Article

Design and Coverage Path Planning of a Disinfection Robot

Pengjie Xu, Xinyi Chen and Qirong Tang *

The Laboratory of Robotics and Multibody System, School of Mechanical Engineering, Tongji University, Shanghai 201804, China
* Correspondence: qirong.tang@tongji.edu.cn

Abstract: Eliminating pathogen exposure is an important approach to control outbreaks of epidemics such as COVID-19 (coronavirus disease 2019). To deal with pathogenic environments, using disinfection robots is a practicable choice. This research formulates a 3D (three-dimensional) spatial disinfection strategy for a disinfection robot. First, a disinfection robot is designed with an extensible control framework for the integration of additional functions. The robot has eight degrees of freedom that can handle disinfection tasks in complex 3D environments where normal disinfection robots lack the capability to ensure complete disinfection. An ingenious clamping mechanism is designed to increase flexibility and adaptability. Secondly, a new coverage path planning algorithm targeted at the spraying area is used. This algorithm aims to achieve an optimal path via the rotating calipers algorithm after transformation between a 2D (two-dimensional) array and 3D space. Finally, the performance of the designed robot is tested through a series of simulations and experiments in various spaces that humans usually live in. The results demonstrate that the robot can effectively perform disinfection tasks both in computer simulation and in reality.

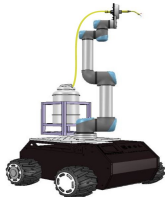**Keywords:** COVID-19; disinfection robot; coverage path planing; extensible control framework

1. Introduction

The outbreak of COVID-19 has caused an unprecedented global threat to public health and the economy. Although the spread of some new variants has led to a situation of low death rates worldwide, improving the configuration of healthcare facilities to better respond to the status quo is still urgent. Moreover, when facing the challenges of unknown contagious diseases, reducing or eliminating exposure is always one of the crucial ways to maintain health. Traditional methods of exposure elimination, such as manual disinfectant spraying, are restricted by the availability of operators in terms of work time and concentration. Disinfection robotic systems available for remote control represent a practical and reasonable tool to fulfill various anti-epidemic purposes. Disinfection or sterilization robots can remain unhindered by peak hours or terrible operating conditions and carry out their duties all day long to prevent pathogenic attacks [1].

Many researchers in automation and engineering have also actively responded to this conception. Since the onset of the pandemic, MIT and Boston Dynamics have begun to cooperate in the design of patient interaction robots in order to reduce the frontline virus exposure of healthcare personnel [2]. The use of ultraviolet (UV) light is especially noticed because its energy is sufficient to destroy the DNA or RNA of any microorganism, and UV lights have been installed on many experimental mobile robots in an environment to be sanitized [3]. These robots utilize annotated points of interest on the map to scan the area and automatically reach the location to perform specific disinfection tasks. However, limited by the characteristics of UV light and their operating distance, these robots can only execute on-ground tasks. In a previous study [4], an intelligent disinfection robotic system was developed to enable robots to spray disinfectants in operating theaters or patients' rooms, based on the results of controlled experiments and the requirements for hospital disinfection. A key feature of this research is the application of a novel CLO_2 disinfection technology to sterilize bacteria and viruses in the air and on surfaces. Another noteworthy

feature is that this robot is also based on a mobile platform, by which the disinfection abilities are severely curtailed in 3D space, such as on table tops, door handles, elevator buttons, and other regular locations that humans usually touch [5,6]. There are additional applications for disinfection robots. For example, SOMATIC is a commercial cleaning robot that can perform approximately eight hours of cleaning work each time it runs. It can even open doors and ride elevators to move around the building. The service range of this robot includes airports, casinos, office spaces, and other places such as large restrooms [7]. The designed robot is compared with two typical disinfection robots as listed in Table 1.

Table 1. Robots used for disinfection.

Related Works	Paper [2]	Paper [3]	Our robot
Disinfection method	ultraviolet light	spraying	spraying
Disinfection range	open-sided 3D space	open-sided 2D space	3D space
Degrees of freedom	5	3	8
Robot			

In order to better accomplish disinfection tasks, reasonable planning of the spraying path is a key element of the designed robot. We can convert this problem into a coverage path problem in 3D space. The coverage path planning (CPP) algorithm is a popular topic and has been substantially studied. It is applied for underwater or land inspection, unmanned aerial vehicles, milling, floor cleaning, painting, and medical tasks [8]. Both offline and online CPP algorithms have been previously proposed. For example, Han [9] proposed a complete CPP obstacle avoidance algorithm for underwater gliders in the sea, with shorter path planning and economized energy supply. Only three years have passed since the outbreak of COVID-19, so few researchers have performed in-depth studies on the disinfection path problem. Among them, Hong [10] designed a path planning model to acquire an optimal path based on the map and the robot's current location; however, the setting is restricted to conference rooms. Another study [11] investigated how the robotic path can be optimized to achieve the maximum UV-irradiation performance based on environmental geometry. Bähnemann et al. presented a path planner for low-altitude terrain coverage in known environments with unmanned rotary-wing micro aerial vehicles [12]. In this research, the algorithm was applied in conference rooms and was tested to verify the accuracy, cost, and efficiency of the robotic execution of disinfection tasks.

In line with the above studies, the objective of the present research is to design a disinfection robot that can provide effective zero-contact disinfection services for common environments in 3D space. The main characteristics of this research that differ from the existing studies are as follows.

(1) This research provides a unique design of robotic architecture from software to hardware. A key feature of our robot is that it is endowed with eight DoFs (degrees of freedom) to conduct disinfection tasks in common 3D environments based on the integration of a mobile platform, a manipulator, and the designed disinfection instrument.

(2) According to the characteristics of the designed robot, a CPP algorithm is used for disinfection in 3D space. In contrast to [9,11], the particularities are taken as the factors in the disinfection strategy.

(3) Mechanical design, multisensor data fusion, control systems, path planning, spraying disinfection, and other technologies are adopted to conduct a series of simulations and experiments to verify the performance of the robot and the proposed algorithm.

The remainder of this research is structured as follows. The second section explains the design of the disinfection robot in detail; the third section describes the proposed CPP algorithm; the fourth section verifies the performance of the robot and the proposed algorithm with two numerical examples and an experiment; and the fifth section, i.e., the conclusion, summarizes the results and outlines future work. The patents obtained based on this research are listed in the sixth section.

2. Robot Design

2.1. Function Analysis and Design

According to the requirements of COVID-19 prevention, the disinfection robot should have flexibility in 3D space that traditional robots lack. Namely, the robot should be able to correct its end-effector pose based on the specific disinfection demands and be able to smoothly adapt to various application scenarios. Remote control is also a prerequisite for dealing with unique working conditions. Therefore, power supply, communication, storage performance, and other corresponding modules of the robot should be taken into consideration. The specific requirements are listed as follows:

1. The width of the robot should be less than 1.2 m to move through elevators, bedroom doors, and other similar scenarios;
2. The spraying height should be around 1.6 m to cover most common environments;
3. The manipulator should have ample DoFs (operability and flexibility in 3D space) to access any corner that humans can touch;
4. One charge should enable a working duration of 6 h for the robot, and the robot should be able to work at different voltage values to fit different disinfection instruments;
5. The gripper should be a general part to fit different disinfection instruments;
6. The hardware and software of the robot should be extensible to achieve multifunctional integration.

