contribute to establishing standards for comparisons between individuals and groups raised in different environmental conditions.

Keywords: animal welfare; laying hens; movement rating; precision livestock farming; YOLO

1. Introduction

The welfare of chickens in a production environment depends on several conditions on the farm and the birds' health. Van Veen et al. [1] verified with laying hen producers that the activity and movement of the hens are important indicators to assess health and well-being. In this direction, several studies have sought to better understand the behavior of animals to infer welfare [2–4].

Human monitoring of animal behavior can be exhaustive and imprecise [5–7]. In this field, computer vision has gained space for using non-invasive sensors and ensuring good accuracy in analyzes [8–12].

The ability of chickens to walk may indicate lameness [13–15], while the level of agitation of the flock may indicate stress to the rearing environment [16]. Thus, tracking

chickens in a production environment has several applications for health and comfort diagnosis that can support mitigating measures and reduce losses in production [12].

Computer vision systems based on deep learning techniques make it possible to automatically discover data representations more easily, enabling the extraction of location and distance measurements [7]. The automatic detection of chickens housed on shavings bedding using computer vision cannot be considered a trivial task, as in the monitored area, there are variations in brightness, shadows, concealment by equipment, and the distortion of the camera lens itself [17]. In this sense, several studies propose different methods of detecting birds [12,16,18].

Several studies sought to track chickens in different rearing environments [6,18–22]. A common problem in tracking organisms is the loss of detection caused by traditional methods, causing ID changes during tracking. Banerjee et al. [23] tested the Strong Simple Online Real Time Tracking (Strong SORT) model on C. elegans worms and verified the effectiveness of the Id maintenance method. Young-Chul et al. [24] describe the use of a combination of association based on the proximity of Cartesian coordinates using the Hungarian algorithm and an association based on visual characteristics to maintain the trajectory of the detected object.

The BoT-SORT algorithm [25], which uses a Kalman filter and a Hungarian algorithm and is associated with the track model in Yolo, presents itself as a new possibility to track objects using a one-step detection method. We hypothesize that this tool can efficiently maintain the identification of chickens in flocks, making it possible to extract important information that may indicate states of agitation and the well-being of individuals and the flock. Our system is different from others, such as Deep-Sort [26], which is state-of-the-art in object tracking and leverages Deep Learning using a large residual network to extract features related to the appearance of an object. BoT-SORT is used to extract object movement variables such as speed, acceleration, and displacement, where these attributes were used to maintain tracking and ignore detection failures of the bird and the behavior we wanted to observe.

This study aimed to develop an intelligent system to automatically identify chickens in flocks whose housing density represents a challenge for computer vision, track the paths taken by each bird, and extract information such as distances and movement speeds of individuals.

2. Materials and Methods

2.1. Experiment and Video Recording

This study utilised a 15-minute video of 20 chickens housed in a reduced-scale shed. The camera was situated atop the aviary to capture bird images from an overhead perspective. The entire shed area was monitored by a POWER $^{®}$ camera, model AP2688W, equipped with an analog Charge-Coupled Device (CCD) image sensor. Eighty layers of the Lohmann lineage were monitored from 10 June to 8 September 2020. The birds were 29 weeks old at the beginning of the experiment and were obtained from a commercial farm. The chickens were randomly divided into four groups of 20 birds each. The feed was administered daily in the amount of 110 g/bird. Access to water was free. Light management was similar to that adopted by the original farm, with a photoperiod of 17 h of light. Four models of small-scale warehouses were used, with usable dimensions of 140 cm × 300 cm and 150 cm in height, arranged in an east-west orientation. The birds were housed on a 15 cm high wood chip bed. Each henhouse had two nest boxes (40 cm × 40 cm × 40 cm) installed side by side, a pendulum feeder (Φ 30 cm), and four nipple drinkers. The complete experiment was described by Fernandes et al. [16].

2.2. Dataset and Hardware Setup

From the set of all videos recorded in the experiment, 3623 frames were randomly extracted, divided into 3123 for training, 259 for testing, and 241 for validation. The images are 352 × 240 pixels, and the capture rate was 30 fps. In each frame, all the birds appearing

in the image were marked, using the Label Studio (https://labelstud.io/, accessed on 5 June 2023) tool (Figure 1). The coordinates of the birds' bounding boxes were saved in text files and, together with the images, formed the dataset for generating the Yolo model [27–31].

Figure 1. Image with the bounding boxes upper laying hens.

The hardware configuration of the computer utilized in this study is detailed in Table 1. The scripts for processing the videos were developed and executed using Python 3.9.16, and the Pytorch 1.13.1 framework was employed for deep learning purposes.

Table 1. Workstation hardware configuration.

CPU	CPU Intel® Core i7-10700k
CPU frequency	4.85 GHz
Cores/threads number	8/16
Memory RAM	64 GB
GPU	Nvidia RTX 2070 Super, 8 GB (2.304 CUDA cores)

2.3. Detection Model—Yolo v8n

A Yolo v8n model was generated and trained using the specified dataset for 50 epochs. Model performance was evaluated using the following metrics.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ positive} \tag{1}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{2}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{3}$$

where AP_i is the average precision for each class and N is the number of classes. In this study, mAP equals AP.

As described in Equation (4), the loss function was utilized to evaluate how the algorithm modeled the data set and achieved the desired outcome.

$$Loss_{yolo} = l_{box} + l_{conf} + l_{class} \tag{4}$$

$$l_{box} = \lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{i,j}^{obj}\left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2\right]$$

$$+\lambda_{coord}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{i,j}^{obj}\left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2\right] \tag{5}$$

$$l_{conf} = \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{noobj} \left(C_i - \hat{C}_l\right)^2 \tag{6}$$

$$l_{class} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{obj} \sum_{C \in classes} (p_i(c) - \hat{p}_i(c))^2 \tag{7}$$

where l_{box}, l_{conf}, l_{class} are the losses of location, trust and class; λ_{coord} increase weight for loss in bounding box coordinates; $I_{i,j}^{obj} = 1$ if the chicken in box j and cell i match, $I_{i,j}^{obj} = 0$ where not; $I_{i,j}^{noobj}$ it's just the opposite; x_i, y_i, w_i h_i are the coordinates (x, y) of the location, width and height of the ith bouding box; λ_{noobj} reduces loss when detecting background; C is the total number of classes, which in our model is equal to 1; $\hat{p}_i(c)$ denotes the conditional class probability for the class c in the cell i.

2.4. Tracking Algorithm

The BoT-SORT multi-object tracking model of Yolo v8n was utilized in this study, which employs a state vector based on the Kalman filter and Hungarian algorithm [25].

