2014

Automated Tracking and Behavior Quantification of Laying Hens Using 3D Computer Vision and Radio Frequency Identification Technologies

Akash D. Nakarmi  
*Iowa State University*, akashdev@iastate.edu

Lie Tang  
*Iowa State University*, lietang@iastate.edu

Hongwei Xin  
*Iowa State University*, hxin@iastate.edu

Follow this and additional works at: http://lib.dr.iastate.edu/abe_eng_pubs

Part of the Agriculture Commons, Bioresource and Agricultural Engineering Commons, and the Poultry or Avian Science Commons

The complete bibliographic information for this item can be found at http://lib.dr.iastate.edu/abe_eng_pubs/612. For information on how to cite this item, please visit http://lib.dr.iastate.edu/howtocite.html.
AUTOMATED TRACKING AND BEHAVIOR QUANTIFICATION OF LAYING HENS USING 3D COMPUTER VISION AND RADIO FREQUENCY IDENTIFICATION TECHNOLOGIES

A. D. Nakarmi, L. Tang, H. Xin

ABSTRACT. Housing design and management schemes (e.g., bird stocking density) in egg production can impact hens’ ability to perform natural behaviors and production economic efficiency. It is therefore of socio-economic importance to quantify the effects of such schemes on laying-hen behaviors, which may in turn have implications on the animals’ wellbeing. Video recording and manual video analysis is the most common approach used to track and register laying-hen behaviors. However, such manual video analyses are labor intensive and are prone to human error, and the number of target objects that can be tracked simultaneously is small. In this study, we developed a novel method for automated quantification of certain behaviors of individual laying hens in a group-housed setting (1.2 m × 1.2 m pen), such as locomotion, perching, feeding, drinking, and nesting. Image processing techniques were employed on top-view images captured with a state-of-the-art time-of-flight (ToF) of light based 3D vision camera for identification as well as tracking of individual birds in the group with support from a passive radio-frequency identification (RFID) system. Each hen was tagged with a unique RFID transponder attached to the lower part of her leg. An RFID sensor grid consisting of 20 antennas installed underneath the pen floor was used as a recovery system in situations where the imaging system failed to maintain identities of the birds. Spatial as well as temporal data were used to extract the aforementioned behaviors of each bird. To test the performance of the tracking system, we examined the effects of two stocking densities (2880 vs. 1440 cm² hen⁻¹) and two perching spaces (24.4 vs. 12.2 cm of perch per hen) on bird behaviors, corresponding to five hens vs. ten hens, respectively, in the 1.2 m × 1.2 m pen. The system was able to discern the impact of the physical environment (space allocation) on behaviors of the birds, with a 95% agreement in tracking the movement trajectories of the hens between the automated measurement and human labeling. This system enables researchers to more effectively assess the impact of housing and/or management factors or health status on bird behaviors.

Keywords. 3D vision, Behavior monitoring, Laying hen, RFID, Stocking density, Tracking.

The spatial requirement for laying hens and its impact on their welfare remains one of the most debated topics among egg producers and advocates of animal welfare. With the 2012 European Union (EU) ban on conventional cages for laying hens and recent developments in the U.S., non-cage or alternative housing systems are likely to become more predominant (Zimmerman et al., 2006). The United Egg Producers (UEP) and consumer food chain McDonald’s put forward welfare guidelines in 2000. The UEP guidelines recommended that cage floor space be increased over a five-year period, ending in 2008, from the U.S. industry standard of 348 cm² hen⁻¹ to a range of 432 to 555 cm² hen⁻¹ (UEP, 2000), whereas the McDonald’s recommended welfare practices call for cage floor space of 465 cm² hen⁻¹ (McDonald’s, 2000). The EU, on the other hand, recommended cage floor space for conventional cages to be 550 cm² hen⁻¹ until 2012 (Hy-Line, 2000). Without well-controlled, large-scale experiments, it is difficult to assert if and how increasing space allocation actually improves the welfare of laying hens. Different potential indicators of welfare status should be considered before the effect of stocking density (SD) can be assessed.