2.2. Gripper Design

As one of the key components of the robot, a flexible gripper is designed to grip the spraying disinfection instruments. The gripper consists of two modules, namely the fixed and the motion modules. The fixed module comprises a fixed base body, a guide seat, and a fixed seat. The motion module comprises clamping jaws, a guide seat, and a drive plate. More definitions of the corresponding structures are shown in Figure 1.
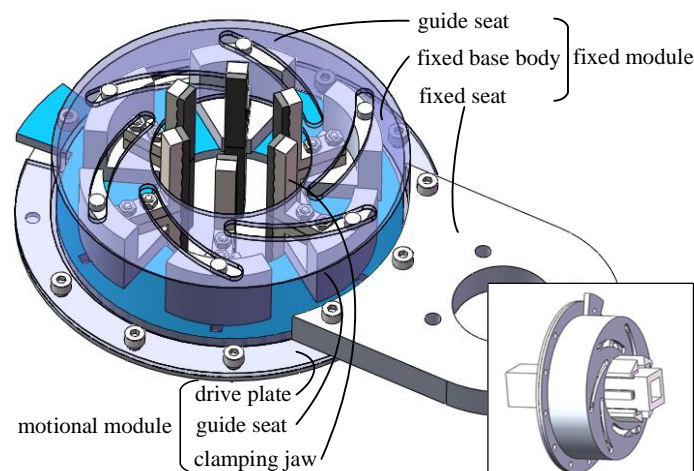
Figure 1. Structure of the gripper.

Clamping jaws are installed on the guide seat along its circumference in the radial direction in a guiding way, while the guide seat and the drive plate are fixed to the base body so that each clamping jaw can extend to clamp or retract to release an object in the radial direction when the guide seat and the drive plate can rotate relatively. The design

of the clamping mechanism is simple in structure, enabling convenient adjustment of the clamping range; the clamping jaws can also be replaced to accommodate different shapes, as shown in the subgraph of Figure 1 (bottom right). The mechanical principle of the gripper can be seen in the accompanying video.

2.3. The Robotic System

To achieve the above-mentioned functions, a mobile manipulator needs to be installed to operate the spraying disinfection instruments. The mobile manipulator consists of a UR5, a serial-chain manipulator that is mounted on a nonholonomic mobile platform with two DoFs, as shown in the Figure 2. The robot should support the weights of all modules up to 50 kg and be able to move freely across a flat surface. The left and the right motors of the mobile platform are driven by 12-volt batteries. The function of the manipulator is to conduct disinfection operations in 3D space via the disinfection instrument, which is composed of a compressed air device, a thin pipe, a nozzle, a gripper, etc. The nozzle can be controlled by an electric valve with an IO interface. In order to fix the disinfection instruments to the end of the manipulator tightly and suitably for different applications, the gripper is installed on the flange of the UR5. With USB hubs, functional extension can be achieved by linking to different external sensors.
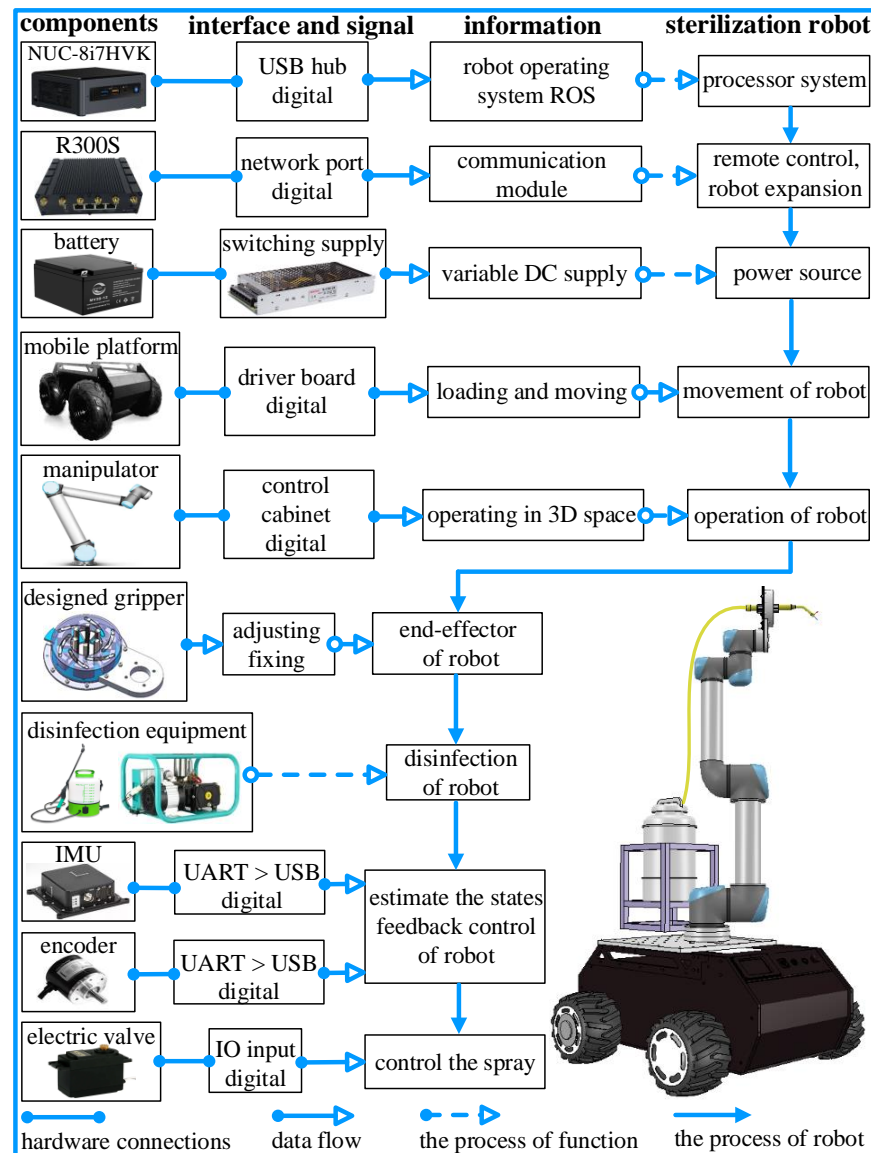

Figure 2. The designed disinfection robot system.