The tracking model was elaborated through a map of stages for the execution of the system, described in Figure 2.
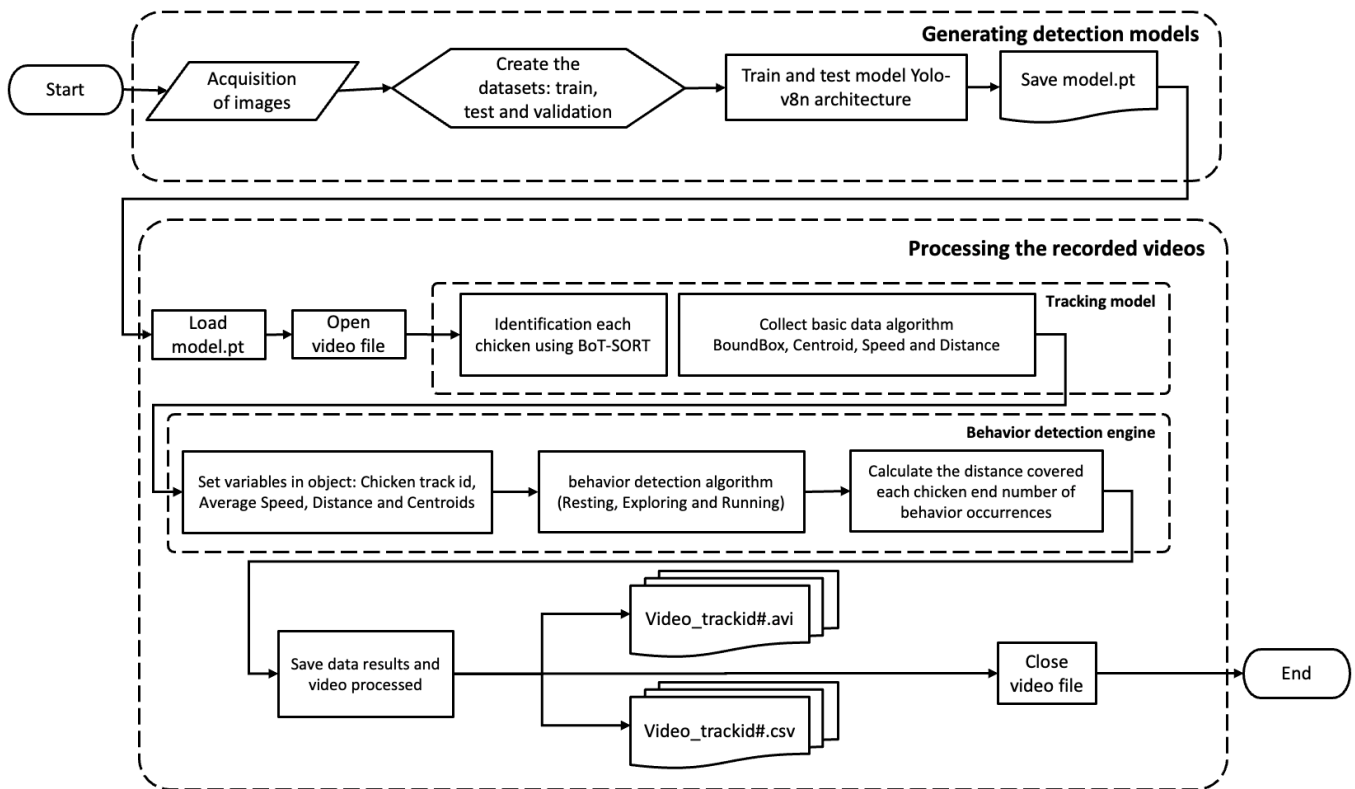

Figure 2. Stages of the tracking system development.

For the chicken tracking task, the default Kalman filter is based on a linear observation model and constant velocity movement, governed by the following linear stochastic difference equations, available at Aharon et al. [25].

$$x_k = F_k x_{k-1} + n_{k-1} \tag{8}$$

$$z_k = H_k x_k + v_k \tag{9}$$

where F_k is the transition matrix for the discrete-time of $k - 1$ for k; H_k is the matrix with observed values; n_k e v_k are random variables that represent the process and measurement noise, respectively.

The process noise covariance matrix Q_k and the measurement noise covariance matrix R_k can change at each step in time. The Kalman filter consists of predicting the current time and updating over time according to the equations given below.

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} \tag{10}$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

$$K_k = P_{k|k-1} H_k^T \left(H_k P_{k|k-1} H_k^T + R_k \right)^{-1} \tag{11}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \left(z_k - H_k \hat{x}_{k|k-1} \right)$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

For each time step k, the Kalman filter estimates the state $\hat{x}_{k|k-1}$ and the covariance matrix $P_{k|k-1}$, and later updates the later estimated state $\hat{x}_{k|k}$ given the observation z_k and the covariance matrix $P_{k|k}$, calculated based on the Kalman gain K_k.

For the BoT-SORT model, the Kalman Filter state vector is described in Equation (12) and the measurement vector in Equation (13), which is implied in the corrections of the process noise covariance matrices (Q_k) and measurement noise covariance (R_k) shown in Equations (14) and (15), respectively [16].

$$x_k = \left[x_c(k),\ y_c(k),\ w(k),\ h(k),\ \dot{x}_c(k), \dot{y}_c(k), \dot{w}(k),\ \dot{h}(k) \right] \tag{12}$$

$$z_k = \left[z_{x_c}(k),\ z_{y_c}(k),\ z_w(k),\ z_h(k) \right] \tag{13}$$

$$
\begin{aligned}
Q_k = diag\Bigg(& \left(\sigma_p \hat{w}_{k-1|k-1} \right)^2,\ \left(\sigma_p \hat{h}_{k-1|k-1} \right)^2, \\
& \left(\sigma_p \hat{w}_{k-1|k-1} \right)^2,\ \left(\sigma_p \hat{h}_{k-1|k-1} \right)^2, \\
& \left(\sigma_p \hat{w}_{k-1|k-1} \right)^2,\ \left(\sigma_p \hat{h}_{k-1|k-1} \right)^2, \\
& \left(\sigma_p \hat{w}_{k-1|k-1} \right)^2,\ \left(\sigma_p \hat{h}_{k-1|k-1} \right)^2 \Bigg)
\end{aligned} \tag{14}
$$

$$R_k = diag\left(\left(\sigma_m \hat{w}_{k|k-1} \right)^2,\ \left(\sigma_m \hat{h}_{k|k-1} \right)^2 \right), \tag{15}$$

where x_c and y_c are the coordinates of the chicken's bounding box; w and h are the width and height of the bounding box k and $k - 1$ is the current and previous state; σ_p, σ_v and σ_m are the standard deviations of the noise factors.

2.5. Measurement of Distance and Velocity of Travel

The distance covered was calculated from the coordinates of the centroids of each bird between two frames using the Euclidean method.

$$D_{(F_i - F_{i+1})} = \sqrt[2]{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{16}$$

where $D_{(Fi-Fi+1)}$ is the distance between two points, the first being captured in the frame Fi and the second in the frame F_{i+1}, x_i and y_i are the coordinates of the point in the frame F_i and x_{i+1} and y_{i+1} are the coordinates of the point in the frame F_{i+1}.

Therefore, for every two frames, we calculate the distance traveled by a chicken centroid and accumulate these values during the complete shooting. However, these values are expressed in pixels and cannot be compared with other distances from other videos due to differences in camera placement and image resolution. To make the measurements comparable, we convert pixels into centimeters using a ratio rule from a known measurement of the scene.

Birds were identified as they appeared on the scene. When the bird left the scene, the path data for that identification was closed so that the same bird, when it returned to the scene, received a new identification. Thus, the distances covered were recorded for each identification.

$$D_{Id} = \sum D_{(Fi-Fi+1)} \tag{17}$$

where are D_{id} is the distance traveled by a chicken with the identification Id by the number of sequential frames that this identification was maintained.

The speed was calculated from the distances covered by the Ids, according to the equation below.

$$S_{Id} = \frac{D_{Id}}{\Delta t} \tag{18}$$

where S_{Id} is the speed of the bird identified by Id, D_{Id} is the distance traveled by the bird Id and Δt is the time between frames (in our work $\Delta t = 1/30$ s).