Researchers have explored many possible indicators of welfare and methods of measurement. Behavior is one such important indicator of animal welfare. Xin and Ikeguchi (2001) developed a measurement system to quantify feeding behavior of individual poultry in order to study effects of biophysical factors such as light, ration, noise, and thermal variables. Gates and Xin (2001) developed and tested algorithms for determining individual feeding statistics and pecking behavior from time-series recordings of feeder weight. Puma et al. (2001) developed an instrumentation system to study dynamic feeding and drinking behaviors of individual birds. Persyn et al. (2004) used the measurement system and computational algorithm developed by Xin and Ikeguchi (2001) to quantify feeding behaviors of pullets and laying hens with or without beak trimming. Cook et al.

Submitted for review in November 2013 as manuscript number SE 10505; approved for publication by the Structures & Environment Division of ASABE in July 2014.

The authors are Akash D. Nakarmi, Postdoctoral Fellow, Lie Tang, ASABE Member, Associate Professor, and Hongwei Xin, ASABE Fellow, Director of Egg Industry Center and Distinguished Professor, Department of Agricultural and Biosystems Engineering, Iowa State University, Ames, Iowa. Corresponding author: Lie Tang, 2346 Elings Hall, Ames, IA 50011; phone: 515-294-9778; e-mail: lietang@iastate.edu; and Hongwei Xin, 1202 NSRIC, Ames, IA 50011; phone: 515-294-4240; e-mail: hxin@iastate.edu.

Transactions of the ASABE
Vol. 57(5): 1455-1472 © 2014 American Society of Agricultural and Biological Engineers ISSN 2151-0032 DOI 10.13031/trans.57.10505 1455
(2006) adapted and expanded the behavior measurement system and analytical algorithm developed by Persyn et al. (2004) to investigate stocking density effects on feeding behavior of group-housed laying hens.

Behavioral characteristics are usually evaluated using audiovisual tools by a human observer, which is time and labor intensive, subjective to human judgment, and only applicable for a limited observation period (Abrahamsson, 1996). Quantification of animal behaviors, and hence animal welfare, in livestock using image processing brings along specific problems. The appearance of animals varies according to their posture, which makes processing and interpretation of images difficult (Van der Stuyft, 1991). Researchers have used visual monitoring to study group behaviors of animals. Image processing techniques have been used to monitor the weight distribution in poultry flocks (De Wet et al., 2003; Chedad et al., 2003), spatial distribution of pigs (Shao et al., 1998; Hu and Xin, 2000), and trajectory of a flock of poultry (Vaughan et al., 2000). Monitoring behavior of an individual animal within a group requires tracking of the animal. This problem can be alleviated by constraining the animal of interest so that it is in a standard position with no other animals around. This has been applied to pigs to monitor weight (Schofield et al., 1999) and back fat (Frost et al., 2004). Leroy et al. (2006) developed an automatic computer vision technique to track individual laying hen and detect six different behavior phenotypes: standing, sitting, sleeping, grooming, scratching, and pecking. The system involved, however, monitored behaviors of individually caged hen. For freely moving animals, such as a group of laying hens in a cage or a pen, constraints are impractical.

Tracking multiple laying hens for behavior monitoring is a challenging task with interesting features from a computer vision perspective. Segmenting laying hens from the background can be difficult, as the litter floor on which the hens live can often be of similar intensity as that of their feathers. Laying hens tend to flock together, and because laying hens are not highly mobile animals, difficulty in separating individual hens can persist for a prolonged time. Conversely, certain hens may make sudden and quick moves, thereby creating a discontinuous trajectory, which can create difficulties in tracking as well.

The literature on classical multi-target tracking is based on the use of data association after foreground detection in the image. Uchida et al. (2000) proposed a robust method for tracking many pedestrians by viewing from an upper oblique angle. They extracted individuals by background subtraction. When pedestrians overlapped one another, they tracked targets robustly based on their trajectories. However, the movement of poultry in group settings can be rather complex and random.

Computer vision has been applied to tracking animals. Sumpter et al. (1997) tracked a group of ducks at high frame rate. Sergeant et al. (1998) developed a poultry tracking system in which a camera was placed above poultry. They detected poultry silhouettes based on color information and segmented the silhouettes of poultry by using the information on the contours of the silhouette. The identities of the animals between two subsequent images were maintained using a set of simple heuristics. These techniques were further enhanced as model-based tracking, which allows for more robust and accurate shape tracking, including locations on the animal body that are not detectable through image features (Tillett et al., 1997). Fujii et al. (2009) used a computer vision technique based on particle filters to track multiple laying hens. However, like other developed systems, their system was not able to track laying hens for a prolonged period of time, as the particle filters were not able to track the hens when they made sudden quick movements.