The software design of the disinfection robot is based on the robot operating system (ROS 1) [13]. The communication network uses a TCP/IP protocol to facilitate telecommunication with the upper computer or other intelligent robots [14]. An Intel NUC (i7-8559, 16GB RAM) works as a local computer, while the motion and control functions are encapsulated as ROS packages and executed in the NUC. The super potential of the robot is achieved by reserving ample computing power and interfaces for function extensions, such as for image processing and machine learning. Compared to various other disinfection robotic systems, this robotic system incorporates its operational parameters and various functionalities into its hardware and software structures (see more details in Figure 2).

2.4. Kinematic Analysis

The robot has eight DoFs (two in the mobile platform and six in the manipulator) to conquer difficulties and realize the desired disinfection motions. Before the kinematic analysis, one can use $\boldsymbol{P}_g = \left[x_g, y_g, z_g, \alpha_g, \beta_g, \gamma_g\right]^{\mathrm{T}}$ to describe the motions of the robot in the world frame (Σ_g), where the first three elements in $\boldsymbol{P}_g$ represent the position, and α_g, β_g, and γ_g are used to specify the orientations in the triplet of roll, pitch, and yaw (RPY) angles, respectively [15]. In order to achieve better spraying, the motions along the x_g and y_g directions are achieved by a mobile platform. The UR5 manipulator aims to realize the rest of the absolute motion in the world frame or absolute motion in the frame of the mobile platform. The desired joint-space trajectories can be obtained based on inverse kinematics [16].

One more thing should be noted: we deployed an underactuated system for the mobile platform. The DoFs of the mobile platform can be denoted as $\boldsymbol{q}_p = [q_l \; q_r]^{\mathrm{T}}$. The velocities constrainted between $\boldsymbol{p}_p = \left[{}^g x_p \; {}^g y_p \; {}^g \gamma_p\right]^{\mathrm{T}}$ and the intermediate variables $\boldsymbol{h}_p = [v \; \omega]^{\mathrm{T}}$ are expressed as in [17],

$$\dot{\boldsymbol{p}} = \boldsymbol{A}(\boldsymbol{p}) \cdot \boldsymbol{h}_p = \begin{bmatrix} \cos(\theta_p) & \sin(\theta_p) & 0 \\ 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}} \cdot \boldsymbol{h}_p, \tag{1}$$

where $\boldsymbol{h}_p$ represents the linear and angular velocities of the mobile platform, $\boldsymbol{A}$ is a 3×2 transformation matrix, and θ_p is the joint angle of mobile platform.

In order to obtain a solution in joint space, the inverse kinematics of the mobile platform can be written as,

$$\dot{\boldsymbol{q}}_p = \boldsymbol{J}_p \cdot \boldsymbol{h}_p = \begin{bmatrix} 1/r & W/r \\ 1/r & -W/r \end{bmatrix} \cdot \boldsymbol{h}_p, \tag{2}$$

where $\boldsymbol{J}_p$ is a Jacobian matrix, and r and W are the wheel radius and rear tracks of the mobile platform, respectively.

The motions $\left[\alpha_g, \beta_g, {}^m \gamma_p\right]^{\mathrm{T}}$ need to be realized by the manipulator, where ${}^m \gamma_p$ is the relative motion in the mobile platform frame (Σ_p). Hence, inverse kinematics are used for the manipulator to solve the joint variables according to $\boldsymbol{p}_m = \left[{}^p \alpha_m, {}^p \beta_m, z_g, \alpha_g, \beta_g, {}^p \gamma_m\right]^{\mathrm{T}}$. The kinematic models of UR5 and their solutions have been sufficiently explored in an abundance of studies [18]. In this research, iterative methods are applied for the detailed computation of UR5 [19,20]. Moreover, the proposed kinematics model of the manipulator is based on but not limited to UR5.

Furthermore, knowledge of spray distribution is required to generate a motion trajectory. The nozzle starts from the origin of the 3D surface, which is a protruding point if the surface is unfolded. It can be assumed that the spray particles form a cone and that the distribution pattern is roughly circular when the nozzle is perpendicular to the starting point (Σ_s), as shown in Figure 3. The kinematic parameters of the nozzle include the spray angle (θ_s) and the spray radius (R_s). The rate of spray impacts the element division of the disinfection area. In order to make the function of disinfection applicable, a reasonable margin of safety is defined, which can also be used to determine the effect of spraying.
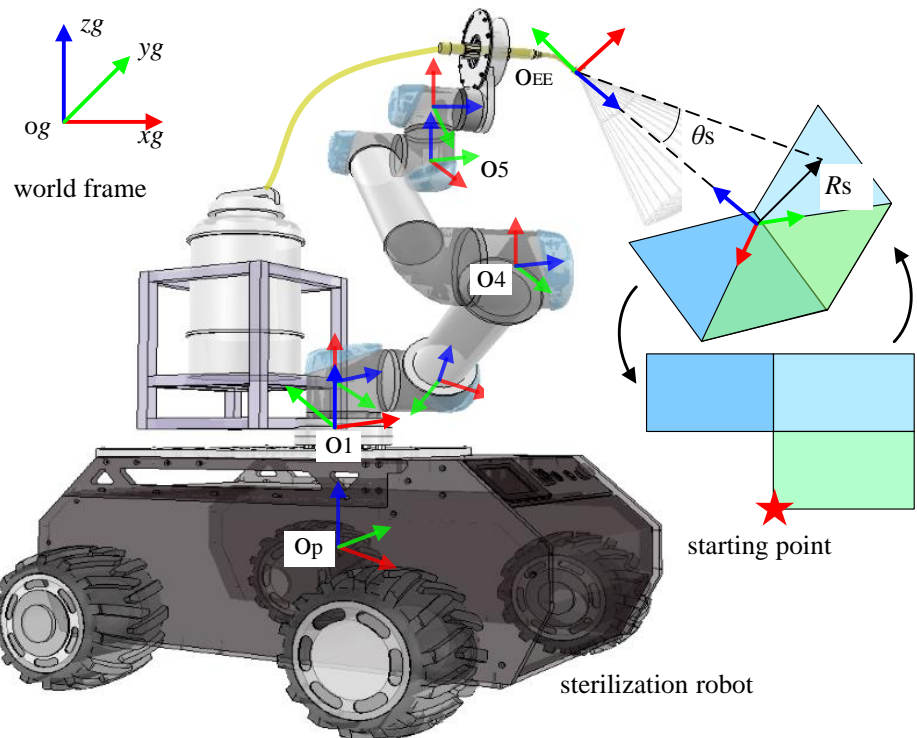
Figure 3. Kinematic analysis of the disinfection robot.