2.6. Distinction of Running, Exploring, and Resting Behaviors

In the process of detecting chickens in the images, different values of displacements in time were established to differentiate three movement states: resting, exploring, and running. Initially, a noise filter was applied to eliminate erroneous identifications, along with the definition of limits that demarcate the transitions from one behavior to another.

To classify the behaviors, a minimum frame count was defined to indicate the beginning of a new behavior in chickens. Specifically, "Resting" requires a minimum of 5 frames, "Exploring" requires four frames, and "Running" requires three frames. Furthermore, to qualify for detection of one of these behaviors, a bird must maintain a minimum "lifetime" of 18 frames. This implies that the identified bird, designated by its unique track_Id, must be present in the video for at least 18 consecutive frames.

Using these data, the velocity parameter was employed to distinguish the behaviors. This involved calculating the average instantaneous velocity over the last ten frames. For the "Resting" behavior, a speed of 1 cm/frame or less was used as a limit, considering that the resting chicken can move only the head or other body parts, leading to changes in the position of the centroid, even if it remains still or without displacement.

For the "Exploring" behavior, velocities were defined that vary between more than 1 cm/s and less than 100 cm/s, determined in the analysis of the results. "Running" behavior was classified when the velocity was greater than 100 cm/s. To increase accuracy and reduce false positives when identifying "Running" behavior, an additional criterion has been implemented. This criterion measured the distance between the initial and final positions within a vector containing centroids of the last ten associated frames when the reference velocity was exceeded, and we only considered when the distance measured between the extremes was greater than 20 cm. In our tests, this condition differentiated the "Running" behavior from other behaviors with fast movements and without significant movement of the hen.

3. Results

3.1. Yolo Tracking Model

Figure 3 shows the mAP (mean average precision) and Loss plots of the generated Yolo v8n model. The model achieved 98.5% accuracy. These plots depict the evolution of

mAP and loss throughout training, offering a detailed view of the model's performance in chicken identification. The model demonstrated robustness, as the mAP curve remained stable above 98% despite a decrease in the loss curve starting from epoch 15. This suggests the model continues to identify chickens with high precision after an initial training phase.
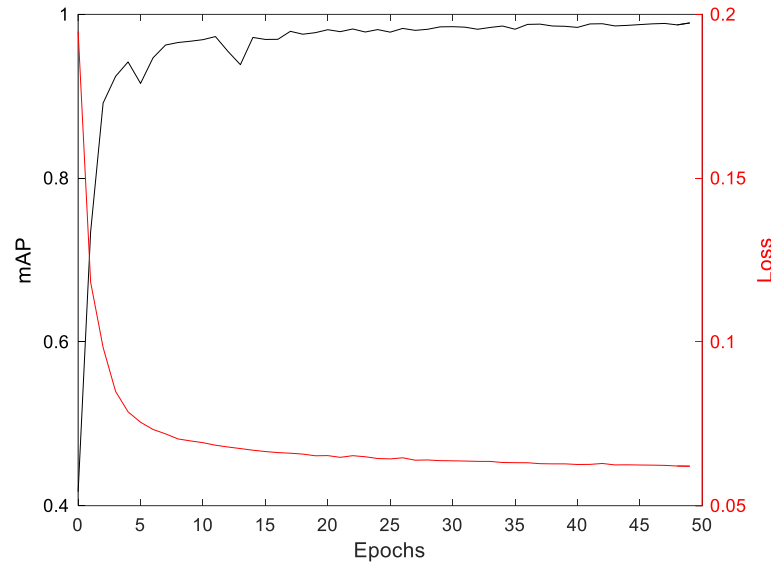

Figure 3. Graph of the evolution of the mean average precision (mAP, black line) and loss (red line) during the Yolo v8n chicken detection model training.

As illustrated in the images presented in Figure 4, our system was capable of detecting chickens within the nest, albeit with less certainty. Moreover, certain areas, such as those behind the feeder, remained concealed. Additionally, the impact of camera lens distortion on image quality at the periphery also merits consideration.
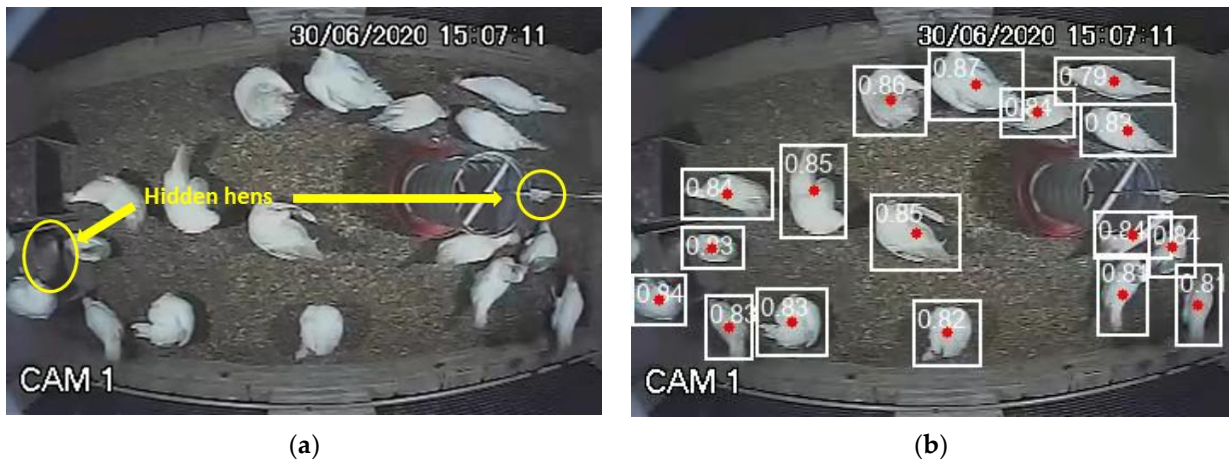

(**a**)   (**b**)

Figure 4. The original frame (**a**) was processed with Yolo v8n (**b**) showing that the model's confidence value in detecting laying hens inside the nest and behind the feeder is low.

The quality of detection of chickens is fundamental for successful tracking. Chickens that can be confused with background, or vice versa, generate centroid coordinates that have the potential to interfere with the identification of birds by the Kaman filter. This is a crucial point in the development of the tracking system.

3.2. Appearing and Hiding of Chickens

We had problems hiding the chickens in the scene during our tracking tests. In Figure 4, we highlight two common places where the chicken identification is lost in the image due

