

Research Article

Open Access



An informative planning-based multi-layer robot navigation system as applied in a poultry barn

Tingjun Lei¹, Guoming Li^{2,3}, Chaomin Luo¹, Li Zhang⁴, Lantao Liu⁵, Richard Stephen Gates^{2,3,6}

¹Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, MS 39762, USA.

²Department of Agricultural and Biosystems Engineering, Iowa State University, Ames, IA 50011, USA.

³Department of Animal Science, Iowa State University, Ames, IA 50011, USA.

⁴Department of Poultry Science, Mississippi State University, Mississippi State, MS 39762, USA.

⁵Department of Intelligent Systems Engineering, Indiana University, Bloomington, IN 47408, USA.

⁶Egg Industry Center, Iowa State University, Ames, IA 50011, USA.

Correspondence to: Prof. Chaomin Luo, Department of Electrical and Computer Engineering, Mississippi State University, 406 Hardy Road, Mississippi State, MS 39762, USA. E-mail: Chaomin.Luo@ece.msstate.edu; ORCID: 0000-0002-7578-3631

How to cite this article: Lei T, Li G, Luo C, Zhang L, Liu L, Gates RS. An informative planning-based multi-layer robot navigation system as applied in a poultry barn. *Intell Robot* 2022;2(4):313-32. <http://dx.doi.org/10.20517/ir.2022.18>.

Received: 10 Jun 2022 **First Decision:** 29 Jul 2022 **Revised:** 13 Aug 2022 **Accepted:** 25 Aug 2022 **Published:** 12 Oct 2022

Academic Editor: Simon X. Yang **Copy Editor:** Jia-Xin Zhang **Production Editor:** Jia-Xin Zhang

Abstract

Many real-world robot applications, as found in precision agriculture, poultry farms, disaster response, and environment monitoring, require search, locate, and removal (SLR) operations by autonomous mobile robots. In such application settings, the robots initially search and explore the entire workspace to find the targets, so that the subsequent robots conveniently move directly to the targets to fulfill the task. A multi-layer robot navigation system is necessary for SLR operations. The scenario of interest is the removal of broiler mortality by autonomous robots in poultry barns in this paper. Daily manual collection of broiler mortality is time- and labor-consuming, and an autonomous robotic system can solve this issue effectively. In this paper, a multi-layer navigation system is developed to detect and remove broiler mortality with two robots. One robot is assigned to search a large-scale workspace in a coverage mode and find and locate objects, whereas the second robot directly moves to the located targets to remove the objects. Directed coverage path planning (DCPP) fused with an informative planning protocol (IPP) is proposed to efficiently search the entire workspace. IPP is proposed for coverage directions in DCPP devoted to rapidly achieving spatial coverage with the least estimation uncertainty in the decomposed grids. The detection robot consists of a developed informative-based directed coverage path planner and a You Only Look Once (YOLO) V4-based dead bird detector. It refines and optimizes the coverage path based on historical data on broiler mortality distribution in a broiler barn. The removal robot collects dead broilers driven by a new hub-based multi-target path routing (HMTR) scheme, which is applicable to row-based environments. The proposed methods show great potential to navigate in broiler barns ef-



© The Author(s) 2022. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



efficiently and safely, thus being a useful component for robotics. The effectiveness and robustness of the proposed methods are validated through simulation and comparison studies.

Keywords: Directed coverage path planning, informative planning protocol (IPP), broiler mortality, YOLO V4, hub-based multi-target routing (HMTR) scheme

1. INTRODUCTION

1.1. Background

Global broiler production has been growing since the 1960s; in the United States alone, over 9.22 billion broilers were produced in 2020 at a value of 21.7 billion dollars^[1]. A modern broiler barn holds 25,000 to 100,000 broilers on the open litter floor. Within the intensive production system, broiler mortality could be largely due to disease and metabolic problems and unsuitable environmental conditions and management^[2]. The mortality rate is commonly 5% in an eight-week production cycle^[1] and can reach 10% without appropriate management^[3]. As a daily task, producers need to collect and remove broiler mortality timely, which is an arduous, time-consuming, and unpleasant task. A farmer spends hours daily identifying, gathering, and transporting dead birds in a typical broiler barn (e.g., 25,000 birds/barn). Manual mortality collection is relatively simple when broilers are young because they are lightweight and small. However, older broilers are bigger and weigh more, and, to avoid overwork, farmers may deposit the mortality in scattered piles within the broiler barn for subsequent removal^[3]. A dead bird residing on a litter floor can result in higher levels of bacterial accumulation and increase the risk of disease spread via direct contact or vector transmission^[3]. Additionally, caretakers risk their health if exposed to harsh working environments (e.g., concentrated ammonia and dust) for a long term^[2]. Therefore, autonomous robotic systems for collecting broiler mortality are warranted to improve labor efficiency and reduce biosecurity issues.

Robotic systems have been developed to facilitate poultry production^[4,5]. Vroegindeweij *et al.* developed a PoultryBot to avoid obstacles and pick up floor eggs in cage-free hen housing systems^[6]. Li *et al.* developed an egg picking robot based on a deep learning detector and a robot arm^[7]. Some poultry robots are commercialized to help farmers monitor environments, inspect bird health and welfare status, stimulate bird movement, aerate litter, and disinfect poultry barns. However, robots for poultry mortality removal are still not commercially available. Many aspects need to be considered for developing such a robot^[8]. Among them, robot path planning is one of the most essential parts of effective and efficient determination of robot routes^[9,10].

1.2. Related work

Many approaches have been proposed to achieve reliable autonomous robot motion planning, such as ant colony optimization (ACO)^[11-13], fireworks algorithm (FWA)^[14], bat-pigeon algorithm (BPA)^[15], graph-based method^[16], and neural network models^[17-19]. Lei *et al.* proposed a hybrid model to optimize the trajectory of the global path using a graph-based search algorithm associated with an ant colony optimization (ACO) method^[11]. A hybrid fireworks algorithm with LIDAR-based local navigation capable of generating short collision-free trajectories in unstructured environments was developed^[14]. The cuckoo search algorithm has also been successfully applied to the efficient and safe navigation of robots^[20]. A bat-pigeon algorithm^[15] was developed with crack detection-driven autonomous vehicle navigation and mapping, in which a local search-based bat algorithm and a global search-based pigeon-inspired optimization algorithm are effectively integrated to improve the speed and performance of robot path planning and mapping. Wang and Meng^[16] suggested a nonuniform sampling technique, which efficiently computes high-quality collision-free paths based on a generalized Voronoi graph. A biologically motivated neural network model using a shunting equation was proposed by Yang and Luo^[17] for real-time path planning with obstacle avoidance. Luo *et al.* extended the model of trajectory planning with safety consideration in conjunction with the virtual obstacle

algorithm^[21]. However, the design and implementation of path planning in a broiler barn involve multiple aspects for mortality removal. Especially with the large scale of modern broiler barns (e.g. 12 m × 200 m), poultry mortality is widely distributed anywhere in the barn, which increases the difficulty of detecting and removing dead broilers. Therefore, we propose a system for searching and removing broiler mortality with two robots. One is a broiler mortality detection robot and the other is a broiler mortality removal robot. The major function of the detection robot is to precisely and efficiently search the whole broiler barn and indicate the position of dead broilers. Thus, coverage path planning (CPP) algorithms need to be introduced to assist robots in search and exploration tasks, such as bio-inspired neural networks^[22], Boustrophedon grid decomposition^[23], deep learning^[24,25], *etc.*

Luo and Yang^[22] developed the bio-inspired neural network (BNN) method to navigate robots to perform complete coverage path planning (CCPP) while avoiding obstacles within dynamic environments. The robot is attracted to unscanned areas and repelled by the previous scanned areas or obstacles based on the neural activity through the BNN model. The next position of the robot depends on the current position of the robot and neuron activity associated with its current position. Zhu *et al.* proposed a complete coverage path planning model using Glasius BNN, which is extended to multiple robots to lower the overall search time and improve efficiency^[19]. Unlike the BNN approach, the boundary representation model that defines the workspace is adopted by the Boustrophedon cellular decomposition (BCD) method and deep reinforcement learning approach (DRL). The BCD method proposed by Acar and Choset^[23] decomposes the environment into many line scan partitions explored through a back-and-forth path (BFP) in the same direction. BCD is an effective CCPP method for more diverse, non-polygonal obstacles in a workspace. In trapezoidal decomposition as a cell, it is covered in back-and-forth patterns. For a complex configuration space with irregular-shaped obstacles, BCD needs to construct a graph that represents the adjacency connections of the cells in the Boustrophedon decomposition. Similarly, Nasirian *et al.* utilized traditional graph theory to segment the workspace and proposed a deep reinforcement learning approach to solve the CCPP problem in a complex workspace^[24]. Lei *et al.* proposed a deep learning method to detect the workspace and generate turning waypoints for the robot to complete the coverage of the entire workspace^[25]. The trajectory generated by the above-presented algorithms completely covers the workspace in light of the size of the robot, which is more suitable for a large-sized workspace with a broad sensing range. However, with the limited sensing range of robots in broiler barns, a complete coverage trajectory requires a large amount of time and energy, which may not be affordable to the farmers.

The methods of object detection have been extensively studied, and many object detectors are utilized in many agricultural settings. Li *et al.* proposed a fast region-based CNN (R-CNN) detector with high accuracy to adapt to the illumination of different colored lights in the farm and detect the cage-free floor eggs^[26]. A mask R-CNN object detector was developed by Li *et al.* to detect hen preening behaviors in barns to automatically monitor poultry behaviors, judge the comfort level of hens, and assist welfare-oriented poultry management^[27]. Bochkovskiy *et al.* proposed a YOLO V4 object detector, which is a deep learning-based object detection technique with high accuracy^[28]. It is an effective object detector to localize objects in real-time processing.