The objective of this study was to develop an automated tracking and behavior quantification system for individual hens housed in groups at different stocking densities (SDs). For experimental purpose, the hens were housed in groups of five or ten (SD5 and SD10, respectively) in which each hen was tracked, and each hen’s perching, nesting, feeding/drinking, and movement behaviors were monitored and delineated.

MATERIALS AND METHODS

The developed laying-hen tracking system consisted of hardware and software subsystems (fig. 1). The hardware subsystem consisted of a structural framework of experimental pen, electronic devices (imaging system, RFID components, and communication modules), and a computer. The software subsystem consisted of a data acquisition component and an offline data processing component.

EXPERIMENTAL PEN DESIGN AND SETUP

A 1.2 m × 1.2 m pen was constructed to house multiple laying hens (fig. 2): 0.6 m and 1.2 m long feeders were attached outside the north sidewall when the hens were housed at SD5 and SD10, respectively. A water source (two nipple drinkers) was mounted on the inside of the south sidewall. A 1.2 m × 0.31 m nest box was placed just outside the east sidewall. Entrances (exits) to the nest box were located at the north and south sides. The nest box entrances were 15 cm above the floor. A perch was placed inside the pen 20 cm from the west wall and 25 cm above the floor. Sawdust was used as bedding material for the pen.
An identical pen was made to house hens before moving them into the test (primary) pen for data collection. Fluorescent lighting at an intensity of 10 to 12 lux in the open area and 1 to 2 lux in the nest box was on at 6:00 h and off at 22:00 h, i.e., 16L:8D photoperiod. The resource allowance for the hens in the experiment is shown in Table 1.

The laying hens used in this study were 32-week-old White Leghorns from aviary housing weighing approximately 1.4 kg at procurement. A total of 15 hens were housed in groups of five and ten in two identical pens. First, five birds (SD5) were housed in the primary pen, and ten birds (SD10) were housed in the holding pen. After three days of data collection, five other birds from the holding pen were moved into the primary pen, and data were collected for three more days with ten hens in the primary pen (i.e., a total of 15 hens). The hens were acclimatized for at least five days between data collection. The hens were fed twice a day at 9:00 h and 17:00 h. Eggs were collected once a day at 17:00 h. The litter was cleaned every two weeks during the experiment.

**RFID Antenna Network Design and Interfacing**

A total of 20 antennas (RI-ANT-G02E-30, Texas Instruments, Dallas, Tex.) were used to create an antenna grid, with 18 antennas laid underneath the floor and other two antennas mounted beneath the entrances to the nest box. The 18 antennas on the floor were 30 cm apart on center. Due to their close proximity, the antennas severely interfered with one another. Walls wrapped with aluminum foil were created around each antenna to reduce the interference. However, this foil wrapping significantly reduced the readable range of each antenna, from 28 to 9 cm, which resulted in dead regions between antennas where the antennas did not detect any tags. Figure 3 shows the layout of the 18 antennas installed under the floor. The inner circles represent the readable range of the antennas during operation, and the dead regions are shown in black.

A 4-antenna cluster was created, which was then connected to an RFID reader (RI-STU-251B, Texas Instruments) via a 4-channel multiplexer (RI-MOD-TX8A, Texas Instruments). Five such clusters were created. Figure 3 shows the layout of the clusters, and figure 4 shows the interfacing of the clusters with other devices used in the RFID system. The communication protocol between the 4-channel multiplexers and the RFID readers was RS485. The readers were configured to work in a master/slave synchronization scheme, with the first reader working as the master and all others as the slaves. This configuration allowed the system to read all 20 antennas in less than 0.5 s. With the use of five 4-channel multiplexers, five antennas (one from each 4-antenna cluster) could be read simultaneously.

The RFID readers were connected to serial-to-Ethernet servers (VESR901, B&B Electronics, Ottawa, Ill.) and fi-
nally interfaced to the computer using an off-the-shelf Ethernet hub. Each serial-to-Ethernet server was assigned a unique IP address. The communication protocol between the RFID readers and the serial-to-Ethernet servers was RS485, while TCP/IP was the Ethernet protocol used for interfacing the RFID clusters with the computer. Figure 5 shows the instrumentation used in the RFID network.

**SOFTWARE SUBSYSTEM**

As previously stated, the software subsystem consisted of two components: data acquisition and offline data processing. The data acquisition component included two independently running threads: one for image acquisition and the other for RFID data acquisition.