3. Coverage Path Planning

CPP is an important part of the robotic disinfection project. All common environmental areas should be marked and traversed, and the planned path can be obtained by connecting the nodes. Then, the robot follows this path to conduct the disinfection task, ensuring that all possible blind spots are covered with the minimal consumption of disinfectant. The scheme of the proposed CPP algorithm is shown in Figure 4.
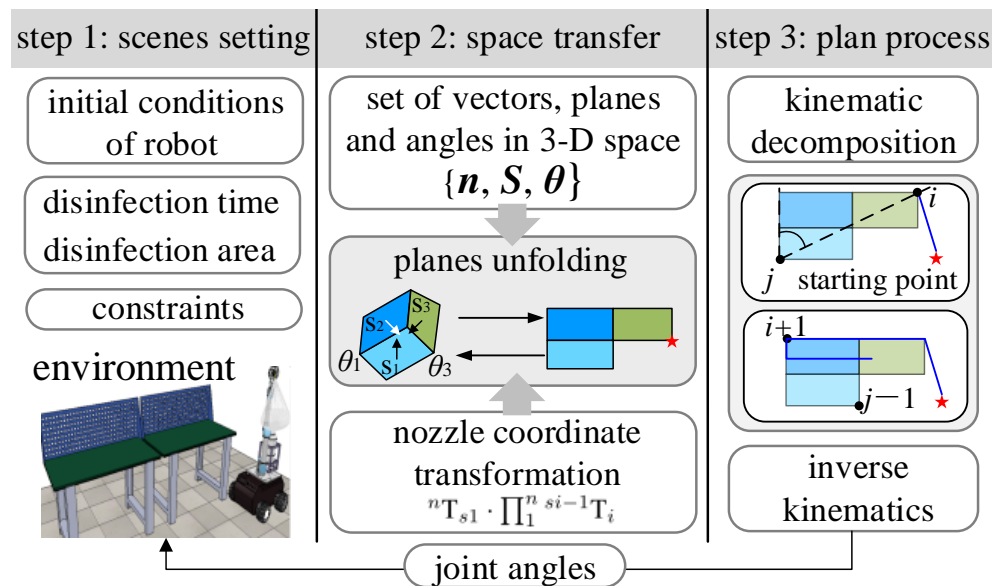

Figure 4. Scheme of coverage path planning.

3.1. Space Transfer

In contrast to traditional path planning methods, each of the areas in the selected environment should be disinfected. Here, space transfer is considered, converting square pixels in 3D space into 2D arrays so that all areas are covered adequately. Each pixel (s_i) is regarded

as an element for set S in 2D space, as shown in Figure 3. Set $\boldsymbol{n} = \{\boldsymbol{n} \mid \vec{n}_{s1}, \vec{n}_{s2}, \ldots, \vec{n}_{sn}\}$ is defined as the normal vectors of each unfolded plane, and set $\boldsymbol{p} = \{\boldsymbol{p} \mid p_1, p_2, \ldots, p_n\}$ represents the vertices of each array in the unfolded plane. Based on set S, the set of angles between two planes is $\boldsymbol{\theta} = \{\theta_{12}, \theta_{13}, \ldots, \theta_{nn}\}$. In practice, the initial and terminal points of the nozzle should be considered and noted as p_i and p_t.

Let $^i T_n$ be a 4×4 homogeneous transformation matrix between the coordinate systems from point p_i to p_n. Therefore,

$$^i T_n = ^i T_1 \cdot ^1 T_2 \ldots ^{n-1} T_n, \tag{3}$$

The geometric parameters can be obtained with defined sets.

After the above process, the CPP algorithm can be viewed as a sequence of robot configurations by which each array is visited at least once. Each array should be evaluated to determine whether or not it is basically reachable by the robot based on its kinematic capability. If the spray does not fully cover the meshed grid, one can tag the corresponding arrays as unreachable, based on which the relative pose and the path are optimized. In summary, after this step, every array will be assigned a corresponding robot configuration. Collision detection and 'reachable'-oriented redundant configuration control are explicitly considered in the CPP algorithm. Thus, the actual coverage path may be lower than estimated in this preprocessing step.

Once the entire surface is covered, the final spray trajectory is obtained and rendered as an input to the end effector to request motion execution on the desired surface.

3.2. Coverage Path Planning

Figure 4 explains how to support the robot with the end effector to get close to the most reachable pixels in its current pose. The optimization problem of the end-effector path can be formulated as a path planning problem, the goal of which is to obtain the minimum path with the corresponding constraints and the robot configurations. After preprocessing, a path sequence (P) can be found to cover the unfolded plane (the initial and terminal points are included) within the minimum time. Based on the CPP algorithm with the rotating caliper algorithm, an unfolding plan can be introduced as $Q = \{V, E\}$, where $V = \{1, 2, \ldots, n\}$ is the set of vertices, and $E = \{(1, 2), \ldots, (n, 1)\}$ is the set of edges. Then, the nozzle, as defined by the search pattern, follows a path to cover the area (Q), where each point in the path will be reached by the nozzle moving in straight lines. All points in the path can be denoted as $P = \{p \mid p_i, p_1, p_2, \ldots, p_n, p_t\}$.

Using rotating calipers, antipodal pairs i and j ($i, j \in V$) are two points on the surface of a polygon that are exactly opposite each other. The idea of the method is to compute all the antipodal pairs, find the best path for each antipodal pair, and then select the path with the lowest cost in combination with the takeoff and landing points [8]. The caliper is roated clockwise until it touches the edge then, it is rotated counter-clockwise until it touches a second edge. An *angle*() function is then defined to compute the angle swept out by a line as it rotates from its position parallels. Next, both options are measured to determine the minimum width of the formed polygon. In application, one can define vertex i close to the initiation point. Then, the algorithm performs a sequence of decisions and procedures that do not depend on the number of vertices. Based on the antipodal pair and optimized angle, a path can be computed by the $2Dpath()$ function using a back-and-forth pattern. Finally, the plan (S) should be folded in 3D space based on an inverse kinematic solution using Equation (3). More details about its implementation are shown in Algorithm 1.

Algorithm 1 Coverage path planning algorithm

Input:
normal vectors of unfolded plane
$\boldsymbol{n} = \{\boldsymbol{n} \mid \vec{n}_{s1}, \vec{n}_{s2}, \dots, \vec{n}_{sn}\}$;
the sizes of each unfolded plane $\boldsymbol{S} = \{\boldsymbol{s} \mid s_1, s_2, \dots, s_n\}$;
plane angle $\boldsymbol{\theta} = \{\theta_{12}, \theta_{13}, \dots, \theta_{nn}\}$;
vertices of unfold plan $\boldsymbol{p} = \{\boldsymbol{p} \mid p_1, p_2, \dots, p_n\}$;
safety margin h and spray radius R_s;
antipodal pair (i, j);
Output:
waypoints $\boldsymbol{p}$ in 3D space;