1.3. Proposed methods and original contribution

A directed coverage path planning (DCPP) fused with an informative planning protocol (IPP) is proposed to efficiently search the entire barn. The broiler barn is first decomposed into large grids based on the workspace dimensions to obtain overall global trajectories with coverage directions. IPP continues to rapidly achieve spatial coverage with the least estimation uncertainty in the decomposed grids. Flexible and efficient trajectories are formed by utilizing historical information of broiler mortality spatial distribution^[29] and the direction information from the previous steps. With the assistance of current state-of-the-art computer vision algorithms in precision agriculture, such as deep learning techniques^[30], the robot could detect dead birds in the grids of vision range and find the mortality location for the removal robot. Such a comprehensive path planning

method for the localization robot could pinpoint targets and avoid unnecessary searching trajectories, saving the robot time and energy.

After the locations of dead broilers are precisely and efficiently identified, the removal robot is deployed to collect the mortalities. To save time and energy, the total length of the path to remove multiple dead broilers in succession should be minimized^[31]. In that regard, a new hub-based multi-target routing (HMTR) scheme is developed in this paper, which originates from the row-based environment routing of broiler barn. By introducing hub grids, computational efficiency is improved for distance calculations between targets. Based on the locations of the dead birds, the best visiting sequence is determined while planning collision-free trajectories simultaneously. Furthermore, a reactive local navigator is developed to dynamically update the path and map in real time to avoid moving obstacles and unforeseen obstacles in the dynamic environment. This is of great help to robot safety and obstacle avoidance and prolongs the life of the robot.

Overall, we propose a system for detecting and removing broiler mortality with two robots. The detection robot consists of an informative directed coverage path planner and a You Only Look Once (YOLO) V4-based dead bird detector. The removal robot collects dead broilers by the hub-based multi-target path routing (HMTR) scheme. Meanwhile, with the reactive local navigator, the robots avoid obstacles and reach the target position through the real-time updated information of the onboard LIDAR sensors.

The major contributions of this paper are summarized as follows:

- A multi-layer robot navigation system for search, locate, and remove (SLR) operations is developed for the removal of dead broilers in a poultry barn to reduce the daily laborious and time-consuming work.
- A directed coverage path planning (DCPP) method integrated with an informative planning protocol (IPP) is developed to efficiently search targets in a large-scale workspace, which is initially decomposed into grids, and the optimal coverage directions are obtained based on the workspace. The detection robot adaptively explores the workspace to generate the trajectory in the grids in light of the historical data of searched targets and the coverage directions.
- The informative planning protocol (IPP) is developed to integrate coverage directions devoted to rapidly achieving *spatial* coverage with the least estimation uncertainty in the decomposed grids.
- To efficiently reach the targets, a new hub-based multi-target path routing (HMTR) scheme is proposed to row-based environments. The total visiting distance is minimized through obtained targets to generate an optimal sequence to connect all the targets.
- Practically, to apply this navigation system in the poultry barn, multiple robots with this developed navigation system are utilized, consisting of a dead broiler search robot to search and locate the dead broilers and a dead broiler removal robot to collect the dead broiler afterwards.

2. PRELIMINARY DEFINITION

2.1. Workspace setup and robot configuration

The complex barn environment is simplified into a rectangular robot workspace (152 m long × 12 m wide) based on typical United States broiler barn dimensions^[32]. The workspace where the robot needs to collect dead broilers consists of multiple feeding and drinking lines that run the length of the barn, start and end positions of the robot, and dead and live broilers that are randomly located in the barn. Two feeding and three drinking lines are arranged in parallel in the middle, each measuring 140 m long and 0.3 m wide [Figure 1]. The distance between lines is 1.75 m, and endpoints of the lines are 6 m away from the ends of the barn. The start and end positions are located near the gate of the barn and could be altered in a real-world application. Live/dead birds are simplified as 20 cm diameter circles. With a stocking density of nearly 929 cm²/bird (1 ft²/bird), the barn is assumed to have 20,000 birds. The workspace is illustrated in Figure 1.

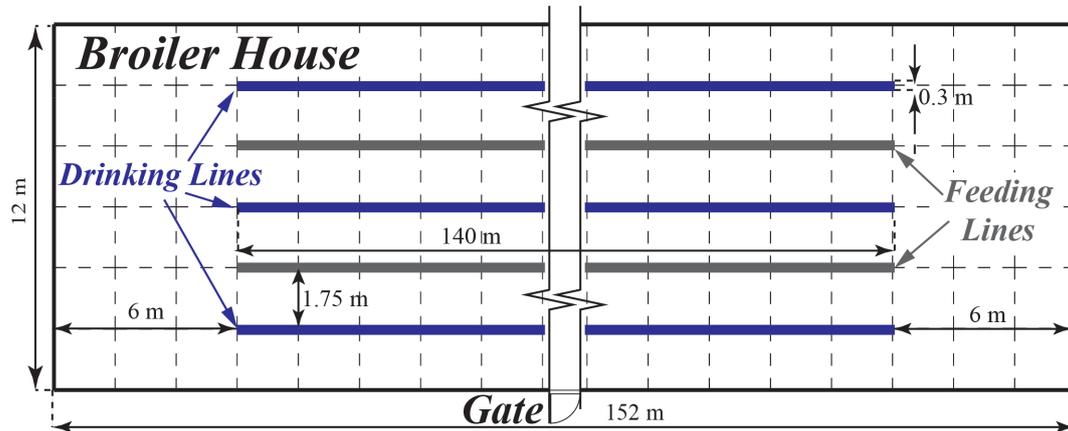


Figure 1. Illustration of simulated workspace based on a broiler barn. Blue lines are drinking lines, while grey lines are feeding lines. In total, 20,000 broilers are randomly distributed in the workspace.

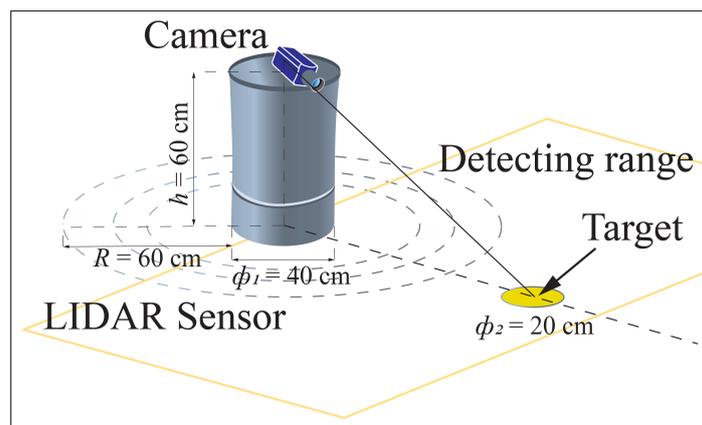


Figure 2. Illustration of the cylindrical robot with a camera and LIDAR sensor. Cylinder represents robot, dot circles represent LIDAR sensor scans, while the yellow circle is a target (dead broiler).

The tested robots in this paper are assumed to be cylinders, each 20 cm in radius and 60 cm in height. The payload based on bird volume (10 cm in height and 11–30 cm in diameter^[33]) is estimated as 30 dead birds. It is equipped with a camera to capture bird images in real time. The entire workspace is decomposed into non-overlapping grids, each measuring 200 cm long \times 200 cm wide [Figure 1]. A LIDAR sensor is mounted into the robot to detect static (i.e., feeding and drinking lines) and dynamic (i.e., live broilers) obstacles. The effective maximum detection range is limited by complex environments to a radius with approximately 60 cm. The basic function of the robot is to search for dead and live broilers during their moving, arrive at the mortality, and avoid moving birds. The parameters of the robot are set as follows: maximum linear velocity of 1 m/s, maximum angular velocity of 20 rad/s, maximum acceleration/deceleration of 0.2 m/s², and maximum angular acceleration/deceleration of 50 rad/s². These parameters are suited to industrial levels. The proposed autonomous robot is illustrated in Figure 2.

2.2. Overall workflow for robot path planning

The overall procedure of the multi-layer robot navigation system is shown in Figure 3. The workspace is decomposed into a grid-based working map and fed into the robots. The whole robotic framework consists of the detection robot and removal robot, where the detection robot is run at first followed by the removal robot. The detection robot considers the map information and historical mortality distribution while it outputs a

