

# Automated Route Planning from LiDAR Point Clouds for Agricultural Applications <sup>†</sup>

Fabian Theurl , Christoph Schmied , Eva Reitbauer  and Manfred Wieser

Institute of Geodesy, Graz University of Technology, 8010 Graz, Austria; schmied@tugraz.at (C.S.); eva.reitbauer@tugraz.at (E.R.); manfred.wieser@tugraz.at (M.W.)

\* Correspondence: fabian.theurl@tugraz.at

<sup>†</sup> Presented at the European Navigation Conference 2023, Noordwijk, The Netherlands, 31 May–2 June 2023.

**Abstract:** This paper develops an algorithm to compute optimal routes for an autonomous compost turner. In commercial composting, the material to be composted is piled up in large heaps called windrows and turned regularly by compost turners. The environment at the composting site is constantly changing, as the locations of the windrows change with each turning procedure. Therefore, we propose a novel method that automatically computes routes on a composting plant from LiDAR data. The LiDAR is mounted on the compost turner together with a dual-antenna GNSS receiver, an IMU, and rotary encoders. An extended Kalman filter is used to obtain the vehicle's pose. Through direct georeferencing, a global point cloud is obtained. The routing algorithm crops, segments, and filters the point cloud until the points along the ridge of each windrow remain. These points are used to compute the optimal routes along each windrow. Furthermore, a user can select the windrows which need to be turned and the algorithm then computes the most efficient path for the compost turner, which also includes the passages between the windrows. The method was tested within a simulation environment using a 3D model of the composting site. The results show that the algorithm detects the windrows and computes the routes with sufficient accuracy for autonomous compost turning.

**Keywords:** mobile mapping; routing; point cloud processing; LiDAR; simulation; digital twin; 3D model; autonomous agriculture; multi-sensor fusion; Kalman filtering



**Citation:** Theurl, F.; Schmied, C.; Reitbauer, E.; Wieser, M. Automated Route Planning from LiDAR Point Clouds for Agricultural Applications. *Eng. Proc.* **2023**, *54*, 54. <https://doi.org/10.3390/ENC2023-15448>

Academic Editors: Tom Willems and Okko Bleeker

Published: 29 October 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In 2020, the European Commission introduced a circular economy action plan [1] as part of its European Green Deal. Within the Waste Framework Directive [2], it specified the target that by the year 2035, a minimum of 65% of municipal waste should be reused or recycled. As bio-waste amounts to 34% of all municipal waste [3], recycling bio-waste plays a key role in meeting the EU's recycling targets.

The most common recycling techniques for bio-waste are composting and anaerobic digestion. According to [4], over 90% of the bio-waste that is separately collected in Europe is composted. Therefore, composting can be considered the most important bio-waste recycling technique in Europe.

In composting, the organic material is turned into compost, a biochemically stable product containing microorganisms that can improve soil properties, such as organic matter content, water retention capacity, and pH buffer capacity [3].

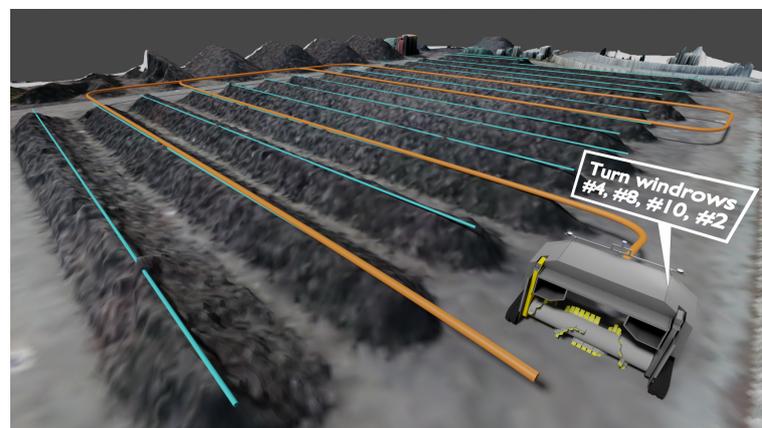
The most common method of commercial composting is windrow composting, where the organic material is stacked into long heaps that are turned mechanically by compost turners to promote aerobic microbial activity, moisture release, heat removal, and aeration. However, the job of turning windrows is monotonous and the operator of a compost turner is exposed to high temperatures, water vapor, and gases. Therefore, it is desirable to develop autonomous compost turners.