1: space transfer with $\boldsymbol{n}, \boldsymbol{S}, \boldsymbol{\theta}$ using Equation (3);
2: define the rotation angle function $angle()$ in clockwise direction;
3: define the distance function $dist()$ between two points by using Euclidean distance formula;
4: define the back and forth path function $2Dpath()$ based on antipodal pair (i, j) and optimized angle;
5: **if** $angle(i, j) < angle(j, i)$ **then**
6:     $k \leftarrow j; l \leftarrow i$;
7: **else**
8:     $k \leftarrow i; l \leftarrow j$;
9: $\phi \leftarrow angle(k, l) - \pi$;
10: $\varphi_k \leftarrow angle(k - 1, k) - \pi; \varphi_l \leftarrow angle(l - 1, l) - \phi$;
11: **if** $\varphi_k < \varphi_l$ **then**
12:     $m \leftarrow k - 1; n \leftarrow l$;
13: **else**
14:     $m \leftarrow l - 1; n \leftarrow k$;
15: **if** $dist(k, l) < dist(m, n)$ **then**
16:     $\boldsymbol{p} = 2Dpath(k, k + 1)$;
17: **else**
18:     $\boldsymbol{p} = 2Dpath(m + 1, m)$;
19: calculate the waypoints in 3D space based on inverse solution of Equation (3);
20: calculate the joint angle of mobile platform using Equation (1);
21: calculate the joint angle of manipulator in numerical iteration method with motions $\left[^p\alpha_m, {}^p\beta_m, z_g, \alpha_g, \beta_g, {}^p\gamma_m \right]^{\mathrm{T}}$.

---

4. Simulation and Experiment

The application of the designed disinfection robot and the proposed CPP algorithm is verified in this section. For the disinfection test in 3D space, we assume that the spray is uniformly distributed, and the sanitizing agents are sprayed into the appointed area every second. Since pathogens can easily be found in common environments (especially on tables and door handles that are frequently touched by humans), the estimated best spraying effect can be achieved by aiming the geometric center point of the spray nozzle at least once. Two simulations and one experiment are carried out to verify the feasibility of the robot and the proposed algorithm. The first simulation focuses on the rationality and capability of path planning for key points in diverse scenarios. The second simulation aims to verify the feasibility of the robot via a disinfection task in a simulation environment. Here, the robot is aimed to disinfect a workbench in a simulation, as shown in Figure 5. For the experiment, the scenario is identical to that in the second simulation. The experimental objective is to test the practical and theoretical performance of the robot. Corresponding videos are provided as supplementary material for this research.
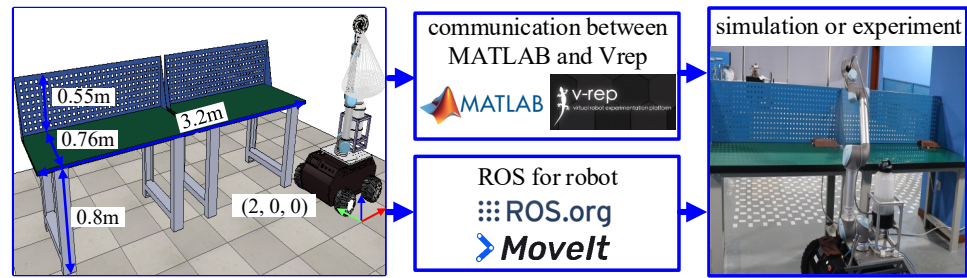
Figure 5. Scheme of the simulation and experiment.

4.1. Simulation and Experimental Settings

Vrep and MATLAB are used as tools for the simulations due to their close-to-physical environments and data processing abilities. The simulation process is conducted as follows. The proposed algorithm runs in MATLAB, and the implementation, and the feedback of the data are performed in Vrep with dynamic properties. First, the data and the algorithm are processed in MATLAB. Secondly, the communication for ROS nodes is established through the programming interface of the application. Then, defined outputs are continuously implemented, while the data results of the proposed algorithm are updated. Within the cosimulation with MATLAB and Vrep, the physical engine and the simulation steps are selected as Bullet (a physical engine) 2.73 and 50 ms, respectively.

For the experiment, the whole control system architecture is realized based on ROS1. A distributed multicomputer communication network for the multirobot system is adopted using the TCP/IP protocol [14]. The ROS master runs on a remote computer (i7-8700 CPU and 8GB RAM), with which the operator can control the robot in a noninfectious environment. For the robot, an Intel NUC (i7-8559 CPU and 16GB RAM) is used as a local workstation. The control function based on the proposed CPP algorithm is integrated and executed in the local workstation. The communication between the mobile platform and the manipulator (UR5) is based on a feature package named ROS MoveIt. They can be connected using twisted pairs. The specific operations of the simulation and the experiment are shown in Figure 5.

To better illustrate the disinfection of the robot and the proposed algorithm, the experimental area is divided into two categories: horizontal groups and vertical groups. We should also point out that a future version of the motion planner will include other parameters such as irregular regions and moving objects. The kinematic and dynamic parameters of the disinfection robot are listed in Table 2.

**Table 2.** Physical parameters of the designed disinfection robot.

Mobile platform	Size (L * W * H)	Weight	Load
	0.88m * 0.68m * 0.45 m	60 kg	100 kg
Manipulator	Weight : [2.58 1.55 1.20 0.52 0.41 0.09] (kg)		
	a_{DH} : [0 0.61 0.57 0 0 0] (m)		
	d_{DH} : [0.12 0 0 0.16 0.12 0.09] (m)		
	α_{DH} : [$\pi/2$ 0 0 $\pi/2$ $-\pi/2$ 0] (rad)		
	θ_{DH} : [0 $-\pi/2$ 0 $-\pi/2$ 0 0] (rad)		
Gripper	Weight	Clamping range	Clamping shape
	0.45 kg	$0 \sim 60$ mm	Arbitrary

The experimental process can be summarized in three steps. First, all the features of motion and path planning are packed as ROS packages. Secondly, communication tests and ROS nodes are initialized. Finally, the ROS packages of motion and path planning are executed with other necessary ROS nodes on either the local or master computers.

4.2. Validation of the CPP Algorithm

For this simulation, the scenarios are diversified, including standard, arbitrary, and concave polygons, while the antipodal pair (i, j) is randomly selected. The size of scenarios varies from 4 m to 20 m and is used to describe the unfolded shapes of the disinfection areas as much as possible. Since the nozzle is assumed to spray uniformly, the scanning radius is defined as 0.5 m in the simulation.