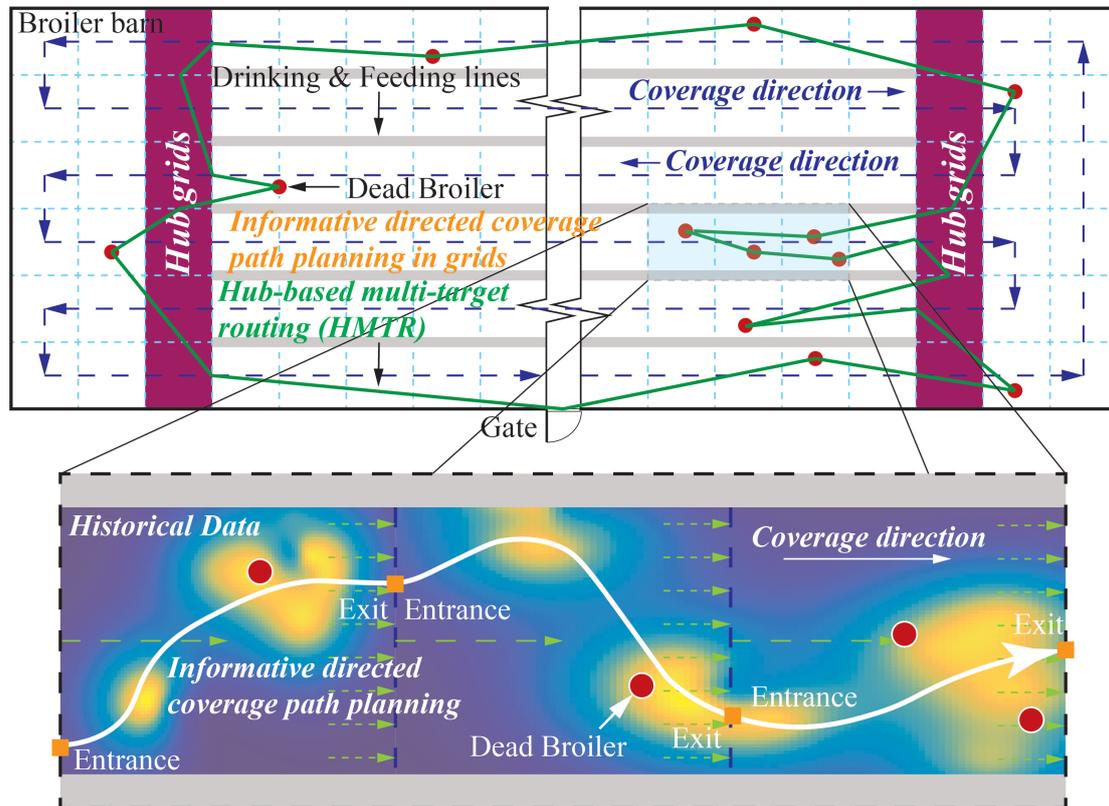


Figure 3. Overall procedure of the multi-layer. The dashed blue lines represent the DCP with the coverage directions. The informative planning protocol integrates coverage directions and continues to rapidly achieve spatial coverage with the least estimation uncertainty in the decomposed grids. Red circles are the dead broilers found by YOLO V4-based dead broiler detector in the boiler barn. The maroon grids are the hubs in the row-based environment. The green lines represent the hub-based multi-target paths. Note that the heat map in this figure only represents the distribution of the historical data of dead chickens, which is different from the information heat maps with generated coverage direction information in the rest of the paper. DCP: Defined contribution pension plan.

mortality-based coverage trajectory. It mainly consists of two parts: (1) informative directed coverage path planning; and (2) target detection. The removal robot determines the final trajectory in real time by utilizing the coordinates of dead birds throughout a broiler barn. It consists of an HMTR scheme.

3. METHOD

3.1. The overall workflow of the proposed work

Details of the corresponding algorithms are described in the following sections. The overall workflow of the path planning for the broiler mortality detection and removal robots is illustrated in Figure 4.

The proposed multi-layer robot navigation system illustrated in Figure 4 consists of informative directed coverage path planning (directed coverage path planning and informative planning protocol), the hub-based multi-target routing (HMTR) scheme, target detection, and YOLO V4. In this navigation system, the initial robot is tasked with finding and locating objects in a large-scale setting, while the second robot proceeds directly to the targets it has discovered to remove the objects. To effectively search the full vast workspace, a directed coverage path planning (DCPP) coupled with an informative planning protocol (IPP) is proposed in the system, as shown in Figure 4. To quickly achieve spatial coverage with the least amount of estimation error in the deconstructed grids, IPP is recommended for coverage directions in DCPP. The detection robot is driven by the You Only Look Once (YOLO) V4-based dead bird detector and an informative-based directed coverage path planner. As applied in the poultry barn, two robots are utilized in this multi-layer navigation system to

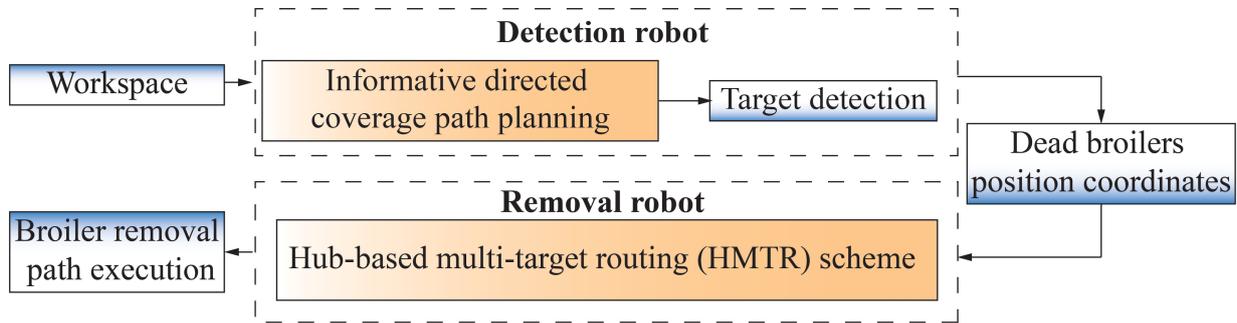


Figure 4. Overall workflow of the path planning for the broiler mortality detection and removal robots. The dynamic window approach (DWA) local navigator is also utilized to avoid live broilers in simulations.

find and eliminate broiler mortality.

3.2. Directed coverage path planning

The directed coverage path planning (DCPP) is utilized to generate an overall trajectory of the broiler mortality detection robot. It should meet the following requirements:

1. The mobile robot must move through all regions of interest in the barn.
2. Motion trajectories should be simplified as straight lines.
3. The robot should cover the regions with minimal overlapping.
4. The path should be optimized with one of the criteria, such as distance, time, and energy.
5. The start and end points for the robots should be co-located so that farmers do not need to enter the barn and retrieve the stopped robot from the barn interior.

In this paper, the criterion to generate a directed coverage path is the total shortest path, which could save running energy for the robots. In the decomposed working map, the adjacent grids share at least one boundary, while nodes of the adjacent grids are connected to form overall trajectories. There are two commonly used methods for CPP: zigzag motion planning and spiral motion planning, namely back-and-forth and boundary sweep methods^[17].

The robot moves along the lengthwise direction of the workspace under the zigzag motion planning, and the total path length is calculated by Equation (1):

$$\mathcal{L}_z = (\mathcal{L}_l - \mathcal{L}_g) \times \left(\frac{\mathcal{L}_s}{\mathcal{L}_g} \right) + \mathcal{L}_s - \mathcal{L}_g \tag{1}$$

where \mathcal{L}_z is the total length of zigzag motion planning; \mathcal{L}_l is the length of the long side of a workspace; \mathcal{L}_s is the length of the short side of a workspace; and \mathcal{L}_g is the length of a grid. The robot under the zigzag motion planning moves along the long side of the workspace $\mathcal{L}_s/\mathcal{L}_g$ times and along the short side $\mathcal{L}_s/\mathcal{L}_g - 1$ times. It turns $2(\mathcal{L}_s/\mathcal{L}_g - 1)$ times.

Under the spiral motion planning, the robot moves along the boundary of the workspace progressively and at the same time toward the center spirally. The total travel distance (TD) on the short side \mathcal{D}_s is

$$\mathcal{D}_s = \frac{\mathcal{L}_s \times (\mathcal{L}_s - \mathcal{L}_g)}{2 \times \mathcal{L}_g} \tag{2}$$

The total travel distance on the long side \mathcal{D}_l is

$$\mathcal{D}_l = \frac{2\mathcal{L}_l\mathcal{L}_s - (\mathcal{L}_s)^2 + \mathcal{L}_g\mathcal{L}_s - 2(\mathcal{L}_g)^2}{2\mathcal{L}_g} \tag{3}$$

$$\mathcal{L}_p = \mathcal{D}_s + \mathcal{D}_l = \frac{2\mathcal{L}_l\mathcal{L}_s - 2(\mathcal{L}_g)^2}{2\mathcal{L}_g} \tag{4}$$

where \mathcal{L}_p is the total length of spiral motion planning. However, the start and end points of the robot coverage path are different based on the two traditional methods, which cannot meet Requirement 5. The total path length should include the returning path towards the start point, which could be suboptimal under the two planning methods.

Lemma 1. *The length of returning path under zigzag planning in the workspace (\mathcal{L}_{zR}) is constrained as*

$$\mathcal{L}_s - \mathcal{L}_g \leq \mathcal{L}_{zR} \leq \sqrt{(\mathcal{L}_l - \mathcal{L}_g)^2 + (\mathcal{L}_s - \mathcal{L}_g)^2} \tag{5}$$

Proof. Since $\mathcal{L}_l \geq \mathcal{L}_s > \mathcal{L}_g$, we have $\mathcal{L}_s - \mathcal{L}_g < \sqrt{(\mathcal{L}_l - \mathcal{L}_g)^2 + (\mathcal{L}_s - \mathcal{L}_g)^2}$.

The equality holds on the left of Equation (5) when $\mathcal{L}_s = 2 \times n \times \mathcal{L}_g$ for $n = 1, 2, 3, \dots$. Then, the $\mathcal{L}_{zR} = \mathcal{L}_s - \mathcal{L}_g$.

The equality is satisfied on the right of Equation (5) when $\mathcal{L}_s = (2 \times m - 1) \times \mathcal{L}_g$ for $m = 2, 3, 4, \dots$. Then, the \mathcal{L}_{zR} is

$$\mathcal{L}_{zR} = \sqrt{(\mathcal{L}_l - \mathcal{L}_g)^2 + (\mathcal{L}_s - \mathcal{L}_g)^2} \tag{6}$$

This completes the proof of Lemma 1. □

Lemma 2. *The length of returning path under the spiral planning in the workspace (\mathcal{L}_{pR}) is given*

$$\sqrt{\left(\frac{\mathcal{L}_s}{2}\right)^2 + \left(\frac{\mathcal{L}_s - 2\mathcal{L}_g}{2}\right)^2} \leq \mathcal{L}_{pR} \leq \sqrt{\left(\frac{\mathcal{L}_s - \mathcal{L}_g}{2}\right)^2 + \left(\frac{2\mathcal{L}_l - \mathcal{L}_s + \mathcal{L}_g}{2}\right)^2} \tag{7}$$

Proof. Since $\mathcal{L}_l \geq \mathcal{L}_s > \mathcal{L}_g$, $\left(\frac{\mathcal{L}_s - 2\mathcal{L}_g}{2}\right)^2 < \left(\frac{\mathcal{L}_s - \mathcal{L}_g}{2}\right)^2$ and $\left(\frac{\mathcal{L}_s}{2}\right)^2 < \left(\frac{2\mathcal{L}_l - \mathcal{L}_s + \mathcal{L}_g}{2}\right)^2$, we have $\sqrt{\left(\frac{\mathcal{L}_s}{2}\right)^2 + \left(\frac{\mathcal{L}_s - 2\mathcal{L}_g}{2}\right)^2} < \sqrt{\left(\frac{\mathcal{L}_s - \mathcal{L}_g}{2}\right)^2 + \left(\frac{2\mathcal{L}_l - \mathcal{L}_s + \mathcal{L}_g}{2}\right)^2}$.