to obstruction of objects. In Figure 5, we show the example of the path of two chickens that appeared and disappeared in the scene near the feeder and near the nest.
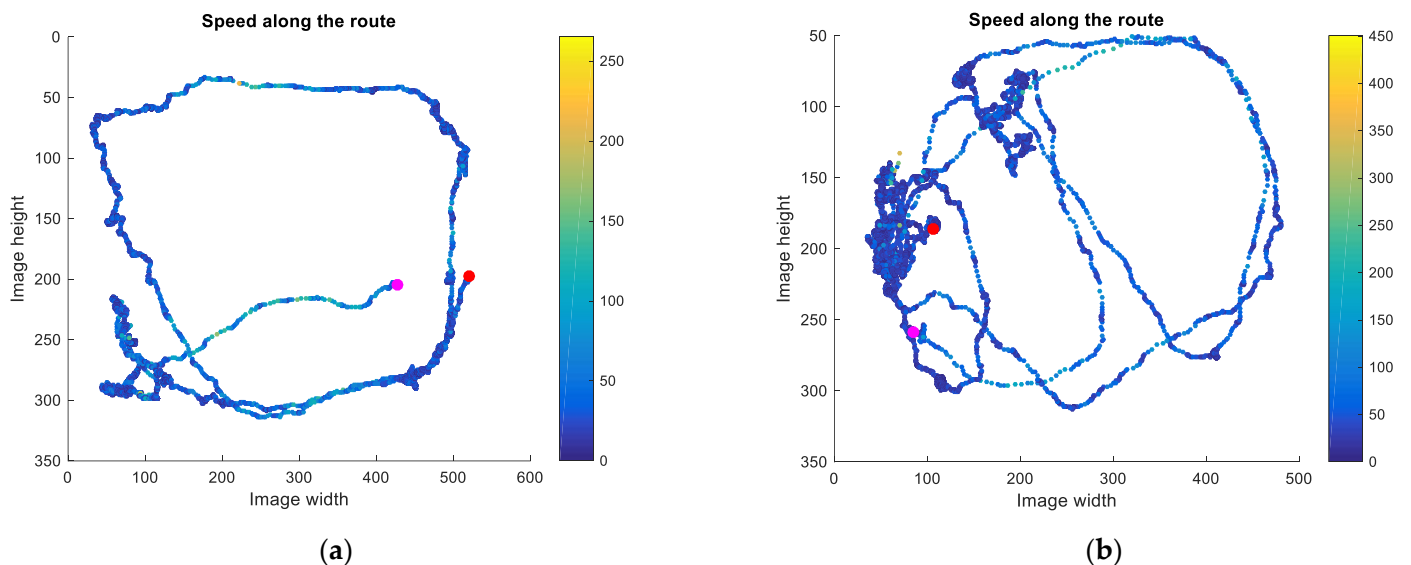

Figure 5. Path of the birds with the marks of the beginning and end of the path (magenta and red, respectively) and highlighting the moments when the bird walked faster (green and yellow dots) for the hen-line (**a**) track_Id 5 that appeared and disappeared near the feeder and (**b**) track_Id 45 that appeared and disappeared near the nest.

Markings in magenta and red correspond respectively to the first and last coordinates of chicken detection in the video, that is, the beginning and end of the mapped path. Along the way, the more yellow the marking, the faster the chicken travels. We can observe that the places of emergence and disappearance of chickens in Figure 5 are close to the nest or feeder. This was due to the already mentioned fact that we lost the identification of the bird when it left the scene, entered the nest, or passed behind the feeder. This condition is expected in the case of monitoring chickens in commercial farms, as the camera capture area will be smaller than the area of the shed, and, therefore, there will be the appearance and disappearance of chickens (same individuals or not) along the edges of the image. Thus, we decided to work with this feature, allowing our methodology to be applied in commercial breeding conditions in the future.

3.3. Direction of Displacement and Minimum Displacement

In the path graphs, we observe that there are times when the bird walks faster, and in others, there is minimal movement, suggesting that the bird is standing still, expressing some natural behavior.

As can be seen in Figure 6, our processing considered small lateral displacements, accumulating as distances covered by the birds. However, situations like the one illustrated can refer to a bird standing still, resting, or expressing some natural behavior. Therefore, our code considers distances greater than 1 cm/frame as displacement of the bird. The variation in the displacement direction also seems to contribute to the identification of the hen's state of movement, so we also started to observe this information.
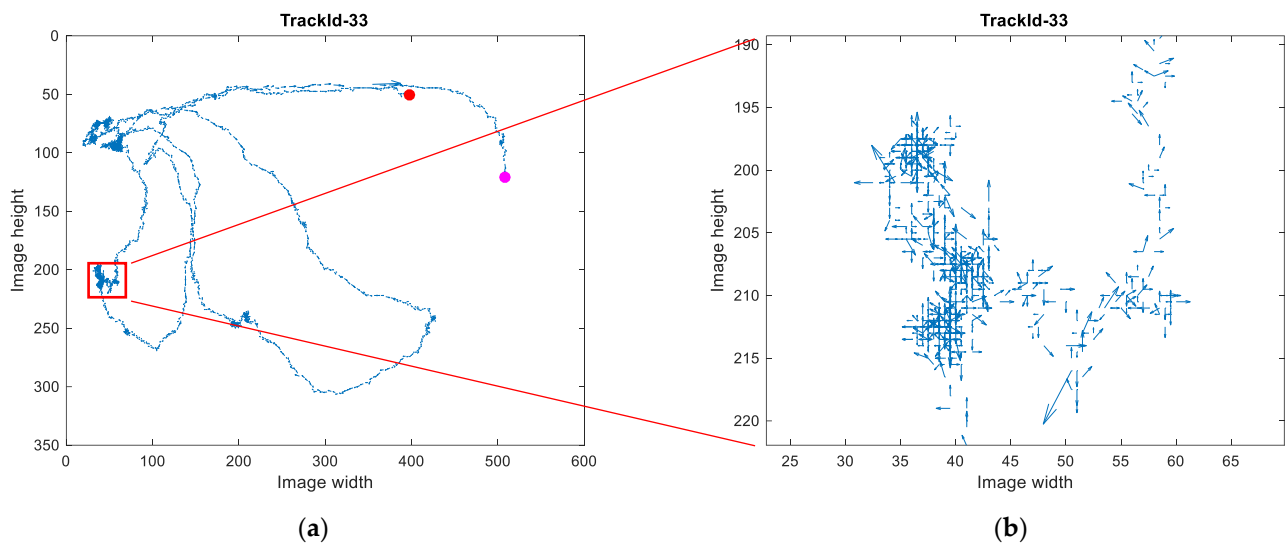
(a)   **(b)**

Figure 6. Arrow chart of chicken track_id 33 (**a**), highlighting a moment when the bird was still (**b**). The red and magenta dots correspond, respectively, to the beginning and end of tracking the monitored chicken.

3.4. Measures to Determine the State of Movement

Once the correction was made, we looked for metrics indicating when a bird was standing still, walking, or running. Analyzing the velocity graph in Figure 7, we observed that the velocity of this displacement varied at values below 100 cm/s when the bird was walking and peaked much above this value when the bird ran.
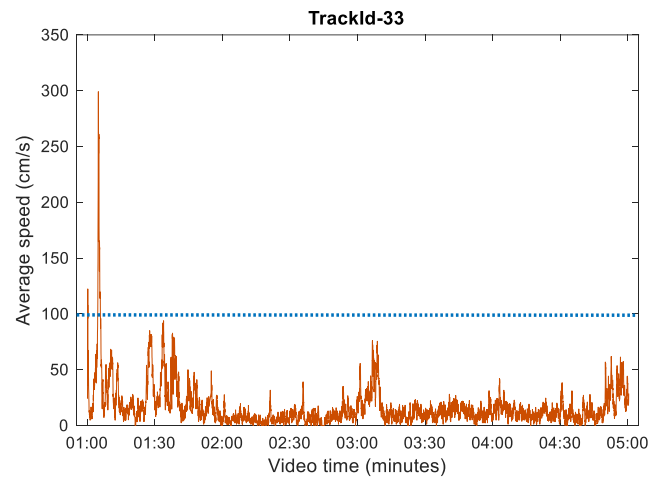

Figure 7. Graph of variation of the average speed of chicken track_Id 33 every ten frames, where the peak speed corresponds to this chicken running.

