Received: 23 May 2023 | Revised: 13 October 2023 | Accepted: 8 January 2024

DOI: 10.1002/rob.22300

RESEARCH ARTICLE

WILEY

Exploring the feasibility of autonomous forestry operations: Results from the first experimental unmanned machine

Pedro La Hera[1] | Omar Mendoza-Trejo[1] | Ola Lindroos[1] | Håkan Lideskog[2] | Torbjörn Lindbäck[2] | Saira Latif[2] | Songyu Li[2] | Magnus Karlberg[2]

[1]Department of Forest Biomaterials and Technology, Swedish University of Agricultural Sciences, Umeå Campus, Umeå, Sweden

[2]Department of Engineering Sciences and Mathematics, Luleå University of Technology, Luleå, Sweden

Correspondence
Pedro La Hera, Department of Forest Biomaterials and Technology, Swedish University of Agricultural Sciences, Umeå Campus, Umea, Sweden.
Email: xavier.lahera@slu.se

Funding information
Energimyndigheten; Kempestiftelserna; Mistra Digital Forest

Abstract

This article presents a study on the world's first unmanned machine designed for autonomous forestry operations. In response to the challenges associated with traditional forestry operations, we developed a platform equipped with essential hardware components necessary for performing autonomous forwarding tasks. Through the use of computer vision, autonomous navigation, and manipulator control algorithms, the machine is able to pick up logs from the ground and manoeuvre through a range of forest terrains without the need for human intervention. Our initial results demonstrate the potential for safe and efficient autonomous extraction of logs in the cut-to-length harvesting process. We achieved a high level of accuracy in our computer vision system, and our autonomous navigation system proved to be highly efficient. This research represents a significant milestone in the field of autonomous outdoor robotics, with far-reaching implications for the future of forestry operations. By reducing the need for human labor, autonomous machines have the potential to increase productivity and reduce labor costs, while also minimizing the environmental impact of timber harvesting. The success of our study highlights the potential for further development and optimization of autonomous machines in the forestry industry.

KEYWORDS
autonomous navigation, autonomous robot, field robotics, motion control, systems engineering

1 | INTRODUCTION

This article presents results of using an unmanned electro-hydraulically actuated machine, which has been prototyped to carry on with fully autonomous forestry logging operations (Lideskog et al., 2015). This machine has been developed since 2014 by the Swedish Arctic Off-Road Robotics Lab (AORO), resulting in a platform for research and development in the area of forestry machine automation (AORO, 2021). The article's topic involves the very first results of performing a fully autonomous forwarding task, that is, the task of collecting and transporting logs out of the forest after tree-harvesting. To this end, the machine is able to perform autonomous navigation, object recognition, and manipulation, as well as robot motion control, combination of which are essential to successfully carry on with the forwarding operation.

1.1 | Background

Forestry has come a long way from the earliest machine made from a recycled tractor to the use of sophisticated hydraulic machinery (Nordfjell et al., 2019). Among the core examples are the harvester and forwarder machines, which are the main cutting-edge industrial tools for tree harvesting in Scandinavia. Among these machines, the harvesters cut trees to small logs within the harvest area, while forwarders collect and transport them out to the roadside for further relocation. This form of operation is known as the fully mechanized cut-to-length system (CTL), a highly productive tree harvesting system, making Sweden the third largest exporter of pulp and sawn timber (Eriksson & Lindroos, 2014; Lundbäck et al., 2021).

Operating forestry machines is a demanding and difficult job (Purfürst, 2010). Yet, standard machines come equipped with old-fashioned open-loop control systems, where every degree-of-freedom (DOF) and function is individually controlled through joysticks and buttons. Therefore, the ability to demonstrate a good working performance depends on the multitasking skills of each individual operator, resulting in large productivity variation (Pagnussat & Hauge, 2020). Apart of this, machine operators need to take thousands of multiple forest management decisions every day, while they control the machine's functions simultaneously. Over time, this cognitive load and the rough machine vibrations have a negative impact on the operators well-being, affecting also the work performance and productivity. Therefore, the interest of working as an operator has been declining in Scandinavia, making it difficult for forestry companies to recruit new qualified operators in recent years.