The equality holds on the left of Equation (7) when $\mathcal{L}_s = 2 \times n \times \mathcal{L}_g$ for $n = 1, 2, 3, \dots$. Then, the \mathcal{L}_{pR} is

$$\mathcal{L}_{pR} = \sqrt{\left(\frac{\mathcal{L}_s}{2}\right)^2 + \left(\frac{\mathcal{L}_s - 2\mathcal{L}_g}{2}\right)^2} \tag{8}$$

The equality holds on the right of Equation (7) when $\mathcal{L}_s = (2 \times m - 1) \times \mathcal{L}_g$ for $m = 2, 3, 4, \dots$

Then, the \mathcal{L}_{pR} is

$$\mathcal{L}_{pR} = \sqrt{\left(\frac{\mathcal{L}_s - \mathcal{L}_g}{2}\right)^2 + \left(\frac{2\mathcal{L}_l - \mathcal{L}_s + \mathcal{L}_g}{2}\right)^2} \tag{9}$$

This completes the proof of Lemma 2. □

The robot follows the first two steps of spiral motion planning, then repeats the zigzag motion planning, and finally returns to the start point. The robot is assumed to run along the feeding and drinking lines without moving across them. The DCPM finally provides the robot with overall trajectories from the start to end points.

The total path length of DCPM (\mathcal{L}_M) is calculated through Equation (10):

$$\mathcal{L}_M = 2 \times (\mathcal{L}_l + \mathcal{L}_s - 2\mathcal{L}_g) + (\mathcal{L}_l - 2\mathcal{L}_g) \times \frac{\mathcal{L}_s - 2\mathcal{L}_g}{\mathcal{L}_g} \tag{10}$$

The procedure of the proposed DCPD in a broiler barn is described in Algorithm 1. It is worth noting that the DCPD does not provide the robot with an actual coverage path, which could vary based on the IPP.

Algorithm 1: Directed coverage path planning procedure in a broiler barn

```

Initialize the left turn times  $\tau = 0$ ;
Initialize the next step state as null,  $\mathcal{S} = []$ ;
Initialize the robot heading to the longer side of the barn;
while (the robot does not reach target position) do
    if ( $\tau < 2$ ) then
        if (no obstacle on the right) then
            | Turn right and move forward;
        else if (no obstacle in the front) then
            | Move forward;
        else if (no obstacle on the left) then
            | Turn left and move forward;
            |  $\tau = \tau + 1$ ;
        end
    end
    if ( $\tau \geq 3$  and the robot can move forward) then
        if (no obstacle in the front) then
            | Set  $\mathcal{S} = [\text{Move forward}]$ ;
        else if (no obstacle on the left) then
            | Set  $\mathcal{S} = [\text{Turn left and move forward}]$ ;
            | Turn left;
        else if (no obstacle on the right) then
            | Set  $\mathcal{S} = [\text{Turn right and move forward}]$ ;
            | Turn right;
        end
        Perform  $\mathcal{S}$ ;
    end
end
end

```

3.3. Informative planning protocol

The informative planning protocol (IPP) further refines and optimizes the coverage path based on historical data on broiler mortality distribution in a broiler barn. In the general case, the IPP incorporates the multi-objective informative planning and uses DCPD and historical data to guide the robots to more frequently visit those places with a higher probability of broiler mortality appearance. This multi-objective optimization problem is stated in Equation (11).

$$\begin{aligned}
 \mathbf{m}^* &= \underset{\mathbf{m} \in \mathcal{M}}{\operatorname{argmax}} \{ \mathcal{D}(\mathbf{m}), \mathcal{J}_1(\mathbf{m}), \mathcal{J}_2(\mathbf{m}), \dots, \mathcal{J}_n(\mathbf{m}) \}, \\
 \text{s.t. } \mathcal{C}_m &\leq \mathcal{B}
 \end{aligned} \tag{11}$$

where \mathbf{m} is a sequence of robot motion planning, \mathbf{m}^* is the selected robot motion with maximum information gain, \mathcal{M} is the list of possible robot motion sequences, \mathcal{C}_m is the cost budget, and \mathcal{B} is a budget with regard to time, memory, energy, maximum iteration times, etc. $\mathcal{J}_i(\mathbf{m}), i = 1, 2, \dots, n$, represents the information gathered by the robot motions and n is the number of the information source in the grid^[34]. $\mathcal{D}(\mathbf{m})$ represents the information from planned DCPD directions.

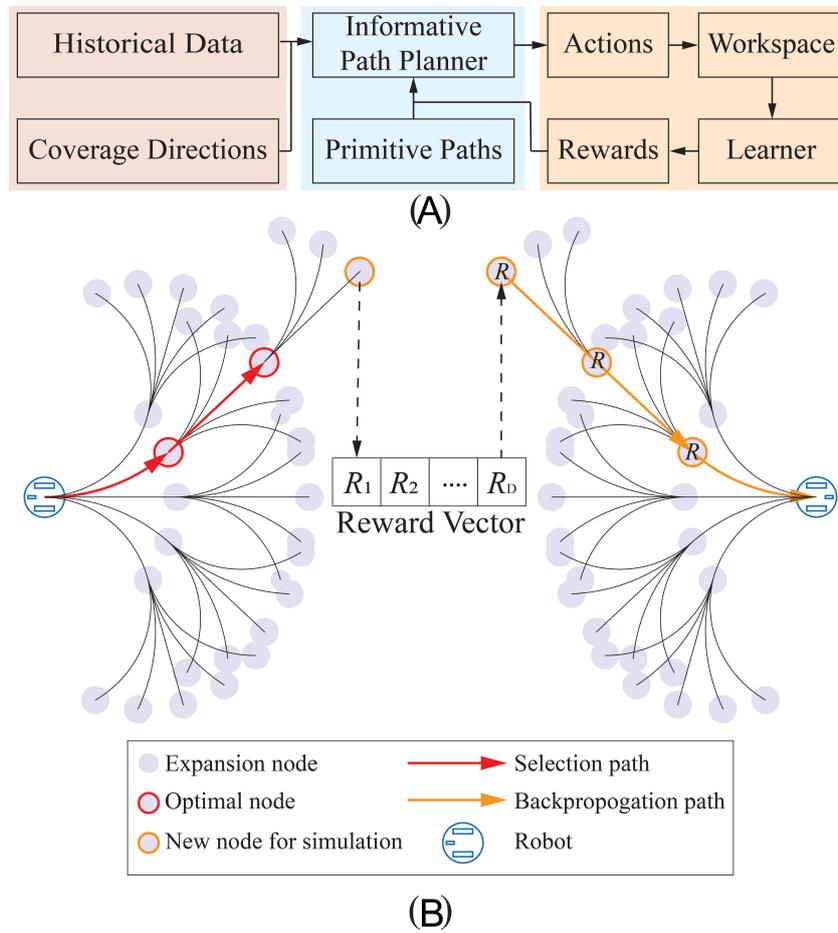


Figure 5. Illustration of the informative planning protocol (IPP): (A) general process of IPP; and (B) a specific example of IPP. The trees in (B) are mirrored for readable presentation, and the two robots are the same.

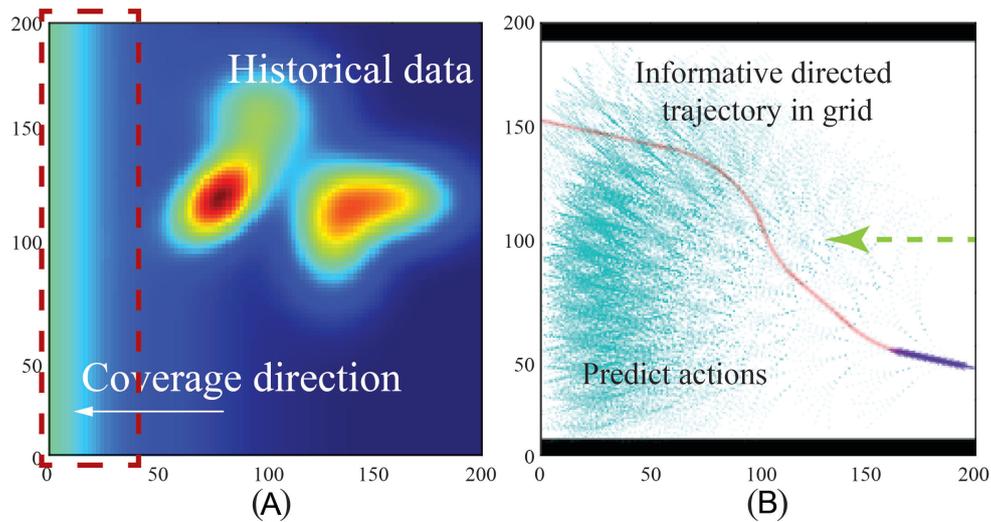


Figure 6. Illustration of the informative directed path planning in the grid, which performs target finding, i.e., where the dead broilers most likely exist: (A) heat map with historical data of dead broilers and DCPD direction information; and (B) the final planned informative directed trajectory in the grid.