In recent years, a research team at Graz University of Technology has focused on the development of a navigation module for autonomous compost turners. A concept for an automatic steering system was first presented in [5]. In [6], tests are conducted to select a suitable set of navigation sensors. Two different multi-sensor fusion architectures for fusing GNSS, IMU, wheel odometry, and stereo camera are presented and evaluated in [7]. Ref. [8] presents an odometry model specifically tailored to tracked compost turners that allows for bridging GNSS outages. In [9], a real-time positioning module to estimate the position, velocity, and attitude of a compost turner is developed and tested in an automatic steering system at a composting plant.

In [10], a concept for autonomous compost plant management is proposed. An autonomous plant management not only requires an automatic steering system that steers the compost turner through a single windrow, but requires more complex maneuvers such as turning and driving from one windrow to the next. To achieve these tasks in an automated manner, a complex routing algorithm is required. However, no study has yet focused on routing algorithms for autonomous compost turners. This is exactly where this paper comes in: it develops an algorithm to compute optimal routes from a mobile mapping system mounted on the compost turner.

A major challenge for developing routing algorithms for a composting plant is the constantly changing environment, as the locations of the windrows change with each turning procedure. Therefore, we propose a novel method that automatically computes routes on a composting plant from LiDAR data. In this paper, the concept is presented and tested within a simulated composting plant environment.

Figure 1 shows an overview of the simulated composting plant where several windrows can be seen. The aim of the routing algorithm is to first detect each separate windrow (cyan) in an initialization phase. Afterward, an operator can specify which windrows shall be turned before the compost turner computes an optimal route (orange) through them.



**Figure 1.** Composting plant in the simulation environment: The routing algorithm first computes the optimal path through each windrow (cyan) and then the final path along which the autonomous compost turner is steered (orange).

Path planning in environments that are observed with a LiDAR scanner is a common task in mobile robotics. For planning global paths, commonly used algorithms are the A\* algorithm, which was first published in [11] and then constantly improved (e.g.: [12–14]), and the Dijkstra algorithm [15]. For high-level planning that considers the robot configuration, a local path planner is used. Examples of such local planners are the Dynamic Window Approach algorithm [16] and the Timed Elastic Band (TEB) algorithm [17]. All planners, global and local, use cost maps to define the traversability of different areas within the map.

This paper uses the Dijkstra algorithm as a global and the TEB algorithm as a local planner. However, the main difference to conventional routing applications is that windrows, which would be classified as obstacles by conventional algorithms, are in our case traversable and part of the desired route. This means that each cluster of points within

the point cloud that represents such a windrow must not be classified as an obstacle but must be processed to determine the optimal route through it. To the authors' knowledge, no similar system has yet been developed.

## 2. Materials and Methods

This chapter describes the steps of the routing algorithm and its validation in the simulation environment. Section 2.1 gives an overview of the developed software and the used software packages. Section 2.2 outlines how the global point cloud of the composting plant is obtained from LiDAR data. Section 2.3 describes how the global point cloud is segmented into individual windrow point cloud clusters. Note that point cloud clusters will be referred to as *clusters* throughout this paper. Section 2.4 explains how the individual routes through each windrow cluster are calculated and Section 2.5 outlines how the final route for the compost turner is computed using input from an operator. Section 2.6 describes the simulation environment that is used to validate the developed algorithm.