Figure 7 presents a graph depicting the variation in velocity during the time a bird was tracked by our model. This particular bird was selected for analysis as it exhibited all three behaviors during the tracking period. At the onset of tracking, a spike in speed is observed, confirmed in the original video as indicative of the running behavior. This pattern was also observed in other birds, and a velocity threshold of 100 cm/s was established for classifying running behavior in our code. Resting behavior is characterized by the bird remaining completely stationary, disregarding any minor displacements (<1 cm/frame). Consequently, any movement detected with a velocity between 0 and 100 cm/s was classified as exploring behavior. To minimize any transitional effects between behaviors, the average speed of the last ten frames (0.33 s) was considered.

3.5. Movement Analysis

Our surveillance and tracking system can identify a "new chicken" instantly. As previously reported, there are covered regions in our footage, and we found that there are chickens that remain with the same identification for a very short time. There are also those that, during the nesting behavior, even if they maintain the identification, remain still during the entire vigil. Our objective is to develop an intelligent system capable of automatically identifying chickens in flocks whose rearing density. In this sense, we focused our attention on chickens that walked more than 300 cm and were monitored by our system for more than 200 s without losing identification, and we made the graphs shown in Figure 8.
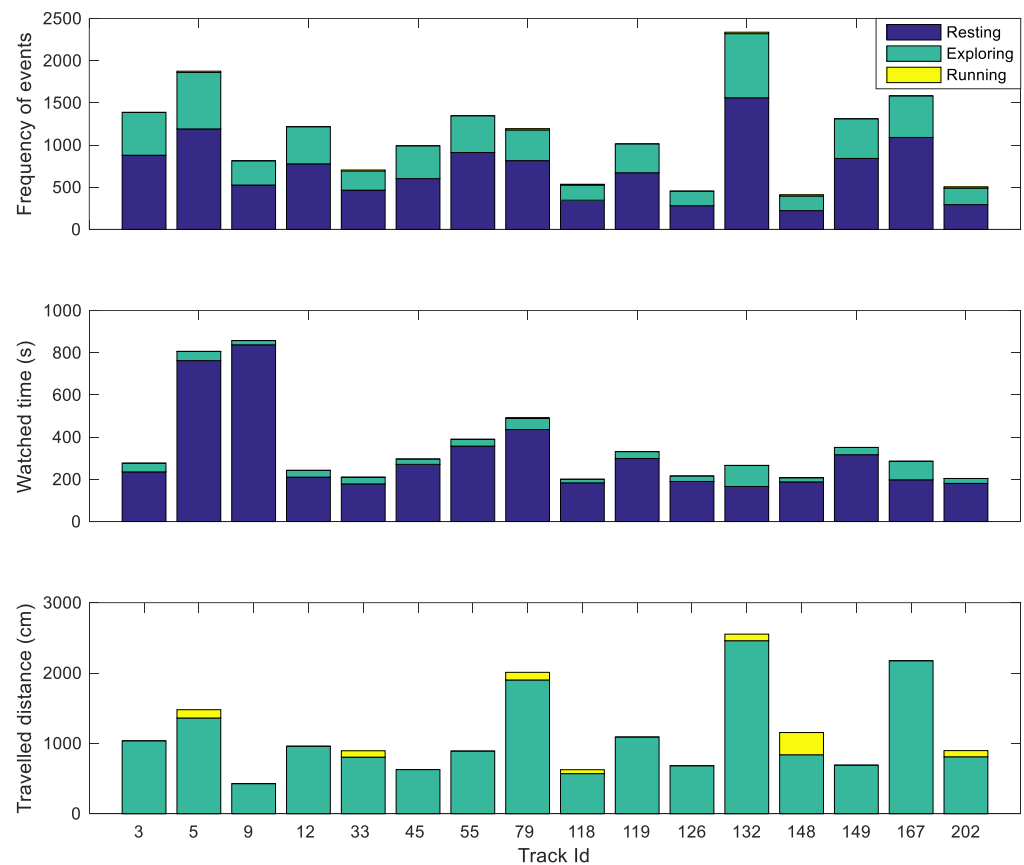

Figure 8. Graphs of frequency of occurrences, total observation time, and total distance covered for resting, exploring, and running behaviors for birds that walked more than 300 cm and were watched for more than 200 s without losing identification.

The chicken with track_Id = 148 caught our attention, which had a greater displacement per run than the other birds. Analyzing the trajectory graph, we noticed that in the region where the chicken supposedly ran, the direction vectors were misaligned. Unlike what happened with the chicken track_Id = 33, during the peak velocity, the direction vectors were aligned. We searched the video for the exact frames of each chicken for these occurrences and verified that chicken 33 ran at that moment and chicken 148 flapped its wings. Figure 9 shows the graphs and frames, highlighting the observed differences.

As observed in Figure 9, the peaks in speed are highly similar for distinct behaviors, indicating that this measure alone is insufficient for distinguishing between these behaviors. Based on these observations, the variation in the direction of movement of the birds was analyzed and compared with their speed. The tangent angle of the chicken's displacement vector for each analyzed frame indicates the intensity of the change of direction of the chicken's path. It was observed that when this route change is abrupt and associated with high speed, there is a high possibility that the bird is opening its wings. When the speed is

high and there is no change in route, it is assumed that the chicken is running. When the route changes but the speed is considered normal, it is assumed that the bird has simply changed direction and continued walking.
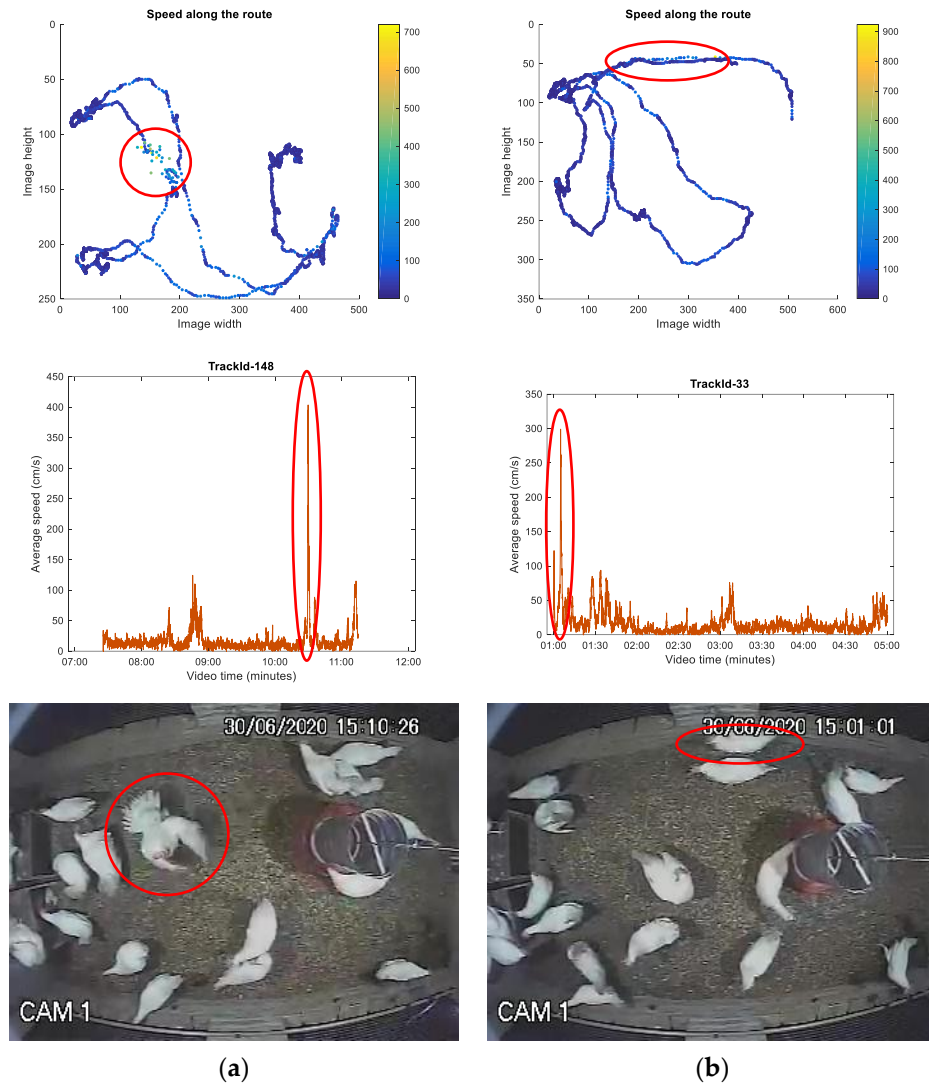