The informative directed coverage path planning is shown in Figure 5. In Figure 5A, the planner weighs different objectives (e.g., rapidly covers the space as driven by DPCC or moves to the most likely area as driven by heat map) and optimizes the multi-objective problem to choose a preferred solution from a given set of optimal solutions. The preferred solution is then reflected as actions planned into the workspace, and the learner further improves it in a back-propagation direction. Finally, the most promising solution is determined based on rewards, and a mortality-based coverage path is constructed accordingly. A specific example of the IPP is depicted in Figure 5B. The major component is a primitive path which is built with robot kinematic constraints and divided into five sub-branches, each connected by two nodes. The path is constructed accumulatively during each iteration, and the most informative nodes are explored and selected first for the area with the highest probability of broiler mortality appearance. The rewards are only utilized in the selected nodes for optimization. The framework consists of selection, expansion, simulation, and back-propagation in each iteration. During selection, starting at the initial position, target nodes are recursively selected until an undependable node is encountered. During expansion, each expandable node is subdivided into five sub-branches. During simulation, the multiple-objective function is operated in each expandable node to gain a reward vector. Finally, during back-propagation, the rewards are backed up in each selected and expandable node.

Note that the information in this paper is defined by the broiler mortality historical data for the past 30 days. Since historical data as a priority map also have uncertainty, we utilize the Gaussian process (GP) to transform them into a continuous information map. The amounts of information in each grid can be varied with the number of dead broilers detected. To ensure the robot's path tends to the next planned grid, we set $\mathcal{D}(\mathbf{m})$ as the average value of the current grid information at the adjacent edge, which proportionally decreases with the distance in the opposite direction of DCPD in the grid. More $\mathcal{D}(\mathbf{m})$ information can be clearly found in the red dashed rectangle near the edge of the next grid [Figure 6A]. The planned path tends to be an optimized decision between the length and information gain along it, which might be conflicting in nature. Note that the IPP is only executed within each grid. The path obtained by DCPD becomes an information-oriented application in each grid, guiding the robot to the next grid. The resultant trajectory in the grid is shown in Figure 6B.

The time complexity depends on the Monte Carlo Tree Search (MCTS) tree depth and the steps of each simulation. Assume in an arbitrary iteration i , the tree depth is d_i . Then, the number of total actions is $O(d_i \times n_a)$, where n_a is the number of defined primitive actions. Assume each simulation process costs $O(S)$, which is a problem-dependent value (e.g., the steps of simulation are preset depending on the simulation time budget, considering that longer simulations are more accurate, while shorter simulations return a result more quickly). Then, the complexity for one search iteration is $O(d_i \times n_a + S)$. If we proceed N iterations, the complexity is $O(N(d_i \times n_a + S))$. For multiple objective computation, the complexity becomes $O(N(d_i \times n_a^2 + S))$ due to required extra computation in the MCTS selection step. For practical runtime assessment, a concrete example is when searching 1000 iterations with five available actions on a standard laptop, MCTS takes around 0.3 s and multi-objective MCTS (formally ParetoMCTS) needs about 1 s. With a better computer, the runtime should be even better. This suffices to make a real-time decision.

3.4. Target detection

The target detection is to localize dead birds in real-time processing^[35,36]. The dead birds are detected by the YOLO V4 dead bird detector^[28] [Figure 7]. The images are captured by the camera and input into the detector. The detector should first be well trained and developed in view of the captured images. Then, the trained YOLO V4 takes the acquired image as an input to convert it into a map with default grids. Predefined anchors are tiled onto each grid cell, and predictions of bounding boxes along with confidences and class names are made accordingly. With the non-maximum suppression rule, the bounding box with the highest confidence is retained for dead bird detection. Finally, the detection results, including indices, centroid coordinates, and

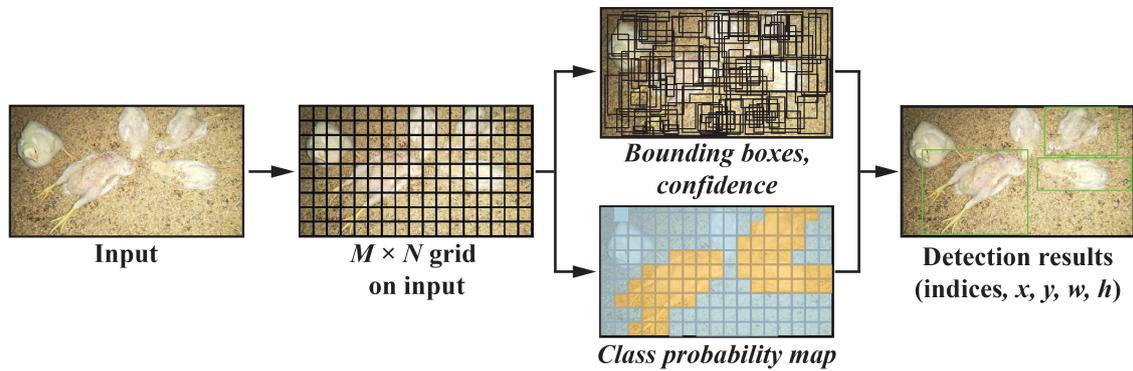


Figure 7. Network architecture of You Only Look Once (YOLO) V4 dead bird detector. Dead birds are enclosed with green bounding boxes.

width and height of bounding boxes, are utilized to pinpoint broiler mortality in each cell. Based on our previous testing, the YOLO V4 achieves on average 90% precision, recall, and F1 scores for detection, 4.8 mm root mean square error for localization, and 7 fps for processing speed when dealing with broiler mortality at various bird ages, light intensities, and bird body gestures. Therefore, it is suitable to serve our purpose of real-time detecting and localizing broiler mortality in this research.

The motivation for selecting the YOLO V4 is that it is a stable, mature, and efficient model and provides reliable and accurate detection results. The major function of this model serving part of the robotic navigation system is to localize dead birds. We also understand that there are more recent deep learning object detection models available, such as YOLO V7 that can be easily trained with custom datasets and replaces YOLO V4. The path planning for the removal robot is completed after the locations of broiler mortality are obtained in the coverage path.

3.5. Multi-target routing scheme

The dead broilers in the coverage path of a broiler barn are located as multiple targets collected by the removal robot. The robot should start from and return to the same point (entrance/exit gate of the barn), as addressed in Section 2.3. Meanwhile, multi-target path planning of the robot should have two major functions: obstacle avoidance and minimal travel distance regarding time and energy consumption. In those regards, multi-target path planning consists of two steps. The first step is point-to-point navigation, which generates the shortest collision-free path between target points. The second step is to navigate the robot to the targets in sequence, so as to minimize the total length of the trajectory.

To achieve the optimal visiting sequence of \mathcal{N} targets, it is necessary to define the cost of path lengths between target points, that is, the shortest distance with obstacle avoidance between target points. Dijkstra's algorithm is utilized to minimize the length of point-to-point navigations. However, Dijkstra's algorithm needs to build a $\mathcal{N}_s \times \mathcal{N}_s$ adjacent matrix. The \mathcal{N}_s is the total number of grids in a decomposed workspace (i.e., 456 for the current broiler barn), resulting in highly expensive computation. Therefore, in light of the broiler layout, we design hub grids to reduce the computational effort. Note that the proposed hub-based multi-target routing (HMTR) scheme is applicable to all row-based environments for multi-target navigation, such as storage buildings, crops, power stations, etc.

The hub grid refers to the grid at the two ends of the feeding and drinking lines. For instance, in Figure 8, six hub grids are shown in the pink dashed line rectangle. It has the characteristics of unobstructed connection of targets in the corridor formed by the feeding and drinking lines, such as the hub grids connecting targets \mathcal{T}_1 , \mathcal{T}_2 , and \mathcal{T}_3 with no collision. It also features a collision-free connection to targets outside the corridor, such as the hub grid that can connect target \mathcal{T}_4 without collision. When the connecting lines between target

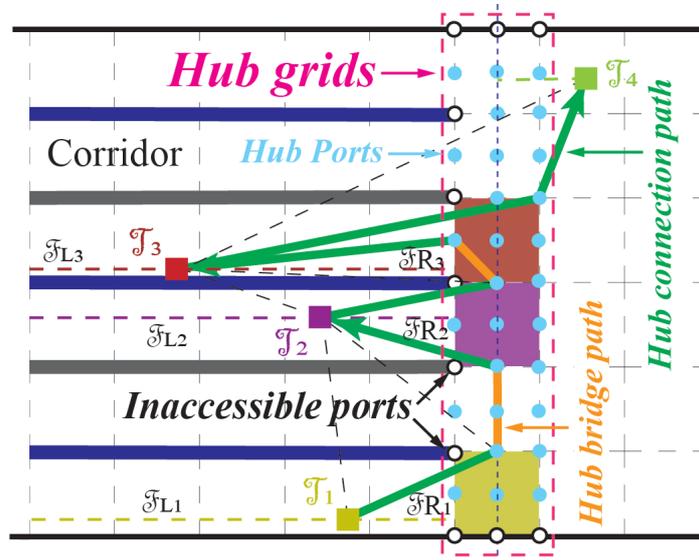


Figure 8. Example illustration of the HMTR scheme.

points are blocked by obstacles, as shown by the black dotted lines in Figure 8, the corresponding hub grids are initially obtained for connection. The selection of the corresponding hub grids to the targets i and j is made by the total distance. Let \mathcal{F}_{Li} be the distance from the target point i to the centerline of the hub grid on the left side and \mathcal{F}_{Ri} be the distance from the target point i to the centerline of the hub grid on the right side. If $(\mathcal{F}_{Li} + \mathcal{F}_{Lj}) > (\mathcal{F}_{Ri} + \mathcal{F}_{Rj})$, the hub grid on the right side is selected. If $(\mathcal{F}_{Li} + \mathcal{F}_{Lj}) \leq (\mathcal{F}_{Ri} + \mathcal{F}_{Rj})$, the hub grid on the left side is selected.