### 2.1. Software Development

The software for the routing algorithm is developed in C++. For the point cloud processing operations, the openly available Point Cloud Library [18] is used. The Robot Operating System (ROS) [19] provides the appropriate framework for the development of the robotic application. The ROS navigation stack offers the 2D navigation capabilities needed for route planning and following.

### 2.2. Obtaining the Global Point Cloud from the Windrow Area

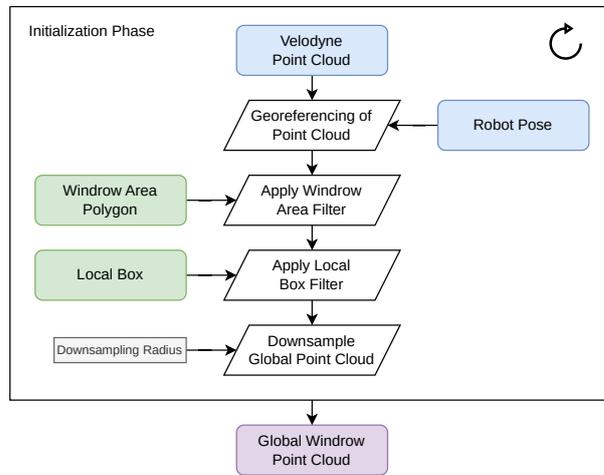
In the first step, an up-to-date global point cloud of the composting plant must be generated. This cloud is acquired in an initialization phase where the compost turner is steered manually around the windrows at the composting plant. For this task, a simulated Velodyne Ultra Puck LiDAR scanner was used. Given the scanner's limited vertical field of view of 40°, it is mounted on top of the compost turner with a forward tilt of 15° to better cover the area in front of the machine.

The full procedure to incrementally build the global point cloud in the initialization phase is shown in Figure 2. Each incoming LiDAR point cloud must first be georeferenced to be in the same coordinate frame as the global point cloud. This is achieved through direct georeferencing using the position and attitude results (*Robot Pose*) from a previously developed Error State Kalman Filter (ESKF). The filter fuses the odometry data of the compost turner with data from an Inertial Measurement Unit (IMU) and a Real-Time Kinematic (RTK) dual antenna Global Navigation and Satellite System (GNSS) receiver to calculate the state vector of the robot in real-time. For a detailed description of the positioning module, the reader is referred to [9].

To keep the size of the global point cloud as small as possible in order to reduce computational load, several further operations are performed:

- **Windrow Area Filter:** The windrows are always located within the same area on the composting plant. Therefore, the extent of this area is defined using a confining polygon (*Windrow Area Polygon*) created from known 2D coordinates. All points outside this polygon are simply filtered out during the initialization.
- **Local Box Filter:** Points are only added to the global point cloud when they are within a certain vertical range from the ground and within a certain horizontal distance from the LiDAR. The vertical range ensures that ground points are removed. The horizontal distance ensures that points that are further away from the LiDAR—and, therefore, have a lower accuracy—are not included.
- **Downsampling:** The constant addition of the current LiDAR point cloud to the global cloud leads to areas that have a very high point density. In a downsampling step, the total amount of points can be reduced, especially in these dense areas, by making sure that only one point is kept within a specified *Downsampling Radius*.

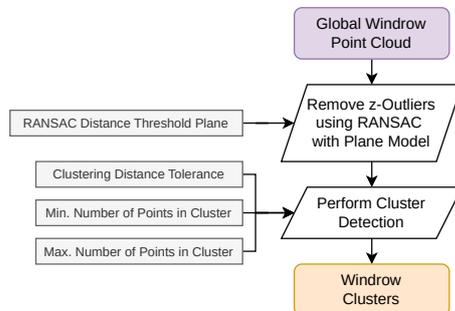
After the initialization phase, the *Global Windrow Point Cloud* contains all points that belong to windrows and provides the basis for extracting the individual windrow clusters.