Research studies show that automation of forestry machines is challenging, due to the complexity of the forest environment and machine dynamics. Nevertheless, partial automation, which relieves some control of the machine from the operator's hands, has shown to be a promising technology to facilitate the work (Lindroos et al., 2017; Morales et al., 2014; Nurminen et al., 2006; Oliveira et al., 2021; Visser & Obi, 2021). As a result, machine manufacturers have recently shown an increasing interest in adopting basic automation technology, as one incremental step to boost work productivity and efficiency by making the control of machines more intuitive for operators. Examples of this adoption features the Intelligent Boom Control from John Deere, the Smart Flow and Smart Crane from Komatsu Forest, and the so-called intelligent hydraulic valves, which have given rise to the innovation of these new products (EATON, 2019; Gingras & Charette, 2017; Komatsu Forest, 2017; Lindroos et al., 2019; Manner et al., 2019; Reitz et al., 2019; Technion Onlinesince, 2017).

Due to the financial impact of forestry in the Scandinavian economy, the interest of moving toward unmanned machines for large-scale forestry operations began decades ago, at least in Sweden (Halme & Vainio, 1998; Lindroos et al., 2019). Early examples consist of radio controlled machines, such as BESTEN (eng. The Beast) The death of the forest (Beast, 2006) and the eBeaver The radio-controlled bio-energy harvester forest (Ebeaver, 2011). However, machines of this kind did not meet industry's expectations, because they were difficult to remotely operate using standard open-loop control commands. Therefore, production of such machines did not reach adoption in the market. Nevertheless, unmanned machines with different levels of automation can address some of the challenges facing forestry today, including reduced soil damage, pollution, costly manufacturing, and lack of skilled machine operators (Lindroos et al., 2017, 2019). Thus, research to move from manual control to full-automation with such machines has continued since.

1.2 | Literature review

1.2.1 | General review about unmanned forestry machines

Automation involving unmanned forestry machines performing real forestry operations is quite rare, and very few projects exists openly available in literature. To the best of our knowledge, most relevant research projects utilize standard commercial machines from well-known brands equipped externally with portable hardware to perform tests. These developments are typically supported by machine manufacturers or forestry companies, since purchasing or owning forestry machines is too expensive for academic research. A relevant example is the excavator presented in Jelavic et al. (2022), which uses a Menzi Muck M545 multipurpose machine equipped with portable hardware to show some of the first autonomous abilities of a heavy-duty machine able to autonomously navigate, recognize objects and use its manipulator to do meaningful tasks. Despite not being an unmanned machine, the work of Jelavic et al. (2022) successfully demonstrates functionalities that are required to carry on with fully autonomous forestry operations, motivating the interest of using robotics in forestry. Although, no specific test case related to an actual logging operation has been demonstrated yet, complex tasks for construction work have been reported by Johns et al. (2020).

Another related example is the research program Forestry 4.0 from the Canadian FPInnovations office (Automated harvesting with robots in the forest, 2020), where the aim is to automate forestry operations using robots. However, apart of small to medium size laboratory test benches, no example of a real forestry robot machine has been demonstrated so far. Similarly, the company Rakkatec produces a commercial version of an unmanned ground vehicle based on a machine once known as the RCM Harveri, an unmanned harvester machine developed in Finland Unmanned ground vehicles for the most demanding conditions (Rakkatec, 2021). Although the company claims that their ground vehicle can be used for autonomous operations, no demonstration of such case has ever been done.