Figure 6 shows the results of the CPP algorithm, in which the black solid line represents the disinfection area, and the solid circles are the initial and terminal points. Some of the points are determined by calculation, while the rest are randomly selected. The lines are the end coverage paths, which are the final results after planning. Based on the proposed algorithm, the path covers every corner to ensure the full disinfection of the surfaces; key points in this path are important for the robot's motion. It can be observed that the proposed algorithm achieves the minimum redundant paths in all environments when they complete the same amount of coverage tasks. This observation is consistent with the results of the theoretical analyses in the third section.
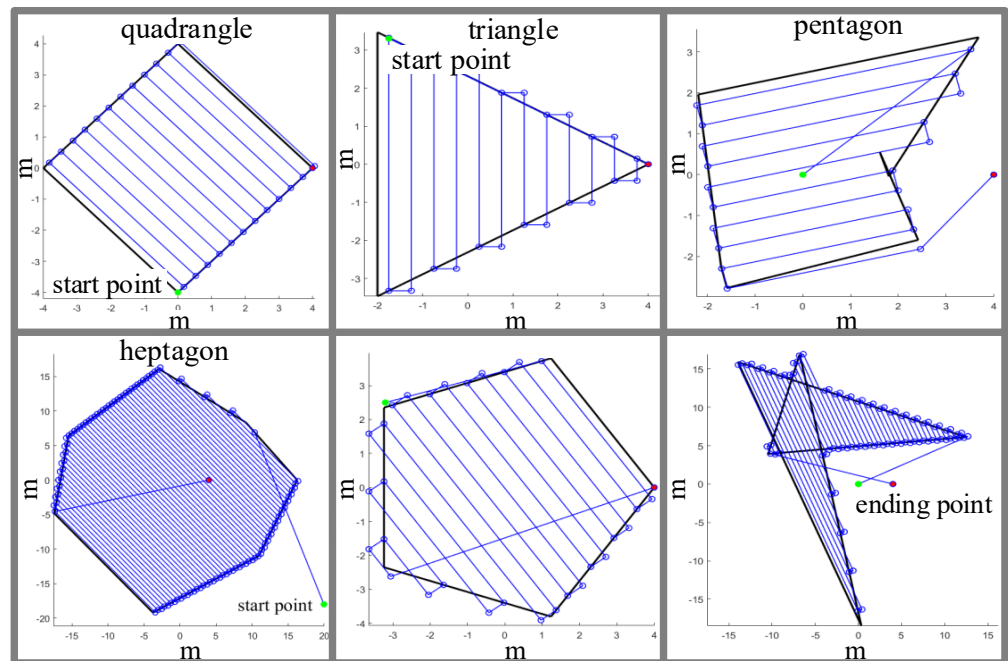
Figure 6. Coverage path planning in different environments.

4.3. Validation for Robot Disinfection in Simulation

To demonstrate the applicability of the robot and the proposed algorithm for epidemic prevention, a coverage path task is performed on a workbench. The overall size of the workbench is 3 m × 0.8 m × 1.5 m (length, width, and height, respectively). These parameters are input into the algorithm as scene setting. The initial generalized coordinates of the robot and the workbench are given in the left subfigure of Figure 5. All the initial values of joints ($\dot{q}$) are defined as zeros.
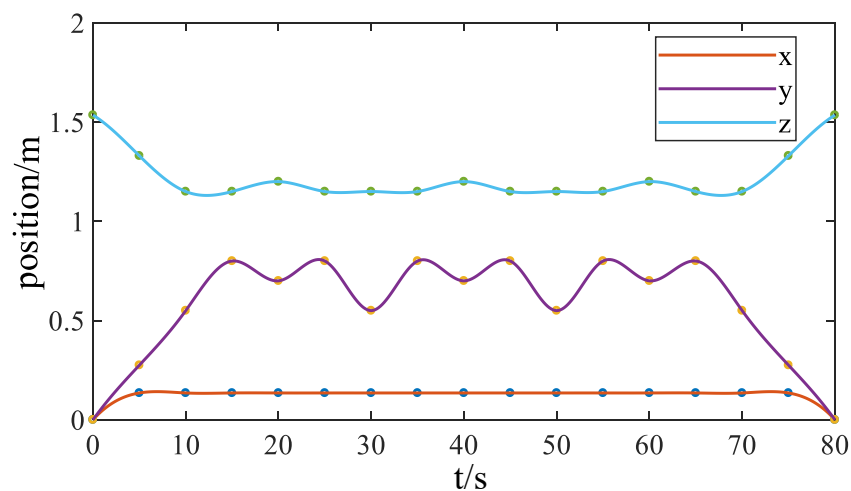
Figure 7 shows snapshots of the simulated disinfection. The snapshot at t = 5 s (top left) shows the nozzle as it begins to spray the workbench. The snapshot at t = 20 s (top right) depicts the robot as it is nearly finished spraying the first workbench. The snapshot at t = 50 s (bottom left) shows the robot as it begins to disinfect the second workbench. The snapshot at t = 80 s (bottom right) represents the accomplished disinfection. More details of the simulation can be found in the attached video. First, the robot and the proposed algorithm achieve path coverage in 3D space, and the nozzle follows a trajectory planned by the proposed algorithm. Secondly, it is feasible and in accordance with the motion plan mentioned in the third section. The end effector can easily track the trajectory of the relatively complex and mobile platform. On one hand, if we remove the data of the path

of the end effector as the key point, the CPP algorithm is more practical and more easily implemented. On the other hand, the higher the altitude of the mobile platform, the larger the path losses and the more robot hovering energy required. Thirdly, the nozzle is flexible to operate due to real-time communication, execution, and feedback between MATLAB and Vrep with minimal time delay.
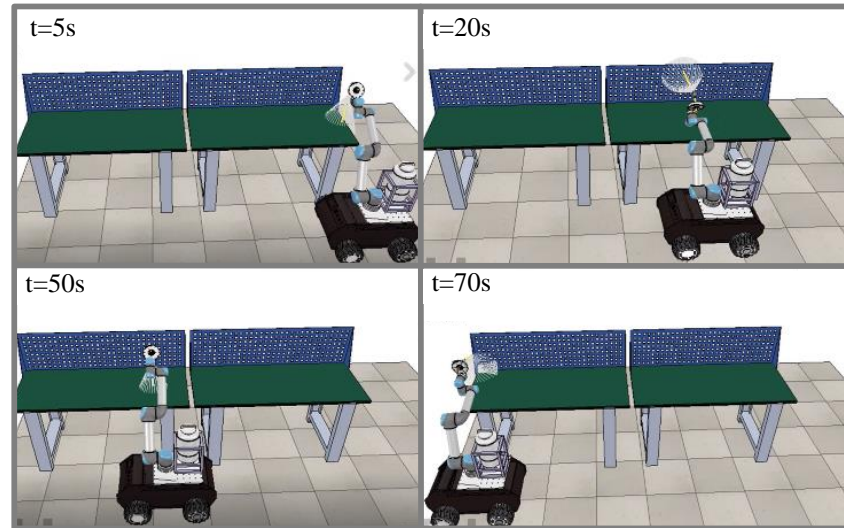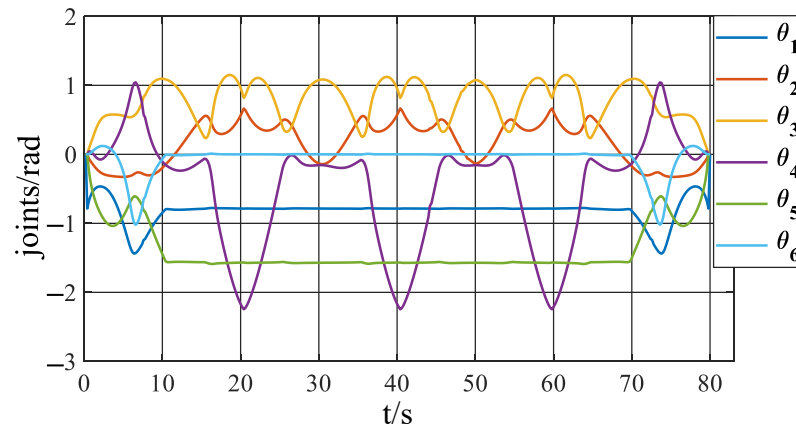

Figure 7. Disinfection in the simulation environment.