**Figure 2.** Incremental procedure to build the global windrow cloud during an initialization phase.

### 2.3. Extraction of the Windrow Clusters

From the *Global Windrow Point Cloud*, the individual windrow clusters can now be extracted. An overview of the clustering process is given in Figure 3. In the first step, an outlier removal using Random Sample Consensus (RANSAC) is performed, since outliers can negatively influence the clustering result. Therefore, RANSAC with a planar model is used to estimate the best-fitting plane through all points. Points further than a specified *Distance Threshold* away from this plane, i.e., the outlier points, are removed.

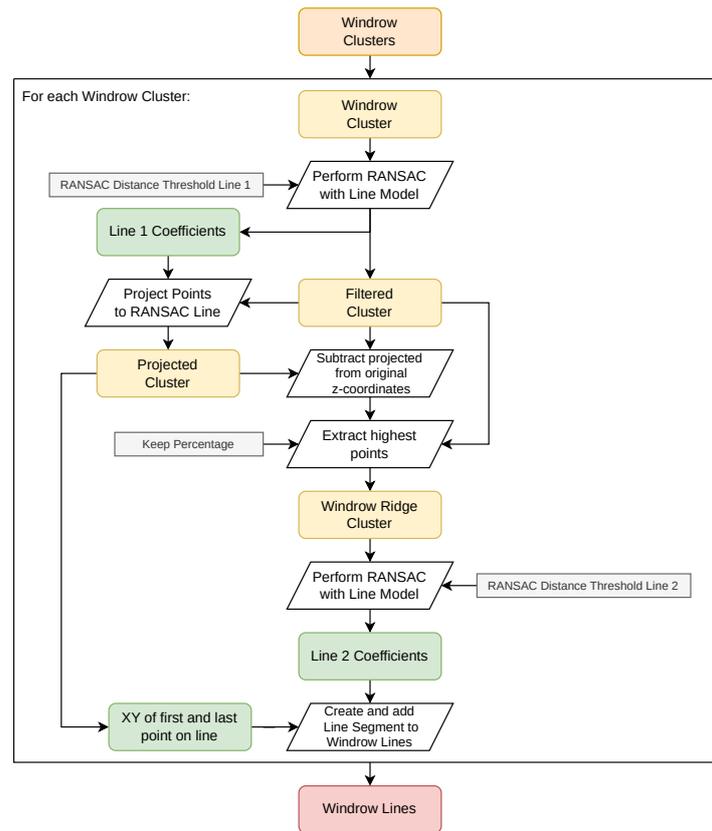


**Figure 3.** Filtering and clustering of the global point cloud.

After the outlier removal, the clustering is done using Euclidean Clustering provided by the Point Cloud Library (PCL). The *Clustering Distance Tolerance* parameter specifies the maximum distance between points to be considered part of the same cluster. Additionally, the cluster size limits (*minimum and maximum number of points* within a single cluster) can be set. These parameters depend on the density of the point cloud. If the *Downsampling Radius* (see Section 2.2) is large, a larger *Clustering Distance Tolerance* and lower cluster size limits are necessary, and vice versa. The resulting clusters each represent one windrow.

### 2.4. Finding the Optimal Route through each Windrow Cluster

In the next step, each *Windrow Cluster* is analyzed and processed to find the optimal route for the compost turner through the windrow. The optimal route follows the ridge of each windrow and thus requires the extraction of the ridge points from the *Windrow Cluster*. A simple line is estimated, as it guarantees a relatively straight windrow after the compost turner finishes moving through the windrow. In Figure 4, the full process of finding this optimal line through each windrow is outlined. The individual steps are explained in the following.