**Data Acquisition System**

The images were captured for 18 h per day, with 10 h of light time and 8 h of dark time. Images were not captured while feeding the hens and collecting eggs from the pen. The hens were given enough time to settle down before the images were captured. During the capture of each frame, tags read by the RFID sensor network were also recorded. The records were stored in the database and accessed later during the image processing phase to determine hen locations and identities.

Multithreading programming allows a data acquisition system to handle multiple tasks simultaneously, and this technique was implemented in this application to ensure maximum data acquisition speed in reading the multiple RFID antennas. Multithreading programming with uniquely configured device IPs enabled the computer to scan data from the RFID readers in different threads. The RFID readers kept transmitting RFID tag numbers to the TCP/IP socket, and the computer did not have to poll each RFID reader, which essentially enabled the system to operate at a maximum sampling speed. The image acquisition thread acquired images at ~5 frames per second (fps). Each frame was sequentially numbered and stored in the user-specified file path.

The RFID data acquisition threads were run first, and we manually ensured that all the devices were working correctly. The image acquisition thread was then run. The main program thread then created a record for each RFID tag, which consisted of ImagePath (user-specified path where the images were stored), ImageNo (frame number), TagID (RFID tag number), AntennaID (RFID antenna that read the tag), and TimeStamp (time at which the frame was captured). The records were stored in the RFID data table in the database.

**Data Processing System**

The offline data processing component primarily consisted of image processing algorithms. The images were read from the user-specified folder and were processed for hen detection. For each frame, the corresponding RFID data were retrieved from the RFID data table in the database. The centroid of a detected hen was used to locate the closest RFID antenna, which in turn was used to associate the hen with her corresponding RFID tag. For each processed frame, the system created a tracking record that consisted of ImagePath, ImageNo, HenID (1 through 5 for SD5, and 1 through 10 for SD10), TagID, CentroidX (x-coordinate of hen pixel mass), CentroidY (y-coordinate of hen pixel mass), MajorAxisLength (major axis of the ellipse fitted on the hen pixel mass), MinorAxisLength (mi-
nor axis of the ellipse fitted on the hen pixel mass), Heading (heading direction of the hen, 0° to 359°), and TimeStamp. The records were then stored in the Tracking data table in the database.

**Image Processing Algorithm Overview**

The images were subjected to a background subtraction method for foreground detection. The foreground image was filtered using an anisotropic diffusion filter, which essentially helped in enhancing object edges. The filtered image was then segmented using a modified watershed algorithm. Regions in close proximity were merged to form laying hens in the first frame. In subsequent frames, overlaps between the previously identified hen regions and currently segmented watershed regions were used to detect laying hens. As the frames were captured at 5 fps, between-frame movements of the hens were limited; therefore, the algorithm sufficiently tracked the hens.

**Noise Reduction**

In order to alleviate over-segmentation caused by contaminated noise in the watershed transform, we usually have to employ a filter that can effectively reduce noise while preserving important edge information. Although linear filtering can reduce noise in the image, it usually causes blurring and possibly fusing of important edges. Perona and Malik (1990) and Gilboa et al. (2001) reported that diffusion filters were more effective in smoothing noise while preserving necessary edge information. In this study, a diffusion filter (Gilboa et al., 2001) was adopted to reduce the noise effect (fig. 6). The algorithm generalizes the linear and nonlinear scale spaces in a complex domain.
by combining the diffusion equation (eq. 1) with the simplified Schrödinger equation (eq. 2). The imaginary part of the fundamental solution for the linear complex diffusion is regarded as an edge detector (smoothed second derivative), which helps preserve edge information while effectively removing noise (eq. 3):

\[ I_t = c\Delta^2 I, \quad I_t|_{t=0} = I_0, \quad 0 < c \in \mathbb{R} \]  

where \( I_t \) is the noise-free image describing the real scene \( (I = I_R + iI_I) \), \( I_0 \) is the observed or initial image with some degradation due to noise, \( c \) is the complex diffusion coefficient, and \( \Delta \) is the Laplacian operator.

\[ i\hbar \frac{\partial \psi}{\partial t} = -\frac{\hbar^2}{2m} \Delta \psi + V(x) \psi \]  

where \( \psi = \psi(t, x) \) is the wave function of a quantum particle, \( m \) is the mass of the particle, \( \hbar \) is Planck’s constant, \( V(x) \) is the external field potential, and \( i = \sqrt{-1} \).