The last example is the Ground Carriage Pully developed by the company Konrad (Visser & Obi, 2021). This was a prototype featuring a semi-autonomous/remotely operated ground unmanned forwarder to transport logs. The vehicle navigated autonomously by following the tension of a winch cable attached to it, after receiving initial commands from an operator. No complex navigation capabilities

were given to this system, more than forward and backward motions in the direction of the winch cable, but it could operate in most terrains. Nevertheless, this system is not part of the company's products, and information about further development of this system is not available.

All these examples show a clear interest of the industry for machines able to perform autonomous forestry operation. However, due to the challenges related to automation of outdoor machines using robotics, real world demonstrations have not been shown so far, at least publicly.

1.2.2 | Literature review about autonomous navigation in forestry

Autonomous navigation relies heavily on precise positioning information from sensors. However, obtaining this information can be challenging in harsh, unstructured, and partially occluded environments such as forest terrains.

In environments with limited GNSS signals, alternative positioning methods are often used for autonomous navigation. In forest environments, research often focuses on path planning and navigation control utilizing sensor fusion and Simultaneous Localization and Mapping (SLAM). One review study investigated positioning methods in forest environments using GNSS, Radio telemetry, Inertial navigation systems, SLAM, Bluetooth, RFID, Acoustic Positioning, bar and QR codes, and relative positioning methods (Keefe et al., 2019). In another research work, obstacle free paths were generated based on forest image data (Wu et al., 2009). In a study on GNSS denied environments, a UGV-SLAM solution was proposed based on SLAM and Shooting and Bellman methods for path planning and tracking (Fethi et al., 2018). In a recent study for unknown environments, an aerial-ground collaborative approach was considered (Zoto et al., 2020). For a similar research problem, navigation of unmanned vehicles using a drone to identify obstacles and execute a global path planner based on rapidly evolving random trees (RRT) using dubin curves was studied (Daniel Tenezaca et al., 2020). A recent research study further proposed a path planning algorithm for partially observed environments based on Task And Motion Planning (TAMP) by using Rapidly exploring Random Graphs (RRGs) and belief space graph methods (Phiquepal et al., 2022). However, the algorithm was evaluated for an indoor partial observable environment only.

1.2.3 | Literature review about computer vision in forestry

Computer vision has the potential to assist in forestry operations by enabling forest machines to perceive their surrounding environment. Although there is still no mature and widely used solution that utilizes computer vision in forestry, relevant research, and experimental attempts are continuously emerging.

Specific computer vision systems have been designed for particular tasks in forestry operations, such as harvesting, forwarding, and seedling planting. For instance, conventional color cameras and machine learning algorithms have been used to detect and estimate the distance of trees near forest machines during navigation (Ali et al., 2008). Similar techniques have been implemented using LiDAR and point cloud processing (Sihvo et al., 2018).

To aid in automatic grasping of objects, researchers have used image segmentation and shape reconstruction with structured light camera data to estimate the position of logs (Park et al., 2011). Additionally, a study has explored the use of a vision system comprising a conventional color camera and machine learning algorithm for the automatic detection of spruce seedlings during planting operations (Hyyti et al., 2013).

In recent years, deep learning algorithms have undergone great performance improvements, making considerable progress in applications that include tree trunk detection (da Silva et al., 2022; Wells & Chung, 2023), tree crown detection (Roslan et al., 2020), tree separation/classification (Liu et al., 2021; Roslan et al., 2020), and tree health detection (Nguyen et al., 2021; Yarak et al., 2021). Deep learning is enabling the deployment of computer vision systems on forest machines to achieve real-time complex forestry operations. For instance, to estimate the posture of stacked logs one study used conventional color cameras and image segmentation based on deep learning (Fortin et al., 2022).

While there is currently no mature industrial application of on-site computer vision systems in forestry, research is increasingly focused on developing such systems. These systems use imaging sensors to acquire data, and corresponding algorithms are built to achieve specific functions across various operational tasks. As real-time computer vision systems become more powerful, they will likely accelerate the development of forest machine automation.