**Figure 4.** Overview of the processing steps needed to obtain the optimal route through a Windrow Cluster.

First, a RANSAC line-fitting algorithm is used to obtain the *Line Coefficients* for the best-fitting line through the cluster points and to further eliminate outliers (*Filtered Cluster*). Projecting the *Windrow Cluster* to the RANSAC line (*Projected Cluster*) assigns every point  $i$  a corresponding  $z$ -coordinate on the line ( $z_{line}^i$ ). Then,  $z_{line}^i$  is subtracted from the original  $z$ -coordinate,  $z_{original}^i$ , of the point:

$$z_{rel}^i = z_{original}^i - z_{line}^i \tag{1}$$

This is performed for all  $n$  points. The resulting set  $Z_{rel} = \{z_{rel}^i\}_{i=1}^n = \{z_{rel}^1, z_{rel}^2, \dots, z_{rel}^n\}$ , which represents the  $z$ -coordinates of the points with respect to the line, is distributed around zero. From this set, the points with the highest  $z$ -coordinates relative to the line can be extracted. The *Keep Percentage* parameter specifies the percentage of points extracted. These points then represent the ridge of the windrow (*Windrow Ridge Cluster*).

Finally, a second RANSAC line-fitting algorithm can be applied to obtain the coefficients for the optimal line through the *Windrow Ridge Cluster*. From the *Projected Cluster*, the first and last points on the line are extracted. These points are used together with the *Line Coefficients* from the second RANSAC to obtain the start and endpoints of a line segment—the optimal route through the windrow.

### 2.5. Route Planning Algorithm for the Compost Turner

After determining the optimal route through each windrow (*Windrow Lines*), the routing algorithm for the global path at the composting plant can be developed. The aim is to find a route through user-selected windrows, which takes maneuvers on the composting plant into account. Such maneuvers include moving from the end of one windrow to the start of another windrow or moving to a specified target with a desired final orientation (e.g., to the charging station).

A user can select which windrows should be turned by the autonomous compost turner. The full route consists of a list of waypoints. One windrow is stored as two separate waypoints, i.e., the start and endpoints of the *Windrow Line*. Depending on which point is closer to the compost turner, the start and endpoints of the estimated route through the windrow can be switched around.

When starting the algorithm, the first stored waypoint is passed to the route planner. The route is planned with the ROS navigation stack, specifically with the *move\_base* package which incorporates a global and a local planner. For the global path planning, the *global\_planner* package, which contains the Dijkstra algorithm, was used. Local planning was integrated using the *teb\_local\_planner* plugin [20], which is based on the timed elastic band (TEB) method. The planner then computes the optimal route to the waypoint and outputs linear and angular velocity commands to the control module of the robot. This is repeated until the last waypoint is successfully reached.

### 2.6. Simulation Environment

The developed routing algorithm is tested and evaluated within Gazebo. Gazebo is a simulation environment that allows for ROS integration. Several plugins exist that enable the simulation of navigation sensors such as GNSS receivers, IMUs, or LiDARs. Within Gazebo, the compost turner is simulated and can be steered manually using a controller. Outputs from a differential drive controller are used and translated to steering commands for the robot.

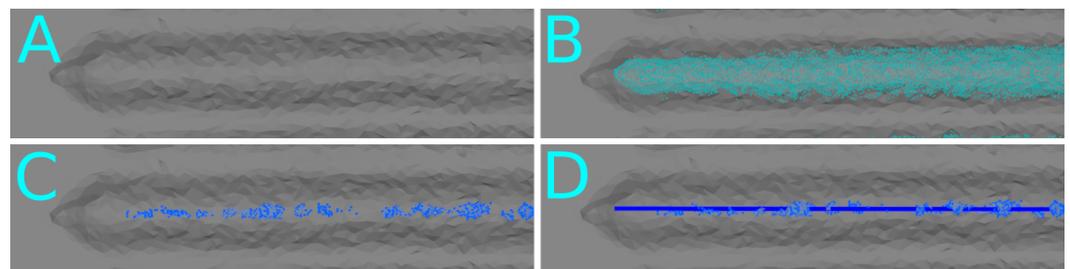
The environment of the composting plant is simulated from a mesh file. This 3D model was created using data from UAV photogrammetry. The environment contains an area of approximately 50 by 70 m. In total, there are 16 windrows, each with a width of approximately 3 m and a height of 1.5 m. The windrows are 40 m long, except for the last two, which are only 15 m long. To properly test the windrow detection algorithm that estimates the optimal routes through each windrow, several tests were conducted within the simulation environment (see Section 3).