Figure 9. Graphs of paths, variation in displacement average speed, and frame of the original video highlighting the differences in measurements observed for the chicken (**a**) track_Id = 148, which expresses the behavior of flapping wings, and (**b**) track_Id = 33, which is running. The circles mark the data in the graphs corresponding to the chickens at a given moment in the video (images) expressing different behaviors.

For the running chicken, we used the measurement of the distance between the centroids of the extreme points during the last ten frames. This approach allows the detection of the directionality of the bird's movement. Consideration of speed is of fundamental importance, as it constitutes a discriminating factor between the "Exploring" and "Running" states. In Figure 10, during flapping, the chicken identification track_Id = 79 is segregated into two identifications (track_Id = 79 and track_Id = 97). After the expression of the behavior, identification 97 ceases to exist. During flapping, there is a rapid change in the positioning of the centroid of the original identification (track_Id = 79), resulting in a speed that falls within the predefined limits for the "Running" and "Exploring" states. The generation of new identifications in different bird regions during wing flapping can also cause deviations in the centroids, mimicking the presence of another bird. The finding of a distance between the centroids of these identifications above 20 cm characterizes the

"Running" behavior. To mitigate the influence of spurious identifications, we adopted a rule requiring a minimum number of 18 frames (called "lifetime") in which the bird remains in the video. The limit of 18 frames was chosen based on observations since birds frequently performed movements that caused abrupt changes in the image centroid. This strategy aims to reduce confusion between genuine behaviors and artifactual occurrences.
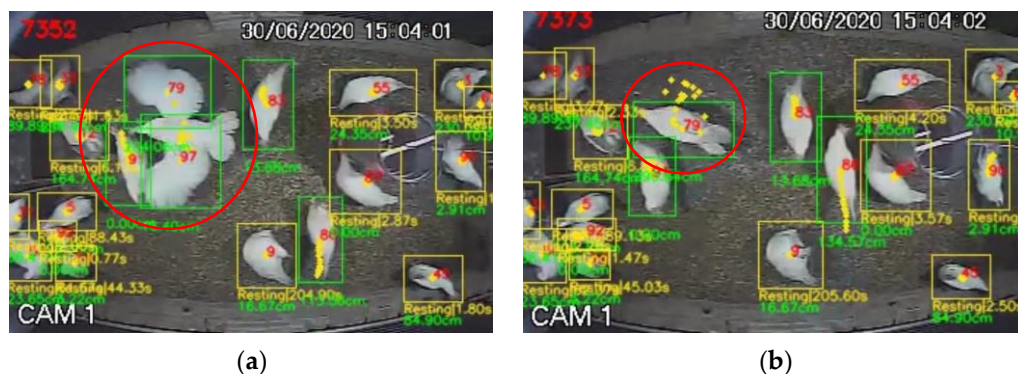

(a) (b)

Figure 10. Capture of frames showing flapping wings. (**a**) chicken 79 was divided into two birds in identification. (**b**) chicken 97 is ceasing to exist.

We implemented these rules in our code (pseudo-code in Appendix A). The processed video that classifies bird movement with all these metric considerations is shown in the Supplementary Material.

In Table 2, we show a summary of the time spent by the birds resting, exploring, and running, which can be used to analyze the state of welfare or health of the birds in time intervals of interest.

Table 2. Summary of occurrences and times spent by birds running, exploring, and resting, disregarding Ids with a minimum lifetime.

Behavior Class	Number of Occurrences (N)	Overall Time (T) Seconds	Average Time (T/N)Seconds
Resting	808	13,537	16.8
Exploring	792	1490	1.9
Running	7	9	1.3

Table 2 presents a summary of data obtained from the entire 15-minute video. This summary can be generated for any desired time interval. It was observed that the birds spend most of their time stationary, resting or engaging in other natural behaviors that do not involve movement. The majority of the route traveled by the bird is spent exploring. A few instances of running behavior were observed, and it was verified that this behavior is expressed for very short periods.

We found no reference values for these behaviors in the literature. However, we hypothesize that by extracting precise measurements of these behaviors, it will be possible to establish reference values for each lineage, allowing the welfare of chickens submitted to different housing environments to be compared.

4. Discussion

Proposing automatic, accurate, and non-invasive approaches to monitoring animals in production is necessary to improve the efficiency and welfare of animals in housing [32]. In this sense, several studies have used convolutional neural networks (CNN) to detect and classify chicken behavior. Neethirajan [19] and Yang et al. [20] demonstrated the feasibility of using Yolo to detect chickens in several videos, suggesting applications such as counting and tracking. Yang et al. [20] classified six behaviors using Yolo-v5 with accuracies ranging from 72.3% to 95.3%. Li et al. [33] developed a computer vision system to detect chickens

in feeders and drinkers. Nasiri et al. [32] estimated feeding time for broilers using Yolo-v3 to detect when heads were within the feeder area.

Individual and group behavior are important indicators of the welfare and health of birds [34–36]. Our work provides a breakthrough for tracking chickens, demonstrating that the Kalman filter and the BoT-SORT algorithm efficiently maintain the identification of birds in our videos, where the density and overlapping of bodies were a challenge for computer vision.

The BoT-SORT algorithm is highly effective in maintaining accurate identification of chickens throughout the tracking process. With our tracking algorithm, we could carry out a detailed analysis of the distances covered and the movement speeds of the chickens. This ability to identify and categorize movement behaviors makes it possible to analyze bird behavior in different environmental conditions.

Measuring the distances covered individually by the hens can indicate the activity level of the flock of individuals and can also be used to assess the gait score [37]. Campbell et al. [2] found that aviary resources were used differently by individual hens and that understanding these usage patterns by both individuals and the group will allow the optimization of cageless housing system designs. Jaihumi et al. [13] developed a Yolo v5 model to assess the mobility of broilers, and Fernández et al. [6] found an association between activity level and patterns of aviary occupancy with hock and leg injuries in broilers using computer vision.

By observing movement patterns, we could detect significant differences between individuals in the monitored group of chickens. These individual distinctions in movements are necessary to assess bird welfare and health. For example, we can identify birds exhibiting anomalous behaviors indicative of health problems or discomfort, allowing for early intervention to ensure appropriate treatment.

Collins [38] observed that some chickens walked greater distances than others. The motivation for moving laying hens may differ from that of broilers. However, in our results, we also observed that there are hens that moved more and at greater speeds than others. Taylor et al. [39] found a relationship between chickens that covered greater distances and increased welfare, and these animals had lower weight, better gait scores, and lower corticosterone responses. Considering chickens under the same rearing model, the observed differences in movement measured by our tracking system may indicate more adapted individuals.