1.2.4 | Literature review about crane motion control

Compared with industrial robotic manipulators, forestry cranes lack motion sensors, as they are seldom designed with having autonomous operations in mind. As a result, the necessary hardware for automation is often not readily available, making implementation of automation challenging. Therefore, forestry cranes are typically operated using joysticks that provide open-loop control commands. The operator sits in a cabin and manually controls the crane joint-by-joint, which can be unintuitive and requires strong multitasking skills for efficient operation (Morales et al., 2014; Purfürst, 2010).

Most research found on the topic of automating forestry cranes focuses on upgrading the control method from joint-by-joint to Cartesian end-effector control. This approach is more user-intuitive, as it allows the operator to control the crane's end-effector directly instead of manipulating each individual link separately. To achieve this, the crane needs to be equipped with motion sensors to implement a feedback control system for motion planning and tracking. According to literature, the most common motion planning

approach used for this case is inverse kinematics (Spong et al., 2006). The work of Hansson and Servin (2010), Mü nzer (2004), Westerberg (2014) showcases this solution with experimental hardware. Recent industrial examples applying this technology include the Intelligent Boom Control from John Deere (IBC), the Smart Crane from Komatsu Forest, and there are many other examples sold by consultancy firms around Scandinavia (Manner et al., 2019; Technion Online since, 2017).

Fully autonomous manipulation of logs is still a challenging task within the field of robotics, which is why literature on unmanned control of forestry cranes is rare. Nonetheless, two examples report experimental results of methods that could lead to fully autonomous crane operations. The first is the work of Ortiz Morales et al. (2014), where the authors present a motion control system for the crane of a Komatsu 830 forwarder machine. The main goal of their development was to showcase the ability to plan and control motions that resemble those of human operators, while also highlighting the potential of autonomous control to outperform humans in terms of speed. However, their development did not feature real world demonstrations in the forest. The second is the work of Jelavic et al. (2022), where the authors use a Menzi Muck M545 multipurpose machine. To control the manipulator, the authors use an adaptation of their former work introduced in Bellicoso et al. (2016), which was later adapted for a task of piling stones in Johns et al. (2020). However, work involving real forest operation tasks has not yet been demonstrated. Outside of academic research, there are no commercial industrial examples reported in literature, featuring any form of unmanned crane control. Nevertheless, industrial prototype machines involving this method have been undergoing development in Scandinavia (CINTOC, 2020).

1.3 | Problem formulation

Clear-cutting of productive forests with the two-machine system has been the dominant silvicultural practice in Sweden since the 1950s (Lundmark et al., 2017). There are two particular characteristics that provide the potential for automation of forwarder machines in clear-cut operations:

1. After the harvester machine finishes its clear-cutting operation in a forested area, the paths it creates during its activities are commonly known as "machine-trails," as discussed in Hosseini et al. (2019). These machine-trails serve as designated routes for forwarder machines to navigate through the forest. Typically, these trails are wide enough to accommodate the size of forwarder machines and may contain occasional obstacles like stones and stumps, which are generally manageable for the forwarder's traversal capabilities. While the digital path information for these machine-trails is not currently readily available, there are several ongoing projects dedicated to collecting, accessing and processing this essential data directly from harvester machines (Hansson et al., 2022). This collected digital path information holds the potential to enable autonomous navigation for forwarder machines.
2. Logs that have been cut by harvesters are systematically organized and stack alongside the machine-trails or network of machine-trails. This arrangement helps to reduce the time needed for the forwarder to gather the logs as it traverses through the harvesting area, specially considering its bulkier build compared with the harvester. It is worth noting that the locations and distributions of the logs, which have been cut by the harvesters, are integral pieces of information that can potentially facilitate the automation of forwarding tasks. While it is noteworthy that the introduction of motion sensors on forestry cranes are a relatively recent development, the collection and provision of this log-related data will likely become a viable option in the future.