In order to express the performance of the end effector, the posture along the x_p^e, y_p^e, and z_p^e directionsis presented in Figure 8; the trajectories are smooth during the whole simulation. At the beginning, the robot adjusts its height and direction to stay in a reasonable range within the workbench. Hence, the trajectory on the z_p^e axis changes regularly. During the disinfection process, the trajectory is regular and smooth. The joint trajectories of UR5 are shown in Figure 9, with a similar regularity in joint space. The disinfection robot has fewer constraints on its movement compared to the situation in the first simulation. Therefore, the algorithm can choose among a wide variety of trajectories, enabling separate evaluation of its capability. The goal of this simulation is to evaluate the disinfection performance of the optimized trajectory planning method described in the previous sections.


Figure 8. Posture of the nozzle along the x_p^e, y_p^e, and z_p^e directions.

Figure 9. Joints of the manipulator in the simulation.

4.4. Experimental Validation of Robot Disinfection

In order to reproduce the properties of the disinfection robot in a physical environment, the configuration and parameters of the experiment are identical to those in the simulation. Figure 10 shows snapshots of the experimental disinfection. The snapshot at t = 5 s (the top left) shows the nozzle as it begins to spray on the workbench. The snapshot at t = 20 s (the top right) shows the robot as it is nearly finished spraying on the first workbench. The snapshot at t = 50 s (the bottom left) shows the robot as it begins to disinfect the second workbench. The snapshot at t = 70 s (the bottom right) represents the accomplished disinfection. More details of the experiment can be found in the attached video.
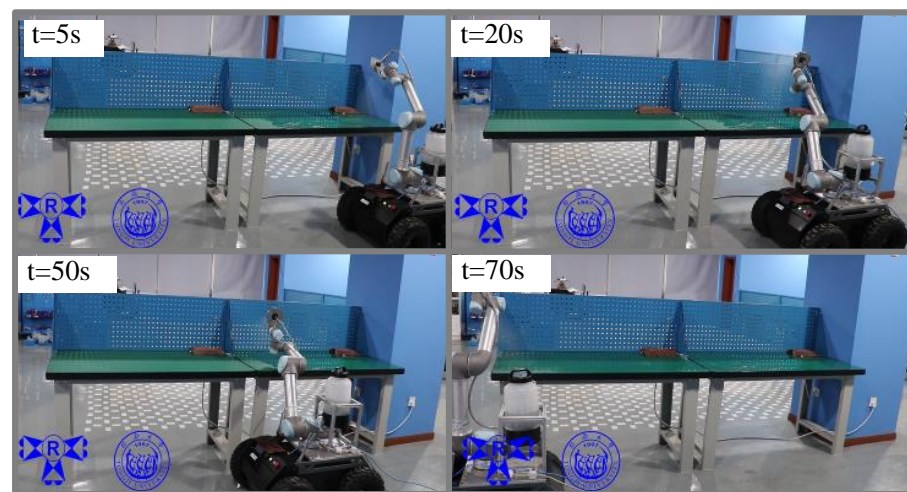
Figure 10. Disinfection in the experimental environment.

In this experiment, the robot and the proposed algorithm can achieve path coverage in real-world 3D space. Validation is conducted by seeking optimal trajectories on the horizontal and vertical surfaces, as shown in Figure 10. In order to express the performance of the end effector, the posture along the x_g, y_g, and z_g directionsis recorded in Figure 11; the trajectories are similar to those in the simulation. At the beginning, the robot adjusts its height and direction to stay in a reasonable range within the workbench. Hence, the trajectory on the z_g axis changes less smoothly. During the disinfection process, the trajectories are regular and smooth. The joint trajectories of UR5 are shown in Figure 12 with a similar regularity in joint space. The characteristics of the spray nozzle in the experiment have more influence on the performance parameters, such as actual wireless communication, spray homogenization, and thickness. We should also point out that this algorithm can handle a variety of surfaces in 3D space that humans usually touch. The

results prove that the proposed disinfection robot is more suitable for spraying tasks in 3D space than regular (move-based) disinfection robots.
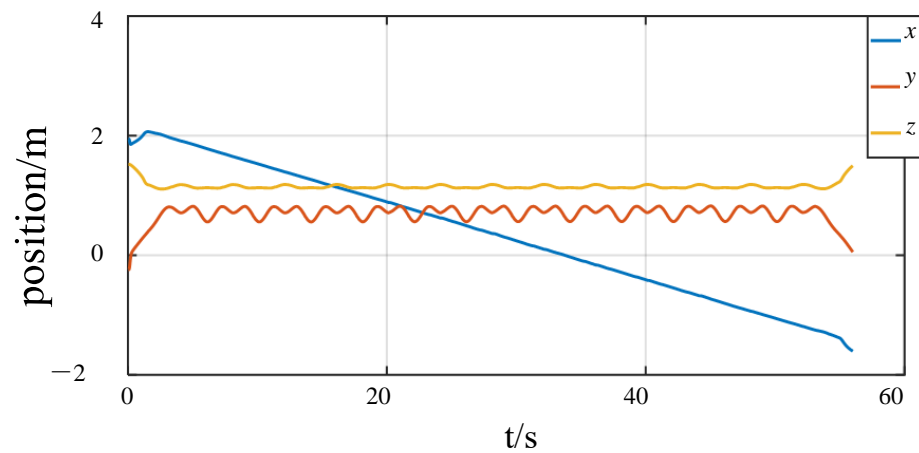

Figure 11. Posture of the end effector in the x_g, y_g, and z_g directions.
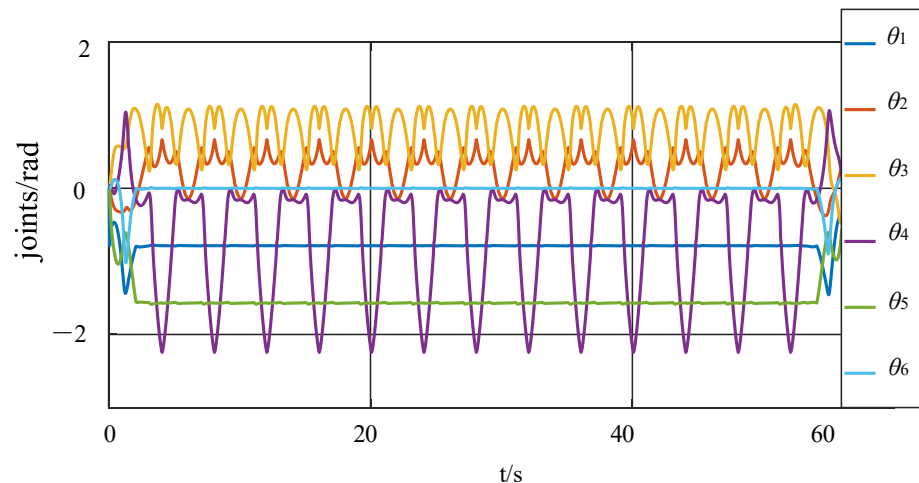

Figure 12. Joints of the manipulator in the experiment.