\[ I_{Rt} = c_R I_{Rx} - c_I I_{Ix}, \quad I_{Rt}|_{t=0} = I_0 \]

\[ I_{It} = c_I I_{Rx} + c_R I_{Ix}, \quad I_{It}|_{t=0} = 0 \]  

where \( c = c_R + ic_I \) is the diffusion coefficient with real component \( c_R = \cos \theta \) and imaginary component \( c_I = \sin \theta \), and \( I_{Rx} \) and \( I_{Ix} \) are second derivatives of \( I_R \) and \( I_I \), respectively.

**Foreground Detection and Gradient Computation**

A background subtraction technique was used to detect foreground objects. An image of the pen was taken without laying hens in it and was subtracted from the image with hens to segment out the foreground. A median filter was used to eliminate smaller regions from the foreground image. Foreground pixels were grouped to form connected components. In the first frame, area threshold was used to decide if a connected component contained one or multiple hens. For subsequent frames, its vicinity was scanned to see if there were other hens around that region in the previous frame. Figure 7 shows the foreground objects detected after background subtraction.

The components that were larger and could be formed from multiple hens were selected for further processing. In the next step, the Sobel gradient operator was used to compute a gradient magnitude image. The operator used two \( 3 \times 3 \) kernels, which were convolved with the original image to calculate approximations of the derivatives in the

---

**Figure 6. Noise reduction:** (a) original distance image, and (b) noise-reduced image after application of diffusion filter.

**Figure 7. Foreground detection:** (a) background image, (b) image with laying hens, and (c) detected foreground objects.
horizontal (eq. 4) and vertical (eq. 5) directions, respectively. The resulting gradient approximations were then combined to compute the gradient magnitude (eq. 6):

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \ast I$$  \hspace{1cm} (4)$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \ast I$$  \hspace{1cm} (5)$$

$$G = \sqrt{G_x^2 + G_y^2}$$  \hspace{1cm} (6)$$

where $G_x$ and $G_y$ are gradient approximations in the horizontal and vertical directions, respectively, $I$ is the original image, and $G$ is the gradient magnitude. The gradient magnitude image was further used for watershed transformation. Figure 8 shows the gradient magnitude computed on selected foreground objects that were then subjected to watershed transformation.

**Foreground Segmentation**

When multiple hens flocked together, it was challenging to separate them before they could be tracked. Edge-based segmentation methods require strong edge information for good segmentation results, which was not always the case when hens came in contact with each other due to the texture of their feathers. Watershed transformation, on the other hand, works well in such situations but is plagued with slower computation and over-segmentation problems.

Watershed transformation is an image processing operation based on mathematical morphology that is analogous to rain falling on a landscape, with each drop flowing down the steepest path toward a body of water called the catchment basin. Classic watershed algorithms are based on successive complete scans of the image. At each step, all the pixels are scanned one after another in a predetermined order, generally with a progressive scan or an interlaced scan. These algorithms do not run in a fixed number of iterations, and the number of iterations is often very large.

On the other hand, the fast watershed algorithm, proposed by Vincent and Soille (1991), is designed such that it does not require scanning the entire image at every iteration. Rather, it allows random access to the pixels of an image and direct access to the neighbors of a given pixel, thereby significantly increasing the efficiency. The fast watershed algorithm is summarized below.

Employing the previously described analogy, when a water drop flows down a relief, it will flow into the region minimum. The Vincent and Soille (1991) watershed segmentation method is based on immersion simulations; starting from the lowest altitude, the water will progressive-
ly fill the different catchment basins of the image. Two steps are involved in the immersion algorithm: sorting and flooding. In the sorting step, the image pixels are sorted in ascending order according to their grayscale values, which enables direct access to the pixels at a certain gray level. The minimum and maximum grayscale values (\(h_{\text{min}}\) and \(h_{\text{max}}\), respectively) are also computed. In the flooding step, the algorithm progressively floods the catchment basins of the image. The algorithm is composed of fast computation of geodesic influence zones and breadth-first scanning of all pixels in the order of altitude (their grayscale values), thereby assigning a distinct label to each minimum and its associated catchment basin. This process is implemented level-by-level using a FIFO (first in, first out) queue of pixels. The output is an image demarcated by the label of the catchment basins. A dam is built to prevent the basins from merging when two floods originating from different catchment basins meet.