Thus, in a clear-cut scenario, the forwarder machine benefits from navigating through nearly obstacle-free paths and having simplified visual recognition of objects, because logs are piled up on the ground making them easy to recognize unless they are covered with snow during winter. These characteristics have led to the hypothesis that forwarder machines could transition to full automation faster than harvester machines in forestry. To break down the overall forwarding process, the combined tasks of the operator and machine can be summarized in the following steps:

1. Navigation through defined machine-trail paths, where the information about the paths are given in digital form from harvester data or through detailed plan maps.
2. Visual log recognition along the machine-trails, done by the machine operator.
3. Decision on how to drive and stop the machine at an angle and proximity appropriate to grab logs with the crane. In most cases, an operator will place the machine in a position that allows grabbing multiple piles of logs without moving the machine.
4. Crane motion from the bunk toward the logs on the ground, coordinated through joysticks by the operator. Subsequently, the grapple is used to grab logs once the crane has reached a nearby location. Given the fact that standard grapples have a big holding area, multiple logs can be collected in one go.
5. Crane motion to carry the logs from the ground toward the bunk. Subsequently, machine operators sort the logs according to assortments, for example, tree species and diameters, within the bunk. The sorting might not be needed, if only one assortment is collected during the given round-trip.
6. Once the bunk is fully loaded, the machine returns through the machine-trail to the unloading area, where logs are piled up for further transport.

These steps usually happen sequentially, and they are repeated until there are no more logs to collect.

Based on these observations, this article is dedicated to present our first results having the following functionalities to approach fully autonomous forwarding tasks:

1. Autonomous navigation capabilities through waypoints, where the locations of these points are manually given as GPS data.
2. Log recognition using a 3D stereo camera, where the processing happens simultaneously as the machine is driving.
3. A task manager that provides the information of how to position the machine to collect logs after the visual system has identified them.
4. The crane motion control system able to autonomously collect logs from the ground to the bunk.

2 | EXPERIMENTAL PLATFORM

The AORO platform has been under development since 2014 as part of different projects of Luleå University (LTU) (see e.g., Lideskog et al., 2015; Rånman, 2015). These projects included course projects in the mechanical engineering program, as well as masters' and doctoral theses. The mechanical design was developed with the intention of utilizing a wide range of readily available off-the-shelf components, aiming to reduce manufacturing and construction time. Components not found in the market were manufactured in-house at a local workshop. It is important to note that no external companies had active participation in the engineering development of the AORO platform, making LTU the sole entity responsible for the machine's construction. An overview of the relevant hardware used in the machine is presented in Figure 1.

2.1 | Computing hardware

Two different computing systems are used on the AORO platform:

1. A Jetson AGX Xavier (henceforth called Jetson), with Linux as operating system, has high GPU capabilities with respect to its size and low power usage and it is used for the more computationally intensive tasks of perception and localization. The Jetson uses libraries, tools, and framework from the Robot Operating System (ROS) (ROS—Robot Operating System, 2021) as well as some Python programs. It also has two exteroceptive sensors directly connected via USB, which we will present later.
2. A UEISIM from United Electronic Industries (UEI) (2021) is used to run most of the machine's internal control systems and to read sensors. It has a light-weight Linux OS and has several I/O cards installed for sensor inputs and control outputs. This is programmed using the built in toolboxes from MATLAB/Simulink The Mathworks (1990). Using Simulink Coder, models are created in Simulink and run in real-time on the UEISIM target.

These two computers are interconnected via ethernet and communicate mainly via UDP messages. This choice is driven by the nature of our data exchange needs. UDP is a connectionless, lightweight protocol that excels in scenarios where rapid data transmission and minimal overhead are paramount. In our case, the data packets being exchanged between the Jetson and the UEISIM are relatively small, corresponding to very few values, and do not require the reliability and error-checking mechanisms inherent in TCP/IP. TCP/IP, while robust and reliable, involves additional layers of communication overhead, including handshaking and acknowledgment mechanisms, which can introduce latency and unnecessary complexity to our real-time control system. By opting for UDP, we can maintain the low-latency, high-speed communication necessary for our control algorithms and vision software while efficiently utilizing network resources between computers.