To further optimize the path selection, we decompose the hub grid into nine ports that can be connected. When the port is blocked by a feeding or drinking line, it is regarded as inaccessible. The total path length between target points i and j is \mathcal{Q}_{ij} , which is the addition of the length of the hub bridge path and the length of the hub connection path shown in Figure 8. Dijkstra’s algorithm is utilized to minimize the length of point-to-point navigations. The trajectory is established between two points, which can be selected from each accessible port or the dead broilers. Each point is recursively connected with the other points, and the distances of connection lines passing through obstacles are assigned by an infinite number. As a result, point-to-point navigation with obstacles is excluded, and feasible solutions are retained. The shortest paths between each pair of points are selected from those feasible solutions. Each dead bird skips intermediate connections between adjacent grids and connect directly to the nearest port in the hub grid or to another dead bird, avoiding obstacles with the shortest traveling distance and improving computational efficiency.

The Miller–Tucker–Zemlin (MTZ) algorithm is then introduced for sequencing the navigation among the target points^[37]. We define the connection variable as \mathcal{C}_{ij} and path lengths between target points i and j as \mathcal{Q}_{ij} . The objective function is then given by

$$\min \sum_i \sum_j \mathcal{Q}_{ij} \mathcal{C}_{ij} \tag{12}$$

To ensure that the result is a valid tour, several constraints must be added [Equation (13)]

$$\begin{aligned} \sum_{i \in \mathcal{V}} \mathcal{C}_{ij} &= 1, \quad \forall j \in \mathcal{V}, i \neq j \\ \sum_{j \in \mathcal{V}} \mathcal{C}_{ij} &= 1, \quad \forall i \in \mathcal{V}, i \neq j \end{aligned} \tag{13}$$

where \mathcal{V} is the set of number of target points. The first set of equalities is a go-to constraint, which requires that each target point is visited only once and is different from the others. The second set of equalities is a come-from constraint, which requires that the departure of each target point also occurs once and is different from the other ones.

Equations (12) and (13) may form multiple enclosed paths for all target points, with the formation of subtours. The MTZ constraints are utilized to eliminate unnecessary subtours and ensure a single enclosed path with the shortest length [Equation (14)]. Each visited node is labeled, resulting in the uniqueness of each visit.

$$\mathcal{U}_i + \mathfrak{C}_{ij} \leq \mathcal{U}_j + (n - 1) \times (1 - \mathfrak{C}_{ij}) \quad 2 \leq j \leq n \quad (14)$$

where \mathcal{U} is the sequence of each target point visited and n is the number of target points. When $\mathfrak{C}_{ij} = 1$, the equality holds ($\mathcal{U}_i + 1 = \mathcal{U}_j$).

Based on the MTZ algorithm, we finally obtain a visiting sequence of dead broilers. The robot with that sequence starts and ends at the gate of the broiler barn with the shortest collision-free paths.

Algorithm 2: Hub-based multi-target routing (HMTR) scheme

Load \mathcal{N} obtained broiler locations as $\mathcal{T}_1, \dots, \mathcal{T}_\mathcal{N}$ and robot initial location as \mathcal{T}_0 .

Initialize the connection variable matrix \mathfrak{C} ;

for $i = 0 : \mathcal{N}$ do

 for $j = 0 : \mathcal{N}$ do

 if \mathcal{T}_i can direct connect to \mathcal{T}_j then

$\mathfrak{L}_{ij} = \sqrt{(x_{\mathcal{T}_i} - x_{\mathcal{T}_j})^2 + (y_{\mathcal{T}_i} - y_{\mathcal{T}_j})^2}$;

 else

 if $(\mathcal{F}_{Li} + \mathcal{F}_{Lj}) > (\mathcal{F}_{Ri} + \mathcal{F}_{Rj})$ then

 Select corresponding right-side hub grids;

 else

 Select corresponding left-side hub grids;

 end

 Select the best ports in the hub grids;

 Use Dijkstra algorithm to obtain \mathfrak{L}_{ij}

 end

 end

end

Use MTZ method to find the optimal visiting sequence.

4. SIMULATION AND COMPARATIVE STUDIES

Simulation and comparative studies were carried out to validate the effectiveness and efficiency of the proposed path planning methods of real-time autonomous mobile robots. The environments referred to are those in a typical broiler barn. The robot was parameterized as described in Section 2.1 based on commercialized poultry robots. These settings were taken to exemplify our proposed method and could be tailored according to specific applications.

Table 1. Comparison of path length and approximated travel time in broiler barn. The robot is assumed to have a linear velocity of 0.4 m/s

Models	Path length (m)	Approximated travel time (min)
Proposed model	1182	49
BCD	16,408	683

4.1. Path planning for the detection robot

The proposed DCPD was initially compared with the zigzag and spiral methods. The start and end points for the broiler mortality robot were the same for the three methods, and our proposed method did not have redundant return paths. The total path length of the three methods was 912 m for the proposed method, 920 m for the zigzag method, and 917 m for the spiral method, respectively. The overall path based on the DCPD is depicted in [Figure 9B](#).

With the assistance of DCPD, IPP is shown in [Figure 9A and C](#). The cases are executed between two feeding/drinking lines, indicated as black bars on the left and right of the figures. The robot accounts for the overall DCPD direction and historical dead bird distribution data depicted as heat maps with red color areas indicating a higher possibility of broiler mortality appearance. Cyan areas indicate multiple branches subdivided by IPP. A path indicated as a red line designates a maximal information gain. Once the robot meets a random obstacle (indicated as the black object in [Figure 10](#)), such as welfare enrichment (e.g., perches, straw bales, and pecking stones)^[38], the information gain of IPP is optimized with the historical data and DCPD direction while avoiding the obstacle. Not only is the trajectory planned with previous data, but it is also updated with new data [[Figure 11](#)], resulting in a real-time efficient path for the robot. In [Figure 11](#), the trajectory changes from roaming between two obstacles to roaming atop the upper obstacle after the broiler mortality distribution data are updated based on the YOLO V4 dead bird detection and localization.

Limited by the working environment in the broiler barn, it is assumed that the detection range of the robot is 50 cm and the linear speed is 0.4 m/s. Through the traditional CCPP algorithm, such as BCD method, the path length of the CCPP is 16,408 m, which requires approximately 683 min to complete. However, DCPD with IPP is fine-tuned on the basis of the information gained on the paths generated by DCPD. In light of various information distribution in different environments, the final generated global trajectory length in each grid is only 12%-37% longer than DCPD. The final path length of the DCPD with IPP is 1182 m, which requires approximately 49 min to complete. Compared with the traditional CCPP, our proposed algorithm has a shorter path length and the travel time also meets the demands of broiler barns. The final results are outlined in [Table 1](#).

4.2. Path planning of the removal robot

With the AI-based advancement of sensing techniques, locations of broiler mortality are obtained in an overall trajectory for the second robot. The HMTR receives the mortality location information and generates a collision-free route to reach the targets in a reasonable and efficient sequence so that the total traveling distance is minimized. To validate the adaptability and efficiency of our algorithms in various number of targets, three datasets were selected for simulation and comparative studies^[39]. In each dataset, we iteratively performed 30 executions to compute the mean and standard deviation of path length. [Table 2](#) summarizes the qualitative comparison results between the features of our algorithm and state-of-art models, such as genetic algorithm (GA), particle swarm optimization (PSO), self-organizing maps (SOM), and imperialist competitive algorithm (ICA). The SOM algorithm is similar to a typical artificial neural network algorithm, except it utilizes a competitive learning process instead of back-propagation that utilizes gradient descent. The ICA algorithm is a biologically inspired algorithm, which simulates the social-political process of imperialism and

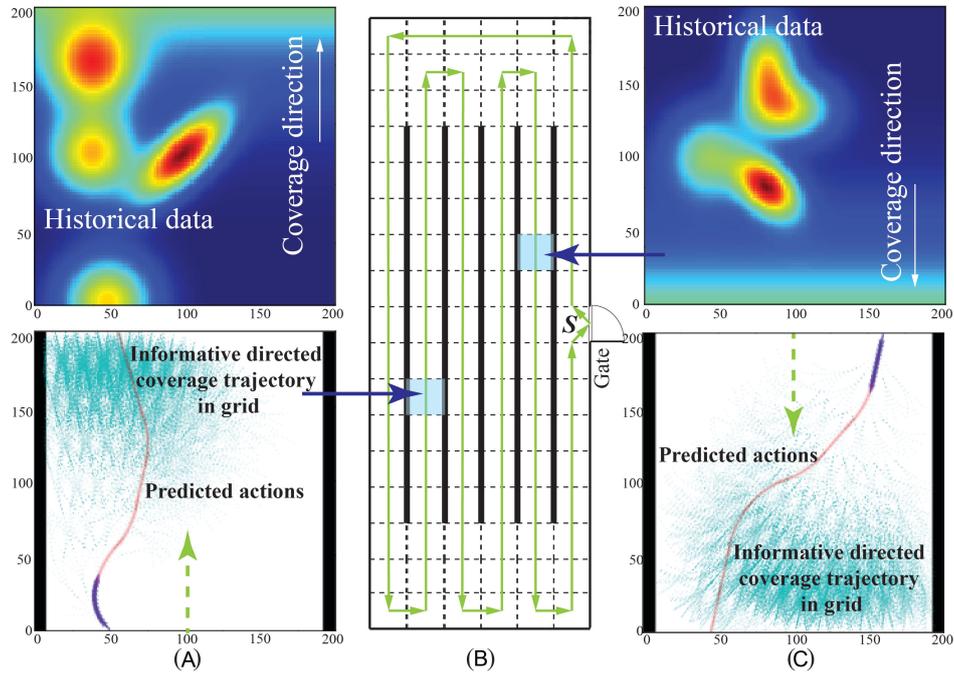


Figure 9. Illustration of robot global path planning in the tested workspace: (A and C) examples of informative trajectory between two feeding/drinking lines in various DCPD directions and historical dead bird distribution data; and (B) the overall DCPD trajectory.