Let \(I : D_I \rightarrow \mathbb{N}\) be a grayscale image, with \(h_{\text{min}}\) and \(h_{\text{max}}\) the minimum and maximum gray levels, respectively. Starting at the gray level \(h = h_{\text{min}}\), the catchment basins with the minima of \(I\) are successively expanded up until \(h = h_{\text{max}}\). Let \(X_h\) denote the union of the set of catchment basins computed at level \(h\). A connected component of the threshold set \(T_{h+1}\) at level \(h + 1\) can either be a new minimum or an extension of a catchment basin in \(X_h\). In the latter case, the geodesic influence zone of \(X_h\) within \(T_{h+1}\), \(IZ_{Th+1}\) is computed, resulting in an update \(X_{h+1}\). Let \(\text{MIN}_h\) denote the union of all regional minima at altitude \(h\). The recursive algorithm explained above is defined in equations 7 and 8:

\[
X_{h_{\text{min}}} = \{ p \in D_I \mid I(p) = h_{\text{min}} \} = T_{h_{\text{min}}}
\]  

\[
X_{h+1} = \text{MIN}_{h+1} \cup IZ_{Th+1}(X_h), \quad h \in [h_{\text{min}}, h_{\text{max}}]
\]  

The watershed transform of \(I\), \(W(I)\), is the complement of \(X_{h_{\text{max}}}\) in \(D_I\), i.e., the set of points of \(D_I\) that do not belong to any catchment basin, and is given by equation 9:

\[
W(I) = D_I \setminus X_{h_{\text{max}}}
\]  

According to recursive equations 7 and 8, it is the case that at level \(h + 1\) all non-basin pixels (i.e., all pixels in \(T_{h+1}\) except those in \(X_h\)) are potential candidates to get assigned to a catchment basin in step \(h + 1\). Therefore, the pixels with gray level \(h' \leq h\) that are not yet part of a basin after processing level \(h\) are merged with some basin at the higher level \(h + 1\).

Pixels that, in a given iteration, are equidistant to at least the two nearest basins may provisionally be labeled as “watershed” pixels. However, in the next iteration, this label may change again. A definitive labeling of a pixel as a “watershed” pixel can only happen after all levels have been processed. Figure 9 shows watershed transformation results.
with watershed lines in black and catchment basins in color. The fast watershed algorithm still suffers from an over-segmentation problem. Therefore, the segmented watershed partitions need to be merged to form individual hen regions.

**Vision-Based Tracking**

After watershed transformation of the foreground image, in the very first frame, the regions in close proximity were merged to form laying hen regions. Large partitions were considered as probable hen regions, and merging of such partitions was avoided. Area and orientation information along with mean height were used during the process. In subsequent images, overlaps between the previously identified hen regions and watershed regions were used to merge regions and form individual hens. Because the images were acquired at ~5 fps, the relative movements of the hens in consecutive frames were limited, and the algorithm, in most cases, was able to track individual hens. When the hens made sudden quick movements, it was difficult to associate watershed regions with previously identified hens. In such situations, information from the RFID antenna network was used to recover the identities of lost birds. The RFID network was also used to recover hen identities when multiple hens were in the nest box and one hen exited the nest box. The vision system was unable to maintain hen identities in such situations. The system then maintained a separate data list to restore information for the hens without identities. As soon as the RFID system picked up their tags and their identities were recovered, the corresponding information saved in the data list was merged to the main list that stored the tracking information. Figure 10 shows hens detected and identified in groups of 5 and 10, respectively. Figure 11 shows laying hens identified in different frames in groups of 5 and 10, respectively.

**Hen Identity Recovery Using RFID Antenna Network**

A passive RFID glass transponder (RI-TRP-WEHP-30, Texas Instruments) with a unique number was taped onto the lower part of each hen’s leg (fig. 12). When a hen stood within the readable range of an antenna, the tag number was read and its approximate position was known based on the location of the antenna that read the tag. The RFID antenna network was therefore helpful in locating hens in situations when the visual tracking failed to correctly track them. The RFID antenna network was also used to track hens moving in and out of the nest box. When multiple hens were in the nest box and one of them appeared in the camera view, it was not possible to maintain the identity of the hen until its tag was read by one of the antennas.