## 3. Results

This chapter presents the results of the validation of the routing algorithm in the simulation environment. First, the processing steps for detecting a single windrow are visualized, and then the results for multiple initialization rounds are presented and compared.

### 3.1. Detection Results for a Single Windrow

Figure 5 shows the result of the windrow detection algorithm for a single windrow. The simulated windrow (A) is observed in the initialization phase as a point cloud (B). Then the ridge points of the windrow are extracted (C). Finally, a line is estimated through the ridge points (D), from which the start and endpoints of the windrow can be derived as a result.



**Figure 5.** Visualization of the processing steps for detecting a single windrow. Simulated windrow (A); Windrow after initialization (B); Extracted ridge points (C); Estimated line through ridge points (D).

### 3.2. Detection Results for Multiple Initialization Rounds

To validate the windrow detection, three sets of ten individual initialization runs were executed in the simulation environment. The three sets differ regarding the input settings (see Table 1). The *Downsampling Radius* was varied from 0.10 m (Set A) to 0.30 m (Set C), which also has an effect on the other parameters for the cluster detection (see Section 2.3).

**Table 1.** Windrow detection settings for the different test sets.

Setting	Set A	Set B	Set C
Downsampling Radius	0.10 m	0.20 m	0.30 m
Clustering Distance Tolerance	0.15 m	0.30 m	0.45 m
Min. Number of Points in Cluster	1000	500	300
Max. Number of Points in Cluster	100,000	50,000	30,000
Local Crop Box:			
Horizontal Distance from LiDAR		25 m	
Vertical Range above Ground		0.9–1.8 m	
RANSAC Distance Thresholds:			
Plane		0.5 m	
Line 1		0.7 m	
Line 2		0.5 m	
Ridge Points—Keep Percentage		40%	

Each initialization run was performed manually by steering the compost turner around the windrows of the simulated composting plant, meaning the routes are always similar but never exactly the same. Each time, all 16 windrows were successfully detected. To validate the results numerically, the estimated start and endpoints as well as the orientation of each windrow were compared. Since each windrow is detected ten times, the standard deviation of the distance from the mean start and endpoint is given in Table 2. Keep in mind that these results show the quality of the detected *Windrow Lines* compared to one another and not to an actual reference position of the windrow.

**Table 2.** Windrow detection results for the three different test sets.

ID	Set A		Set B		Set C	
	Standard Deviations Points (2D) [m]	Standard Deviations Orientation [°]	Standard Deviations Points (2D) [m]	Standard Deviations Orientation [°]	Standard Deviations Points (2D) [m]	Standard Deviations Orientation [°]
1	0.10	0.43	0.07	0.41	0.16	0.41
2	0.05	0.16	0.15	0.35	0.11	0.34
3	0.08	0.29	0.09	0.25	0.16	0.39
4	0.06	0.15	0.06	0.18	0.12	0.35
5	0.07	0.36	0.10	0.28	0.11	0.13
6	0.06	0.28	0.12	0.28	0.13	0.50
7	0.09	0.18	0.08	0.26	0.08	0.14
8	0.10	0.08	0.13	0.23	0.11	0.29
9	0.07	0.24	0.08	0.12	0.13	0.36
10	0.08	0.20	0.11	0.31	0.14	0.43
11	0.06	0.11	0.04	0.14	0.10	0.12
12	0.05	0.16	0.08	0.43	0.10	0.32
13	0.07	0.15	0.10	0.27	0.09	0.11
14	0.04	0.07	0.08	0.22	0.16	0.26
15	0.09	0.38	0.11	0.62	0.10	1.03
16	0.10	1.13	0.09	0.90	0.11	1.45
<b>Average</b>	<b>0.07</b>	<b>0.27</b>	<b>0.09</b>	<b>0.33</b>	<b>0.12</b>	<b>0.41</b>