2.2 | Exteroceptive sensors

Exteroceptive sensors help measure the machine's relation to its environment and external objects. There exists two of these sensors in our machine:

1. The machine's position and orientation relative to the world is described by a dual-antenna GNSS, called Leica GPS80, connected to a national network of fixed reference stations. It is

FIGURE 1 The AORO platform's hardware components. Experiment monitoring is handled by a regular laptop and a manual, wireless emergency stop function is run entirely separate. [Color figure can be viewed at wileyonlinelibrary.com]

called Swepos Network-RTK and it is the Swedish mapping, cadastral and land registration authority's (Lantmäteriet's) satellite positioning support system. Having good satellite communication enables positioning of the antennas in sub-decimeter accuracy in Cartesian world coordinates. The dual antenna setup enables the Leica system to produce a high accuracy reading of the machine's orientation relative to absolute North since both the antenna positions are known and affixed on the machine. When the machine travels in dense forest terrain, other than clear-cuts, the satellite communication rapidly deteriorates to submetres, or worse.

2. A stereo camera, called Stereolabs Zed2, is installed at the front of the machine, pointing downwards to map the ground in front of the machine. The Zed2 stereo camera produces a plethora of sensor data, such as IMU, barometer and magnetometer data. However, so far the data we use comprise a 2D RGB image and a 2D depth map, as well as intrinsic camera data parameters.

2.3 | Vehicle and actively articulated suspension system via swing arms

The core structural elements of the AORO platform, encompassing the machine's front, rear, and pendulum arms, were designed from the ground up by LTU and subsequently fabricated at a local workshop. Other components were purchased from external suppliers.

The platforms are equipped with four swing arms where one end is connected to the frame and the other end to a hydraulic motor and a wheel. The swing arms thereby enable individual height control of each wheel in relation to the frame. The powertrain consists of a diesel engine as power supply with two hydraulic pumps mounted in series. One of the hydraulic pumps (variable displacement) is used to provide flow and pressure to the four hydraulic motors mounted at the end of each swing arm. This drive system is also equipped with an "anti spin" system actuated via flow valves. The other pump (variable displacement and load sensing) provides flow and pressure to the auxiliary equipment (crane in this case), swing arm cylinders, parking brake, and articulated steering cylinders, and so forth.

2.4 | Hydraulic crane

The AORO platform uses a model FC8 crane from the company CRANAB CRANAB FC8 (2021). This is a four degrees-of-freedom hydraulically actuated manipulator that follows a RRRP[1] configuration, according to robotics nomenclature (Spong et al., 2006). The end-effector for grabbing logs is attached at the boom-tip, model CR250 grapple, having two active degrees-of-freedom for orientation and grabbing. However, it is important to note that the grapple system is underactuated, meaning it lacks actuation at the attachment joint indicated as the boom-tip in Figure 2. This underactuation results in the grapple system exhibiting behavior akin to that of a free pendulum.

This FC8 crane belongs to a new line of products from CRANAB developed to support the introduction of smart crane functions in the industry. The special feature of this crane is to have built-in analog encoders as joint position sensors able to measure q_1, q_2, q_3, and q_4. However, at present CRANAB and its partner companies do not offer similar feature for the grapple, as it is difficult to add sensors into it for measuring the rotation q_5 and the opening q_6.

Apart of motion sensors, we equipped the electro-hydraulic valve with pressure sensors to measure pressure at each cylinder's chamber. Referring to Figure 3, all sensors are connected to the 18-bit DAC from the main UEISIM unit. The UEISIM is the main processor where all algorithms for motion control are implemented. Therefore, the UEISIM unit is in charge of providing the control signals to transform desired motion commands into mechanical motion by activating the hydraulic system.