**EXPERIMENTS AND RESULTS**

Figure 13 shows the error distribution between the manually located centroids (i.e., manual labeling) and those generated automatically by the tracking algorithm, where a

![Figure 10. Hen detection and identification: (a) segmented hens and (b) hens labeled for tracking. Top row is at SD5; bottom row is at SD10.](image)
total of 600 images from each stocking density were used for the comparison. Frames where the software detected a movement greater than 5 cm were used, which accounted for 95 centroids in the case of SD5 and 176 centroids in the case of SD10. From this distribution, 95% of centroids lie within 4 pixels, i.e., less than 4 cm, of the manually selected centroids. It was also noted that the manual selection of centroids was expected to exhibit an error of ±3 pixels.

Figures 14 and 15 show comparisons of manually extracted trajectories and software-generated trajectories for laying hens housed in groups of 5 and 10, respectively. In the case of SD10, one of the hens was in the nest box throughout the image sequence. The filled circles represent the positions of hens in the first frames, while the filled diamonds represent the positions in the last frames. It can be seen in these figures that the manual and software-generated trajectories closely resemble each other.

Visual tracking of the laying hens seemed to work by simply using frame-to-frame correspondence based on the overlap of pixels between consecutive frames, given that the hens were correctly identified in the first frame. There were situations when visual tracking was unable to maintain the identities of the hens. One such case is shown in figure 16, in which one of the hens exited the nest box and appeared on the pen floor. In this scenario, the visual tracking system did not know which hen, among those in the nest box, had exited, although it still kept the track of the hen without identifying which hen it was. When the RFID system detected the tag attached to the hen, the visual tracking system recovered the hen’s identity. The tracking data associated with the hen, between the frame when the system lost the hen’s identity and the frame when the RFID
system recovered the hen’s identity, was temporarily stored in a separate table in the database. When the hen’s identity was recovered, this series of data was moved into the main data table where the tracking data for all the hens were correctly associated with their identities.

In another case, when certain hens made sudden quick movements, the visual system failed to keep track of the hens by simply using frame-to-frame correspondence based on overlapped pixels. In the scenario shown in figure 17, hen 6 appeared to make a sudden quick movement between two consecutive frames. When frame-to-frame correspondence was used, hen 8 was misidentified as hen 6. In this case, the RFID system was once again used to recover the identity of the hens, as it was able to read the tags attached to the hens at their correct positions.

The system was able to track individual hens and extract their behaviors, such as perching, nesting, feeding, drinking, and movement. The SD effect was examined by comparing behavioral data for the same 5 hens used at both SD levels. Figure 18 shows the time spent by the hens in the feeding area on different days. The graph clearly indicates that the hens (B1 to B5) spent more time in the feeding area when housed in a group of 5 than when housed in a group of 10.

Figures 19, 20, 21, and 22 show the time budgets of the hens’ perching, nesting, feeding, and drinking behaviors, respectively. The shaded block along the horizontal axis indicates the dark hours of the day. Data collection started at 10:00 h and continued until 16:00 h. Between 16:00 h and 20:00 h, the feed trough was refilled, the water supply was checked, and eggs were collected from the nest box and also from the pen floor, if any. The next round of data collection started at 20:00 h until 8:00 h the next morning. The feed trough was refilled and the water supply was checked before data collection started again at 10:00 h.

Figure 19 clearly shows that the hens spent more time...
Figure 15. Manually extracted vs. software-generated trajectories of laying hens at SD10.

Figure 16. Recovering hen identity when a hen exited the nest box: (a) in frame 401, hen 6 was about to enter the nest box; (b) in frame 402, hen 6 entered the nest box and was outside camera view; (c) in frame 475, a hen whose identity was unknown exited the nest box; and (d) in frame 484, the hen was identified as hen 9.
on the perch at night than during the day. The data also show that the hens spent 348 ±240 min hen\(^{-1}\) d\(^{-1}\) and 265 ±158 min hen\(^{-1}\) d\(^{-1}\) on the perch when housed at SD5 and SD10, respectively, presumably due to the available perch space. Similarly, figure 20 shows time budget of nesting behavior. The hens spent more time in the nest box between 10:00 h and 11:00 h, and the time spent in the nest box slowly declined. It was observed that only 3 or 4 hens spent most of their time on the perch at night, while some hens spent the entire night in the nest box or on the floor despite having enough perch space. The data revealed that the hens spent 99 ±165 min hen\(^{-1}\) d\(^{-1}\) and 78 ±142 min hen\(^{-1}\) d\(^{-1}\) in the nest box when housed at SD5 and SD10, respectively.