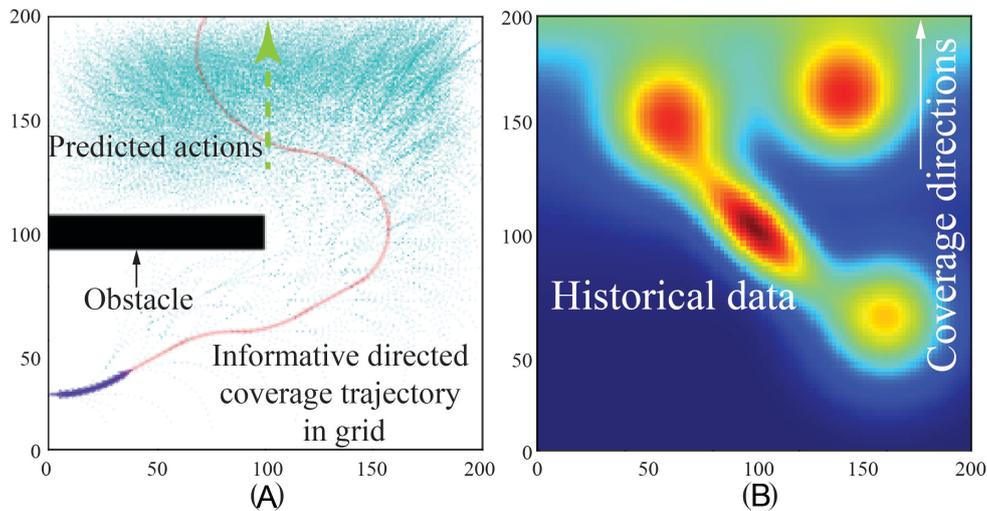


Figure 10. Informative planning protocol avoids a random obstacle in the grid: (A) the final planned informative directed coverage trajectory in grid; and (B) heat map with historical data of dead broilers and DCPD direction information.

imperialistic competition in humans.

Then, the reactive local navigator was utilized to generate real-time commands (e.g., acceleration, deceleration, and turning) for the robot arriving at a dead bird. Figure 12 shows three instances of grids for local navigation. Based on our simulation studies, using the dynamic window approach (DWA) local navigator, the robot successfully reached the existing dead birds and simultaneously avoided moving/still broilers. Robot running time and local trajectories could vary with grid locations, dead bird number in a grid, and distribution of live birds. As shown in Figure 12A, the two connected grids were between feeding and drinking lines surrounded

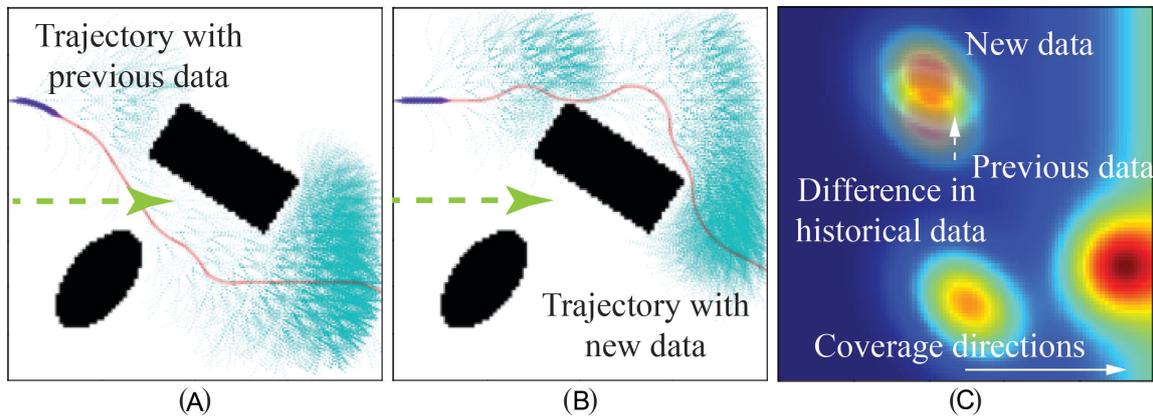


Figure 11. Different trajectories generated by updated historical data: (A) the planned trajectory with previous data; (B) the new planned trajectory with new data; and (C) heat map with different historical data.

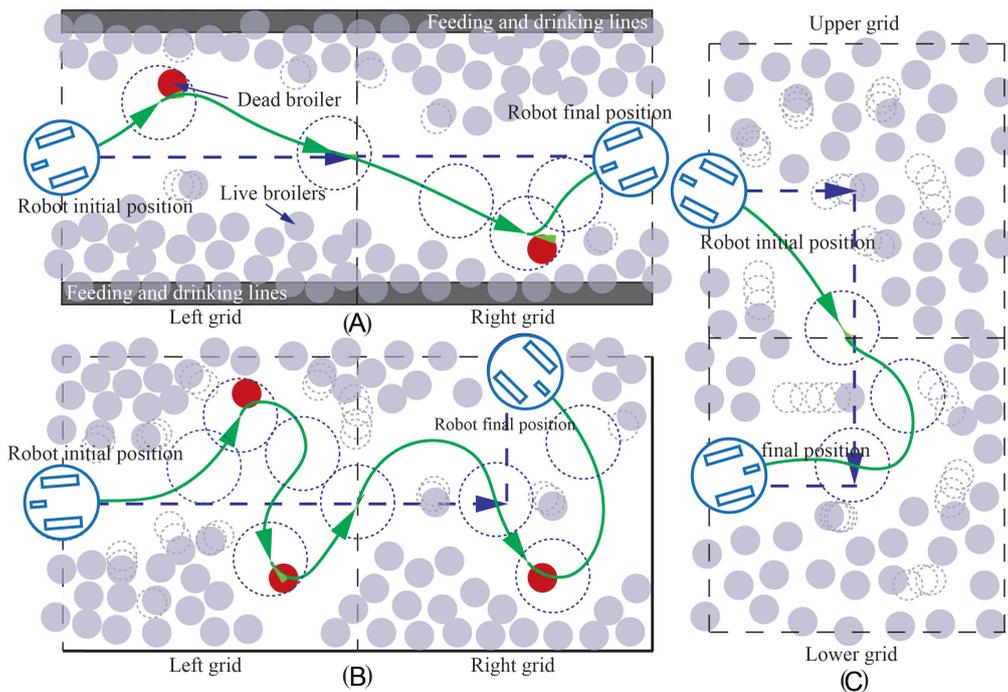


Figure 12. Illustration of robot local navigation at various locations: (A) sample of grids between feeding and drinking lines; (B) sample of grids at the corner; and (C) sample of grids for robot U-turning. Green lines represent the robot real motion trajectory, blue dashed lines represent trajectories generated by global path planning, navy blue solid circles are robot final positions in the figure, navy blue dashed circles illustrate different stages of robot positions, grey rectangles are feeding lines and drinking lines, red dots are dead broilers, and slate grey dots are live broilers.

by birds, which were the most common cases in broiler barns, and two dead birds were distributed separately in the two grids. The trajectory of the robot was relatively straightforward due to less blockage of moving birds, in which the running time was 22–24 s for the left grid and 24–26 s for the right grid. As shown in Figure 12B, the connected grids were at the corner of the barn with two dead birds in the left grid and one dead bird in the right grid, resulting in a running time of 56–58 s for the left grid and 34–36 s for the right grid. As shown in Figure 12C, the robot needed to make a U-turn between the two connected grids to complete the path. Even though no dead birds were detected in the grids, the live birds could still interfere with the robot due to their random distribution and unpredictable movements. Therefore, the running time varied for the two grids:

Table 2. Comparison of minimum path length, average path length, and standard deviation (STD) of path length with other models. The values are reported for 30 executions

Test data set	Model	Min length (m)	Average length(m)	Length STD (m)
Bays29	GA	9.47e+03	9.98e+03	4.46e+03
	PSO	9.07e+03	9.25e+03	2.38e+03
	SOM	1.01E+04	1.28E+04	7.59E+03
	ICA	1.09E+04	1.22E+04	9.87E+03
	Proposed model	9.07e+03	9.07e+03	0
KroA200	GA	2.39e+05	2.57e+05	1.15e+04
	PSO	1.09e+05	1.17e+05	5.50e+03
	SOM	2.13E+05	2.60E+05	2.19E+04
	ICA	2.12E+05	2.60E+05	6.67E+03
	Proposed model	1.06e+05	1.06e+05	0
PA561	GA	1.37e+05	1.93e+05	6.30e+04
	PSO	1.11e+05	1.14e+05	1.60e+03
	SOM	1.01E+05	1.21E+05	1.84E+04
	ICA	1.48E+05	1.51E+05	2.43E+03
	Proposed model	1.03e+05	1.03e+05	0

12–14 s for the upper grid and 19–21 s for the lower grid. In a broiler barn, the cases shown in [Figure 12A](#) took up the most proportion (420 grids), within which most contained no dead broilers, and the average robot running time was 20–28 s in a single grid. Therefore, our proposed methods have a reasonable running time in commercial-scale poultry barns, and they would be beneficial for rapidly detecting and collecting broiler mortality. Simulation and comparison studies demonstrate the effectiveness and efficiency of the proposed real-time robot safety-aware navigation.