The results show that a higher *Downsampling Radius* increases the standard deviations. On average, the standard deviations for Set A of the estimated start and endpoints was 0.07 m and the average standard deviation of the orientation was 0.27°. They increase to 0.12 m and 0.41° when looking at the results of Set C. However, the results for Set C are still sufficiently accurate to navigate the compost turner to the start of a windrow. Additionally, increasing the *Downsampling Radius* (Set C) reduces the size of the point cloud by a factor of 8. This minimizes the computational load and can reduce the hardware requirements for the mobile processing unit of the compost turner.

The last two windrows (IDs 15 and 16) are shorter than the other windrows. Since the orientation is computed from the start and endpoints, the shorter distance between these points explains the larger standard deviation.

#### 4. Discussion

This paper presented a route planning module for an autonomous compost turner. The module automatically detects the windrows on the composting plant in an initialization phase using LiDAR. This is achieved by first generating a global point cloud of the windrows from which the individual windrow clusters are extracted. Then the ridge points of each windrow cluster are retrieved and an optimal route through them is estimated. The routes through each windrow are then incorporated into a global route planning algorithm. Via user input, the windrows that the robot should turn can be selected and the optimal route to, through, and between the selected windrows can be calculated. The algorithm was tested in a simulation environment.

Tests have shown that the algorithm reliably manages to detect all windrows after the initialization phase. The start and endpoints of each route through a windrow were detected with a standard deviation of less than 0.15 m, while the orientation of each route through the windrow was estimated with a standard deviation of less than 0.5°. Therefore, this algorithm is suitable for the estimation of the windrow start and endpoints.

Future research will focus on the evaluation of the method using real-world data, including tests at a real composting plant. Currently, an obstacle detection and avoidance module as well as a precise local approach strategy to the windrows are being developed.

On the path toward smart compost plant management, autonomous compost turners play an important role. When equipped with additional sensors, they can be used to georeference and monitor temperatures and greenhouse gas emissions. This can help to ensure quality control. Furthermore, autonomous compost turners allow increasing efficiency at the composting site.

**Author Contributions:** Conceptualization, F.T., C.S. and E.R.; methodology, F.T.; software, F.T., C.S. and E.R.; validation, F.T.; formal analysis, F.T.; investigation, F.T.; resources, F.T., C.S. and E.R.; data curation, F.T.; writing—original draft preparation, F.T. and E.R.; writing—review and editing, E.R. and C.S.; visualization, F.T. and C.S.; supervision, M.W.; project administration, E.R.; funding acquisition, E.R. and M.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation, and Technology via Österreichische Forschungsförderungsgesellschaft (Austrian Research Promotion Agency) (885368).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available upon request from the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## Abbreviations

The following abbreviations are used in this manuscript:

ESKF	Error State Kalman Filter
GNSS	Global Navigation and Satellite System
IMU	Inertial Measurement Unit
LiDAR	Light Detection and Ranging
PCL	Point Cloud Library
RANSAC	Random Sample Consensus
ROS	Robot Operating System
RTK	Real-Time Kinematic
TEB	Timed Elastic Band
UAV	Unmanned Aerial Vehicle