Figure 21 shows the time budget of feeding behavior. The feeding behavior seems consistent throughout the day, with nearly zero activity at night. It can be seen that the hens spent 87 ±21 min hen\(^{-1}\) d\(^{-1}\) and 60 ±17 min hen\(^{-1}\) d\(^{-1}\) in the feeding area when housed at SD5 and SD10, respectively. It should be noted that the time spent at the feeder, as determined by the data, does not necessarily represent the amount time of feeding. Similarly, as shown in figure 22, drinking behavior seems consistent throughout the day and was nearly zero at night. The hens spent 32 ±12 min hen\(^{-1}\) d\(^{-1}\) and 27 ±11 min hen\(^{-1}\) d\(^{-1}\) in the drinking area when housed at SD5 and SD10, respectively. Figure 23 shows the time budget of movement. The hens’ movement averaged 499 ±236 m d\(^{-1}\) and 540 ±160 m d\(^{-1}\) when housed at SD5 and SD10, respectively.

Figure 24 shows a comparison between the distributions of movement by the hens housed at SD5 and SD10 filtered at 5 cm to ignore smaller movements, which could be the result of erroneous centroid extraction or body stretching (hence centroid shifting) without real locomotion. The total idle time and/or movements smaller than 5 cm accounted for about 96% and 97% at SD5 and SD10, respectively. About 92% of the movements during the day (10 h d\(^{-1}\)) were between 5 and 10 cm long when the hens were housed at SD5, and about 89% were between 5 and 10 cm long when the hens were housed at SD10.
housed at SD10. The average travel speed was found to be 0.38 and 0.45 m s⁻¹ at SD5 and SD10, respectively.

Figure 25 shows the average time spent by the hens performing different activities. The same 5 hens on average spent 32% and 25% of their time on the perch when housed in groups of 5 and 10, respectively. The difference presumably arose from the difference in available perch space. Similarly, the hens on average spent 9% and 7% of their time in the nest box, and 8% and 6% of their time in the feeding area when housed in groups of 5 and 10, respectively. The hens spent 3% of their time in the drinking area at both stocking densities. For the rest of the time (48% and 60%), the hens performed activities such as standing, walking, and sitting when housed at SD5 and SD10, respectively.
Multi-regression statistical analysis with t-test using R on the data showed that the SD effect was significant on the perching behavior of the laying hens (p = 0.0023). The hens spent more time on the perch at SD5 (348 min) than at SD10 (265 min). This is not surprising because of the limited perch space. Similarly, the SD effect was prominent on time spent by the feeder (p < 0.0001): 87 min at SD5 and 60 min at SD10. On the other hand, the SD effect was insignificant on nesting or drinking behaviors (p = 0.3597 and 0.1366, respectively). The result also show that SD did not affect movement of the hens for the given floor space of 1.2 m × 1.2 m (p = 0.2422). It should be pointed out that it was not the purpose of this study to examine effects of
resource allocation on hen behaviors or time budget. Rather, the two stocking densities were used to test the functionality of the automated tracking and behavior qualification system. Follow-up studies will be conducted to delineate such effects.

CONCLUSIONS

A sensor fusion approach to tracking laying hens housed in groups of 5 and 10 has been developed and tested. Due to the varying nature of their appearance related to their posture, and their social behavior of performing activities in groups, detecting individual laying hens housed in groups was a challenging task. However, the image processing techniques developed based on the depth images was able to satisfactorily detect and identify individual hens with occasional help from the developed RFID system, when necessary. The developed tracking system was used to automatically extract behaviors, such as locomotion, perching, nesting, feeding, and drinking, of hens housed in groups of 5 and 10, thereby quantifying the effects of resource allocation (e.g., stocking density) on their behaviors. The system has been demonstrated to be capable of tracking and maintaining identities of individual hens, which is
critical for extraction of time budgets of individual hen behaviors. This unique tracking system will enhance researchers’ ability to examine the impact of physical and management factors on the behaviors and well-being of group-housed animals.

ACKNOWLEDGEMENTS

This project was supported in part by the USDA-NIFA Agriculture and Food Research Initiative Program (Grant Award 2011-67021-20223).

REFERENCES