5. CONCLUSION

A robotic system for detecting and collecting dead broilers in a barn is a promising direction for solving the issue of broiler mortality removal. Aiming at this vision, we developed an informative planning protocol-based multi-layer robot navigation system through detection and removal robots. The detection robot consists of DCPD for constructing the overall trajectory; IPP for detailing the trajectory based on historical data, DCPD direction, and obstacles; and a YOLO V4 dead bird detector for providing the precise locations of broiler mortality along the trajectory. The removal robot receives the mortality location information and plans an optimal ordered route by the HMTR scheme. The comparison and simulation results demonstrate the great potential of the proposed methods for robot navigation, being useful tools for supporting precision broiler management.

There are also many possible avenues for future work. A challenging extension is that we will integrate our algorithm to test the robots in the real scene of the broiler barn and develop a suitable human–robot interaction platform for effective control. Another interesting topic is the application of multiple robots to cooperatively search the environment to reduce the overall work time and simultaneously complete the functions of search, identification, and removal to improve efficiency.

DECLARATIONS

Acknowledgments

The authors would like to thank the editor-in-chief, the associate editor, and the anonymous reviewers for their valuable comments.

Authors' contributions

Made substantial contributions to the research, idea generation, algorithm design, simulation, wrote and edited the original draft: Lei T, Li G, Luo C

Performed critical review, commentary and revision, as well as provided administrative, technical, and material support: Zhang L, Liu L, Gates R

Financial support and sponsorship

None.

Availability of data and materials

Not applicable.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2022.

REFERENCES

1. United States Department of Agriculture NASS. Poultry-production and value 2020 summary; 2018. Available from: https://www.nass.usda.gov/Publications/Todays_Reports/reports/plva0421.pdf [Last accessed on 30 Aug 2022].
2. Tottori J, Yamaguchi R, Murakawa Y, Sato M, Uchida K, et al. The use of feed restriction for mortality control of chickens in broiler farms. *Avian Dis* 1997;433–37. DOI
3. Schwann-Lardner K, Fancher B, Gomis S, Van Kessel A, Dalal S, et al. Effect of day length on cause of mortality, leg health, and ocular health in broilers. *Poultry Sci* 2013;92:1–11. DOI
4. Astill J, Dara RA, Fraser ED, Roberts B, Sharif S. Smart poultry management: smart sensors, big data, and the internet of things. *Comput Electr Agricult* 2020;170:105291. DOI
5. Ren G, Lin T, Ying Y, Chowdhary G, Ting K. Agricultural robotics research applicable to poultry production: a review. *Comput Electr Agricult* 2020;169:105216.
6. Vroegindewij BA, Blaauw SK, IJsselmuiden JM, van Henten EJ. Evaluation of the performance of PoultryBot, an autonomous mobile robotic platform for poultry houses. *Biosyst engineer* 2018;174:295–315. DOI
7. Li G, Chesser GD, Huang Y, Zhao Y, Purswell JL, et al. Development and optimization of a deep-learning-based egg collecting robot. *Trans ASABE* 2021:0. DOI
8. Liu L, Luo C, Shen F. Multi-agent formation control with target tracking and navigation. In: IEEE International Conference on Information and Automation (ICIA); 2017. pp. 98–103. DOI
9. Chen J, Luo C, Krishnan M, Paulik M, Tang Y. An enhanced dynamic delaunay triangulation-based path planning algorithm for autonomous mobile robot navigation. In: Intelligent Robots and Computer Vision XXVII: Algorithms and Techniques. vol. 7539. SPIE; 2010. pp. 253–64. DOI
10. Zhao W, Lun R, Gordon C, Fofana AB, Espy DD, et al. A privacy-aware Kinect-based system for healthcare professionals. In: IEEE International Conference on Electro Information Technology (EIT); 2016. pp. 0205–10. DOI
11. Lei T, Luo C, Ball JE, Rahimi S. A graph-based ant-like approach to optimal path planning. In: *IEEE Congr Evol Comput (CEC)*. IEEE; 2020. pp. 1–6. DOI
12. Lei T, Luo C, Jan GE, Fung K. Variable speed robot navigation by an ACO approach. In: International Conference on Swarm Intelligence. Springer; 2019. pp. 232–42. DOI
13. Wang L, Luo C, Li M, Cai J. Trajectory planning of an autonomous mobile robot by evolving ant colony system. *Int J Robot Autom* 2017;32:406–13. https://www.researchgate.net/profile/Chaomin-Luo/publication/319032211_Trajectory_planning_of_an_autonomous_mobile_robot_by_evolutionary_ant_colony_system/links/5997a952458515644325892e/Trajectory-planning-of-an-autonomous-mobile-robot-by-evolutionary-ant-colony-system.pdf [Last accessed on 30 Aug 2022].
14. Lei T, Luo C, Ball JE, Bi Z. A hybrid fireworks algorithm to navigation and mapping. In: Handbook of Research on Fireworks Algorithms and Swarm Intelligence. IGI Global; 2020. pp. 213–32. DOI
15. Lei T, Luo C, Sellers T, Rahimi S. A bat-pigeon algorithm to crack detection-enabled autonomous vehicle navigation and mapping. *Intell Syst Applic* 2021;12:200053. DOI
16. Wang J, Meng MQH. Optimal path planning using generalized Voronoi graph and multiple potential functions. *IEEE Trans Industr Electron* 2020;67:10621–30. DOI
17. Yang SX, Luo C. A neural network approach to complete coverage path planning. *IEEE Trans Syst, Man, Cybern, Part B (Cybernetics)* 2004;34:718–24. DOI

18. Luo C, Yang SX, Krishnan M, Paulik M. An effective vector-driven biologically-motivated neural network algorithm to real-time autonomous robot navigation. In: IEEE International Conference on Robotics and Automation (ICRA); 2014. pp. 4094–99. DOI
19. Zhu D, Tian C, Jiang X, Luo C. Multi-AUVs cooperative complete coverage path planning based on GBNN algorithm. In: 29th Chinese Control and Decision Conference (CCDC); 2017. pp. 6761–66. DOI
20. Lei T, Sellers T, Rahimi S, Cheng S, Luo C. A nature-inspired algorithm to adaptively safe navigation of a Covid-19 disinfection robot. In: International Conference on Intelligent Robotics and Applications. Springer; 2021. pp. 123–34. DOI
21. Luo C, Gao J, Murphey YL, Jan GE. A computationally efficient neural dynamics approach to trajectory planning of an intelligent vehicle. In: 2014 International Joint Conference on Neural Networks (IJCNN). IEEE; 2014. pp. 934–39. DOI
22. Luo C, Yang SX. A bioinspired neural network for real-time concurrent map building and complete coverage robot navigation in unknown environments. *IEEE Trans Neural Netw* 2008;19:1279–98. DOI
23. Acar EU, Choset H. Sensor-based coverage of unknown environments: Incremental construction of morse decompositions. *Int J Robot Res* 2002;21:345–66. DOI
24. Nasirian B, Mehrandehz M, Janabi-Sharifi F. Efficient coverage path planning for mobile disinfecting robots using graph-based representation of environment. *Front Robot AI* 2021;8:4. DOI
25. Lei T, Luo C, Jan G, Bi Z. Deep learning-based complete coverage path planning with re-joint and obstacle fusion paradigm. *Front Robot AI* 2022. DOI
26. Li G, Hui X, Lin F, Zhao Y. Developing and evaluating poultry preening behavior detectors via mask region-based convolutional neural network. *Animals* 2020;10:1762. DOI
27. Li G, Xu Y, Zhao Y, Du Q, Huang Y. Evaluating convolutional neural networks for cage-free floor egg detection. *Sensors* 2020;20:332. DOI
28. Bochkovskiy A, Wang CY, Liao HYM. Yolov4: optimal speed and accuracy of object detection. *arXiv preprint arXiv:200410934* 2020. DOI
29. Tabler G, Berry I, Xin H, Barton T. Spatial distribution of death losses in broiler flocks. *J appl poultry res* 2002;11:388–96. DOI
30. Li G, Ji B, Li B, Shi Z, Zhao Y, et al. Assessment of layer pullet drinking behaviors under selectable light colors using convolutional neural network. *Comput Electr Agricult* 2020;172:105333. DOI
31. Lei T, Luo C, Sellers T, Wang Y, Liu L. Multi-task allocation framework with spatial dislocation collision avoidance for multiple aerial robots. *IEEE Trans Aerosp Electr Syst* 2022. DOI
32. United States Department of Agriculture NASS. Agricultural resource management survey (ARMS) of the U.S. broiler industry; 2011. https://www.nass.usda.gov/Surveys/Guide_to_NASS_Surveys/Ag_Resource_Management/ARMS_Broiler_Factsheet/Poultry%20Results%20-%20Fact%20Sheet.pdf [Last accessed on 30 Aug 2022].
33. Mendeş M. Growth curves for body weight and some body measurement of Ross 308 broiler chickens. *J Appli Animal Res* 2009;36:85–88. DOI
34. Chen W, Liu L. Pareto Monte Carlo tree search for multi-objective informative planning. *arXiv preprint arXiv:211101825* 2021. DOI
35. Yang Y, Deng Q, Shen F, Zhao J, Luo C. A shapelet learning method for time series classification. In: IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI); 2016. pp. 423–30. DOI
36. Xing Y, Shen F, Luo C, Zhao J. L3-SVM: a lifelong learning method for SVM. In: 2015 international joint conference on neural networks (IJCNN). IEEE; 2015. pp. 1–8. DOI
37. Integer programming formulation of traveling salesman problems. *J ACM (JACM)* 1960;7:326–29. DOI
38. Bergmann S, Schwarzer A, Wilutzky K, Louton H, Bachmeier J, et al. Behavior as welfare indicator for the rearing of broilers in an enriched husbandry environment—a field study. *J Veterin Behav* 2017;19:90–101. <https://www.sciencedirect.com/science/article/pii/S1558787816301915> [Last accessed on 30 Aug 2022].
39. Bixby B, Reinelt G. Traveling salesman problem library; 2022. <http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsp/index.html> [Last accessed on 30 Aug 2022].