## References

1. European Union. Directorate-General for Environment, Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions. A New Circular Economy Action Plan For a Cleaner and More Competitive Europe. Available online: <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=COM:2020:98:FIN> (accessed on 30 April 2023) .
2. European Parliament, Council of the European Union. Directive (EU) 2018/851 of the European Parliament and of the Council of 30 May 2018 Amending Directive 2008/98/EC on waste. Available online: <http://data.europa.eu/eli/dir/2018/851/oj> (accessed on 30 April 2023) .
3. European Environment Agency. *Bio-Waste in Europe: Turning Challenges into Opportunities*; Publications Office of the European Union: Luxembourg, 2020; ISSN 1977-8449. [CrossRef]
4. Razza, F.; D'Avino, L.; L'Abate, G.; Lazzeri, L. The Role of Compost in Bio-waste Management and Circular Economy. In *Designing Sustainable Technologies, Products and Policies: From Science to Innovation*; Benetto, E., Gericke, K., Guiton, M., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 133–143. [CrossRef]
5. Reitbauer, E.; Schmied, C.; Wieser, M. Autonomous Navigation Module for Tracked Compost Turners. In Proceedings of the 2020 European Navigation Conference (ENC), Dresden, Germany, 23–24 November 2020; pp. 1–10. [CrossRef]
6. Reitbauer, E.; Schmied, C.; Schedler, M. Integrated Navigation for Tracked Compost Turners Using GNSS, INS, Odometers, Stereo Camera and 3D Map. In Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2020), Virtual, 21–25 September 2020; pp. 2518–2532, ISSN 2331-5954. [CrossRef]
7. Reitbauer, E.; Schmied, C. Performance Analysis of GNSS/INS/VO/Odometry Sensor Fusion Algorithms for Tracked Agricultural Vehicles. In Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021), St.Louis, USA, 20–24 September 2021; pp. 3250–3262, ISSN 2331-5954. [CrossRef]
8. Reitbauer, E.; Schmied, C. Bridging GNSS Outages with IMU and Odometry: A Case Study for Agricultural Vehicles. *Sensors* **2021**, *21*, 4467. [CrossRef]
9. Reitbauer, E. Multi-Sensor Positioning for the Automatic Steering of Tracked Agricultural Vehicles. Ph.D. Thesis, Graz University of Technology, Graz, Austria, 2022.
10. Cichocki, M.; Reitbauer, E.; Theurl, F.; Schmied, C. Composting 4.0: From the Automatic Steering of Compost Turners Towards an Autonomous Plant Management. In Proceedings of the 7th Central European Biomass Conference (CEBC 2023), Graz, Austria, 18–20 January 2023; p. 189.
11. Hart, P.E.; Nilsson, N.J.; Raphael, B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [CrossRef]
12. Duchoň, F.; Babinec, A.; Kajan, M.; Beňo, P.; Florek, M.; Fico, T.; Jurišica, L. Path Planning with Modified a Star Algorithm for a Mobile Robot. *Procedia Eng.* **2014**, *96*, 59–69. [CrossRef]
13. Erke, S.; Bin, D.; Yiming, N.; Qi, Z.; Liang, X.; Dawei, Z. An improved A-Star based path planning algorithm for autonomous land vehicles. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881420962263. [CrossRef]
14. Tang, G.; Tang, C.; Claramunt, C.; Hu, X.; Zhou, P. Geometric A-Star Algorithm: An Improved A-Star Algorithm for AGV Path Planning in a Port Environment. *IEEE Access* **2021**, *9*, 59196–59210. [CrossRef]
15. Dijkstra, E.W. A note on two problems in connexion with graphs. *Numer. Math.* **1959**, *1*, 269–271. [CrossRef]
16. Fox, D.; Burgard, W.; Thrun, S. The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.* **1997**, *4*, 23–33. [CrossRef]
17. Roesmann, C.; Feiten, W.; Woesch, T.; Hoffmann, F.; Bertram, T. Trajectory modification considering dynamic constraints of autonomous robots. In Proceedings of the ROBOTIK 2012; 7th German Conference on Robotics, Munich, Germany, 21–22 May 2012; pp. 1–6.
18. Rusu, R.B.; Cousins, S. 3D is here: Point Cloud Library (PCL). In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; IEEE: Shanghai, China, 2011.

19. Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; Ng, A. ROS: an open-source Robot Operating System. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Robotics, Kobe, Japan, 12–17 May 2009.
20. Rösmann, C.; Hoffmann, F.; Bertram, T. Integrated online trajectory planning and optimization in distinctive topologies. *Robot. Auton. Syst.* **2017**, *88*, 142–153. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.