The movement measurements obtained and the tracking algorithm developed in this study provide a solid basis for future investigations. This model can assess the welfare and health of chickens and classify and understand other natural behaviors, thus contributing to significant advances in poultry farming and animal welfare research. Such information is essential to improve poultry farming practices, ensuring a healthier and more comfortable environment for these animals.

Limitations and Suggestions for Future Studies

Our tracking method is limited in regions where the camera is obstructed, such as behind equipment, edges of images, and inside nests. Obstruction of the image by equipment was also observed in other studies [21]. In our tests, we preferred not to deal with this problem because, in commercial situations, there will be regions where the bird will leave the camera's field of view. We understand it is better to use strategies to analyze the data recorded this way, as already discussed in other works.

Our footage was low resolution and heavily distorted around the edges. The use of high-resolution cameras with less edge distortion can improve the detection accuracy, the conversion scale from pixels to centimeters, and, consequently, the measurements of the distances covered by the birds. Addressing these distortions can give greater accuracy in displacement measurements. More complex images are subject to greater chances of computational model confusion [20]. In this field, it is still necessary for work to demonstrate the stability of a flock tracking system and longer filming times.

We adopted criteria to reduce confusion in classifying the running behavior with other natural behaviors. We raised the hypothesis that these criteria can be better worked to contribute to the classification of natural behaviors. Dawkins [40] and Kashima et al. [41] proposed that behavior can be used to estimate the need and health of animals, highlighting that behavior analysis should have a more prominent role when used with new technologies. Thus, other studies that seek to classify individual or group behaviors autonomously are necessary and will constitute the future of industrial poultry for monitoring the welfare and health of chickens.

5. Conclusions

Our model obtained a mean average precision of 98.5% in detecting chickens, and the BoT-SORT algorithm was efficient in maintaining the identification of these birds during tracking. From the tracking, it was possible to analyze the distances covered and the speeds of the chickens, which made it possible to classify their movement behaviors. Movement speeds above 100 cm/s of chickens with a lifetime of more than 18 frames were sufficient to differentiate running behavior from exploring behavior. In this study, the tests showed that the chickens spent approximately 85% of their time resting, 10% of their time exploring the environment, and 6% running. Our algorithm and the measures extracted to classify chicken movement contribute to the assessment of chicken welfare and health.

Appendix A

Algorithm: Chicken tracking code and movement classification

Input: Video or stream of video
Output: Video with detections and file of data about this experiment

```
1     # Minimum distance to consider an object has moved
2     min_distance = 3
3
4     # Thresholds to determine whether the object is Resting, Exploring, or Running
5     resting_speed_threshold = 0.8
6     exploring_speed_threshold = 4.17
7
8     # Minimum consecutive counts for each state
9     min_count_resting = 5
10    min_count_exploring = 4
11    min_count_running = 3
12
13    #Minimum distance between 10 frames in running state
14    distance_running_min = 25
15
16    #Lifetime from object detected in a video
17    lifetime = 18
18
19    # Pixel to centimeter conversion
20    pixel_cm = 0.8
21
22    FOR each tracked object with track_id:
23        calculate average_speed using the last 10 frames
24        assign average_speed to the speed property of the tracked object
25
26        IF tracked object's lifetime is less than the lifetime:
27            Discard the object's behavior
28        IF the object's state is "Running" and the distance between the first and last
           frame centroids in the last 10 frames is less than distance_running_min:
29            Discard the object's behavior
30        IF the object's speed is less than or equal to resting_speed_threshold:
31            increment count_resting
32            reset counts count_exploring and count_running
33            reset distance dist_resting
34            IF the object's previous state was not "Resting":
35                increment freq_resting
36            IF count_resting is greater than or equal to min_count_resting:
37                set the object's state as "Resting" and time as "Resting"
38        ELSE IF the object's speed is less than exploring_speed_threshold:
39            reset count_resting
40            increment count_exploring
41            reset count_running
42            IF distance is greater than min_distance:
43                increment distance dist_exploring
44            IF the object's previous state was not "Exploring":
45                increment freq_exploring
46            IF count_exploring is greater than or equal to min_count_exploring:
47                set the object's state as "Exploring" and time as "Exploring"
48        ELSE IF the object's speed is greater than or equal to exploring_speed_threshold:
49            reset count_resting
50            reset count_exploring
51            increment count_running
52            IF distance is greater than min_distance:
53                increment distance dist_running
54            IF the object's previous state was not "Running":
55                increment freq_running
56            IF count_running is greater than or equal to min_count_running:
57                set object's state as "Running" and time as "Running"
```