5. Conclusions

This research proposes an intelligent disinfection robotic system for the disinfection of common environments that humans often access. The robotic system utilizes multiple technologies, including a structural design, communication system, disinfection instrument framework, ROS1 and CPP, etc. The mobile manipulator can grant the disinfection robot more flexibility than common disinfection robots based on a mobile platform, enabling the determination of optimized trajectories that, although slightly complex, result in a better disinfection effect. Planning a feasible path is an important component of this study, for which the CPP algorithm is used to generate a rational coverage path in 3D environments. Simulations and an experiment were conducted to verify the feasibility and effectiveness of the robotic system. The proposed disinfection strategy enables the robot to automatically complete disinfection tasks throughout any target environment, thereby ensuring human safety and reducing labor required for manual disinfection. For future research, the robot can be made more intelligent and multifunctional, and more common environmental scenarios should be verified.

6. Patents

Qirong Tang, Xinyi Chen, Pengjie Xu. clamping mechanism and clamping device. Invention patent CN202111405539.1, 2022.12.27.

References

1. Yang, G.; Nelson, B.; Murphy, R.; Choset, H.; Mcnutt, M. Combating COVID-19 the role of robotics in managing public health and infectious diseases. *Sci. Robot.* **2020**, *5*, 1–2. [CrossRef] [PubMed]
2. Zemmar, A.; Lozano, A.; Nelson, B. The rise of robots in surgical environments during COVID-19. *Nat. Mach. Intell.* **2020**, *2*, 566–572. [CrossRef]
3. Tamantini, C.; Luzio, F.; Cordella, F.; Pascarella, G.; Zollo, L. A robotic health-care assistant for covid-19 emergency: A proposed solution for logistics and disinfection in a hospital environment. *IEEE Robot. Autom. Mag.* **2021**, *28*, 71–81. [CrossRef]
4. Zhao, Y.; Huang, H.; Chen, T.; Chiang, P.; Weng, W. A smart sterilization robot system with chlorine dioxide for spray disinfection. *IEEE Sensors J.* **2021**, *21*, 22047–22057. [CrossRef]
5. Boyce, J. Modern technologies for improving cleaning and disinfection of environmental surfaces in hospitals. *Antimicrob. Resist. Infect. Control.* **2016**, *5*, 112–123. [CrossRef] [PubMed]
6. Ortega, R.; Astolfi, A.; Bastin, G.; Rodriguez, H. Stabilization of food-chain systems using a port-controlled Hamiltonian description. In Proceedings of the 2000 American Control Conference, Chicago, IL, USA, 28–30 June 2000; pp. 2245–2249.
7. SOMATIC: Higher Quality Cleaning. 8 April 2021. Available online: http://getsomatic.com (accessed on 7 April 2023).
8. Gomez, J.V.; Melchor, M.; Lozada, J.H. Optimal coverage path planning based on the rotating calipers algorithm. In Proceedings of the 2017 International Conference on Mechatronics, Electronics and Automotive Engineering, Morelos, Mexico, 26–29 November 2017; pp. 140–144.
9. Han, G.; Zhou, Z.; Zhang, T.; Wang, H.; Guizani, M. Ant-colony-based complete-coverage path-planning algorithm for underwater gliders in ocean areas with thermoclines. *IEEE Trans. Veh. Technol.* **2020**, *69*, 8959–8971. [CrossRef]
10. Hong, H.; Shin, W.; Jieun, O.; SunWoo, L.; TaeYoung, K.; WooSub, L.; JongSuk, C.; Suh, S.; Kim, K. Standard for the quantification of a sterilization effect using an artificial intelligence disinfection robot. *Sensors* **2021**, *21*, 7776. [CrossRef]
11. Tiseni, L.; Chiaradia, D.; Gabardi, M.; Solazzi, M.; Frisoli, A. UV-C mobile robots with optimized path planning: Algorithm design and on-field measurements to improve surface disinfection against sars-cov-2. *IEEE Robot. Autom. Mag.* **2021**, *25*, 59–70. [CrossRef]
12. Bähnemann, R.; Lawrance, N.; Chung, J.J.; Pantic, M.; Siegwart, R.; Nieto, J. Revisiting Boustrophedon Coverage Path Planning as a Generalized Traveling Salesman Problem. In *Field and Service Robotics: Results of the 12th International Conference*; Springer: Singapore, 13 January 2021; pp. 277–290.
13. Ren, Y.; Sosnowski, S.; Hirche, S. Fully distributed coordinated cooperation for networked uncertain mobile manipulators. *IEEE Trans. Robot.* **2020**, *36*, 984–1003. [CrossRef]
14. Marino, A.; Pierri, F. A two stage approach for distributed cooperative manipulation of an unknown object without explicit communication and unknown number of robots. *Robot. Auton. Syst.* **2018**, *103*, 122–133. [CrossRef]
15. Li, K.; Boonto, S.; Nuchkrua, T. On-line self tuning of contouring control for high accuracy robot manipulators under various operations. *Int. J. Control. Autom. Syst.* **2020**, *18*, 1818–1828. [CrossRef]
16. Chen, L.; Yang, Q.; Li, C.; Ma, G. Euler-Lagrange system, formation control, mobile robots, multi-agent systems, multi-robot systems. *Int. J. Control. Autom. Syst.* **2021**, *19*, 1740–1750. [CrossRef]
17. Azzabi, A.; Nouri, K. Design of a robust tracking controller for a nonholonomic mobile robot based on sliding mode with adaptive gain. *Int. J. Adv. Robot. Syst.* **2021**, *18*, 1–18. [CrossRef]

18. Ferraguti, F.; Landi, C.; Costi, S.; Bonfè, M.; Fantuzzi, C. Safety barrier functions and multi-camera tracking for human-robot shared environment. *Robot. Auton. Syst.* **2020**, *124*, 1–19. [CrossRef]

19. Kebria, P.; Al-Wais, S.; Abdi, H.; Nahavandi, S. Kinematic and dynamic modelling of UR5 manipulator, In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016; pp. 9–12.

20. Weyrer, M.; Brandsttter, M.; Husty, M. Singularity avoidance control of a non-holonomic mobile manipulator for intuitive hand guidance. *Robotics* **2019**, *8*, 14. [CrossRef]