References

1. van Veen, L.A.; van den Oever, A.C.M.; Kemp, B.; van den Brand, H. Perception of laying hen farmers, poultry veterinarians, and poultry experts regarding sensor-based continuous monitoring of laying hen health and welfare. *Poult. Sci.* **2023**, *102*, 102581. [CrossRef] [PubMed]
2. Campbell, D.; Karcher, D.M.; Siegford, J.M. Location tracking of individual laying hens housed in aviaries with different litter substrates. *Appl. Anim. Behav. Sci.* **2016**, *184*, 74–79. [CrossRef]
3. Grebey, T.C.; Ali, A.B.A.; Swanson, J.C.; Widowski, T.M.; Siegford, J.M. Dust bathing in laying hens: Strain, proximity to, and number of conspecifics matter. *Poult. Sci.* **2020**, *99*, 4103–4112. [CrossRef]
4. Vasdal, G.; Muri, K.; Stubsjøen, S.M.; Moe, R.O.; Kittelsen, K. Qualitative behaviour assessment as part of a welfare assessment in flocks of laying hens. *Appl. Anim. Behav. Sci.* **2022**, *246*, 105535. [CrossRef]
5. Geffen, O.; Yitzhaky, Y.; Barchilon, N.; Druyan, S.; Halachmi, I. A machine vision system to detect and count laying hens in battery cages. *Animal* **2020**, *14*, 2628–2634. [CrossRef] [PubMed]
6. Fernandez, A.P.; Norton, T.; Tullo, E.; van Hertem, T.; Youssef, A.; Exadaktylos, V.; Vranken, E.; Guarino, M.; Berckmans, D. Real-time monitoring of broiler flock's welfare status using camera-based technology. *Biosyst. Eng.* **2018**, *173*, 103–114. [CrossRef]
7. Li, G.; Li, B.; Shi, Z.X.; Lu, G.; Chai, L.; Rasheed, K.M.; Regmi, P.; Banakar, A. Inter-Individual Distances and Orientations of Laying Hens under Eight Stocking Densities Measured by Integrative Deep Learning Techniques. *Poult. Sci.* **2023**, *102*, 103076. [CrossRef] [PubMed]
8. Yang, X.; Zhao, Y.; Gan, H.; Hawkins, S.; Eckelkamp, L.; Prado, M.; Burns, R.; Purswell, J.; Tabler, T. Modeling gait score of broiler chicken via production and behavioral data. *Animal* **2023**, *17*, 100692. [CrossRef]
9. Subedi, S.; Bist, R.; Yang, X.; Chai, L. Tracking pecking behaviors and damages of cage-free laying hens with machine vision technologies. *Comput. Electron. Agric.* **2023**, *204*, 107545. [CrossRef]
10. Fang, C.; Huang, J.; Cuan, K.; Zhuang, X. Comparative study on poultry target tracking algorithms based on a deep regression network. *Biosyst. Eng.* **2020**, *190*, 176–183. [CrossRef]
11. Guo, Y.; Aggrey, S.; Oladeinde, A.; Johnson, J.; Zock, G.; Chai, L. A Machine Vision-Based Method Optimized for Restoring Broiler Chicken Images Occluded by Feeding and Drinking Equipment. *Animals* **2021**, *11*, 123. [CrossRef] [PubMed]
12. Li, N.; Ren, Z.; Li, D.; Zeng, L. Review: Automated techniques for monitoring the behaviour and welfare of broilers and laying hens: Towards the goal of precision livestock farming. *Animal* **2020**, *14*, 617–625. [CrossRef]
13. Jaihuni, M.; Gan, H.; Tabler, T.; Prado, M.; Qi, H.; Zhao, Y. Broiler Mobility Assessment Via a Semi-Supervised Deep Learning Model and Neo-Deep Sort Algorithm. *Animals* **2023**, *13*, 2719. [CrossRef] [PubMed]
14. Fodor, I.; van der Sluis, M.; Jacobs, M.; Klerk, B.; Bouwman, A.C.; Ellen, E.D. Automated pose estimation reveals walking characteristics associated with lameness in broilers. *Poult. Sci.* **2023**, *102*, 102787. [CrossRef] [PubMed]
15. Nasiri, A.; Yoder, J.; Zhaob, Y.; Hawkins, S.; Prado, M.; Gana, H. Pose estimation-based lameness recognition in broiler using CNN-LSTM network. *Comput. Electron. Agric.* **2022**, *197*, 106931. [CrossRef]
16. Fernandes, A.M.; Sartori, D.L.; Morais, F.J.O.; Salgado, D.D.; Pereira, D.F. Analysis of Cluster and Unrest Behaviors of Laying Hens Housed under Different Thermal Conditions and Light Wave Length. *Animals* **2021**, *11*, 2017. [CrossRef]
17. Trnovszky, T.; Kamencay, P.; Orjesek, R.; Benco, M.; Sykora, P. Animal recognition system based on convolutional neural network. *Adv. Electr. Electron. Eng.* **2017**, *15*, 517–525. [CrossRef]
18. Siriani, A.L.R.; Kodaira, V.; Mehdizadeh, S.A.; Nääs, I.A.; Moura, D.J.; Pereira, D.F. Detection and tracking of chickens in low-light images using YOLO network and Kalman filter. *Neural Comput. Appl.* **2022**, *34*, 21987–21997. [CrossRef]
19. Neethirajan, S. ChickTrack—A quantitative tracking tool for measuring chicken activity. *Measurement* **2022**, *191*, 110819. [CrossRef]
20. Yang, X.; Bist, R.; Subedi, S.; Wu, Z.; Liu, T.; Chai, L. An automatic classifier for monitoring applied behaviors of cage-free laying hens with deep learning. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106377. [CrossRef]
21. Yang, X.; Bist, R.; Subedi, S.; Chai, L. A deep learning method for monitoring spatial distribution of cage-free hens. *Artif. Intell. Agric.* **2023**, *8*, 20–29. [CrossRef]
22. Cui, Y.; Kong, X.; Chen, C.; Lia, Y. Research on broiler health status recognition method based on improved YOLOv5. *Smart Agric. Technol.* **2023**, *in press*. [CrossRef]
23. Banerjee, S.C.; Khan, K.A.; Sharma, R. Deep-worm-tracker: Métodos de aprendizagem profunda para detecção e rastreamento precisos para estudos comportamentais em C. elegans. *Appl. Anim. Behav. Sci.* **2023**, *266*, 106024. [CrossRef]
24. Young-Chul, Y.; Kim, D.Y.; Young-Min, S.; Yoon, K.; Jeon, M. Online Multiple Pedestrians Tracking using Deep Temporal Appearance Matching Association. *arXiv* **2020**, arXiv:1907.00831. [CrossRef]
25. Aharon, N.; Orfaig, R.; Bobrovsky, B.Z. BoT-SORT: Robust Associations Multi-Pedestrian Tracking. *arXiv* **2022**, arXiv:2206.14651v2. [CrossRef]
26. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. IEEE internation-al conference on image processing (ICIP). *arXiv* **2017**, arXiv:1703.07402. [CrossRef]
27. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
28. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767. [CrossRef]

29. Redmon, J.; Farhadi, A. Yolo9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525. [CrossRef]
30. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; SUN, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430. [CrossRef]
31. Yang, X.; Chai, L.; Bist, R.B.; Subedi, S.; Wu, Z. A Deep Learning Model for Detecting Cage-Free Hens on the Litter Floor. *Animals* **2022**, *12*, 1983. [CrossRef]
32. Nasiri, A.; Amirivojdan, A.; Zhao y Gan, H. Estimating the Feeding Time of Individual Broilers via Convolutional Neural Network and Image Processing. *Animals* **2023**, *13*, 2428. [CrossRef]
33. Li, G.; Zhao, Y.; Purswell, J.L.; Du, Q.; Chesser, G.D., Jr.; Lowe, J.W. Analysis of feeding and drinking behaviors of group-reared broilers via image processing. *Comput. Electron. Agric.* **2020**, *175*, 105596. [CrossRef]
34. Papageorgiou, M.; Goliomytis, M.; Tzamaloukas, O.; Miltiadou, D.; Simitzis, P. Positive Welfare Indicators and Their Association with Sustainable Management Systems in Poultry. *Sustainability* **2023**, *15*, 10890. [CrossRef]
35. Jacob, F.G.; Nääs, I.A.; Salgado, D.D.; Baracho, M.S.; Lima, N.D.S.; Pereira, D.F. Does Environmental Enrichment with Music and Strobe Light Affect Broilers' Welfare? Analyzing Their On-Farm Reaction. *Agriengineering* **2022**, *4*, 707–718. [CrossRef]
36. Pereira, D.F.; Lopes, F.A.A.; Gabriel Filho, L.R.A.; Salgado, D.D.; Mollo Neto, M. Cluster index for estimating thermal poultry stress (gallus gallus domesticus). *Comput. Electron. Agric.* **2020**, *177*, 105704. [CrossRef]
37. Li, G.; Gates, R.S.; Meyer, M.M.; Bobeck, E.A. Tracking and Characterizing Spatiotemporal and Three-Dimensional Locomotive Behaviors of Individual Broilers in the Three-Point Gait-Scoring System. *Animals* **2023**, *13*, 717. [CrossRef] [PubMed]
38. Collins, L.M. Non-intrusive tracking of commercial broiler chickens in situ at different stocking densities. *Appl. Anim. Behav. Sci.* **2008**, *112*, 94–105. [CrossRef]
39. Taylor, P.S.; Hemsworth, P.H.; Groves, P.J.; Gebhardt-Henrich, S.G.; Rault, J.L. Frequent range visits further from the shed relate positively to free-range broiler chicken welfare. *Animal* **2020**, *14*, 138–149. [CrossRef]
40. Kashima, M.A.; Green, A.R.; Sales, T.G.; Bahr, C.; Berckmans, D.; Gates, R.S. Performance of an image analysis processing system for hen tracking in an environmental preference chamber. *Poult. Sci.* **2014**, *93*, 2439–2448. [CrossRef]
41. Dawkins, M.S. Using behaviour to assess animal welfare. *Anim. Welf.* **2004**, *13*, 3–7. [CrossRef]