3 | SYSTEM'S FUNCTIONALITY AND METHODOLOGY

As described in Section 1.3, the AORO machine performs a sequential set of steps, result of which resembles tasks made with a human operated forwarder machine. The purpose of this section is to describe the system's functionality step-by-step, and consequently give a description of the methods and algorithms that are implemented on the machine to achieve the expected results.

3.1 | Functionality and limitations

Currently, the AORO platform follows a sequence of steps that repeat in a loop. Referring to Figure 4, these steps and the procedure to use the AORO machine are as follows.

1. *Planning the mission*. Referring to Figure 4a, at this stage, the human supervisor provides the route for the mission by defining GPS coordinates, which work as waypoints for autonomous navigation. These are set through a user interface that communicates wirelessly with the main Jetson computer. The logs to be collected lay on the ground around this route in the crane's reach.
2. *Lifting the vehicle height for navigation*. The pendulum arms for the wheels are initially on their lowest position. To initialize navigation, they are lifted up to a drivability height, to easily traverse the forest over medium size rocks and stumps.
3. *Autonomous navigation*. Referring to Figure 4b, the machine starts traversing the forest at a constant speed of 2.3 km/h, using GPS information, according to the plan specified initially. As the terrain

[1]R = revolute, P = prismatic.

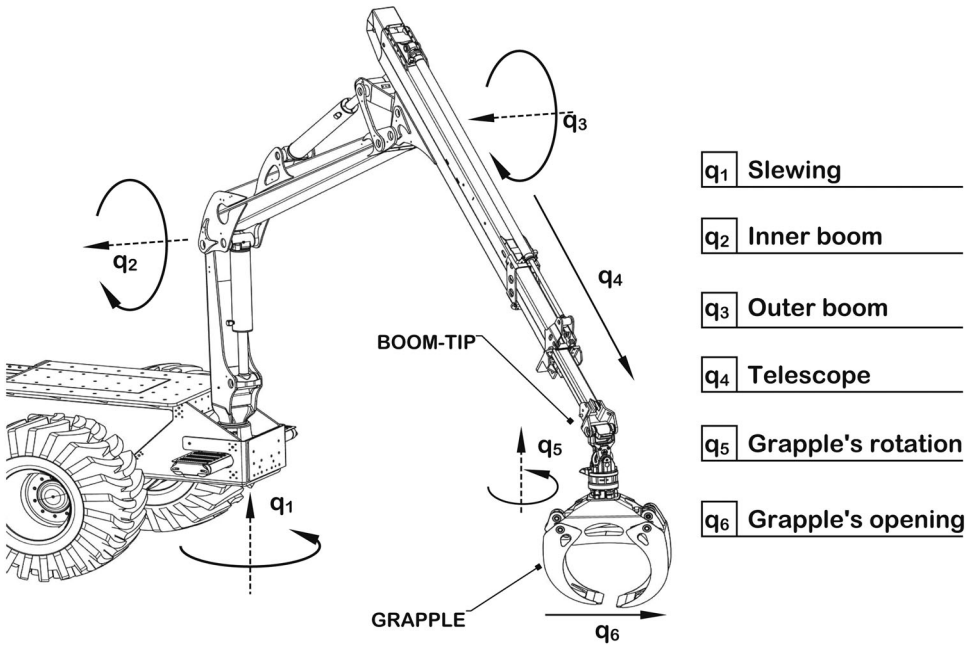

FIGURE 2 Forwarder crane: hydraulic manipulator with four degrees-of-freedom, specified in this graph as the slewing q_1, inner boom q_2, outer boom q_3, and telescope q_4. It holds an end-effector attached at the boom-tip, serving as a tool to grab logs. It is known as the grapple, having two active degrees-of-freedom, specified as q_5 for rotation, and q_6 for its opening. All sensors measure positive in counter-clockwise direction.

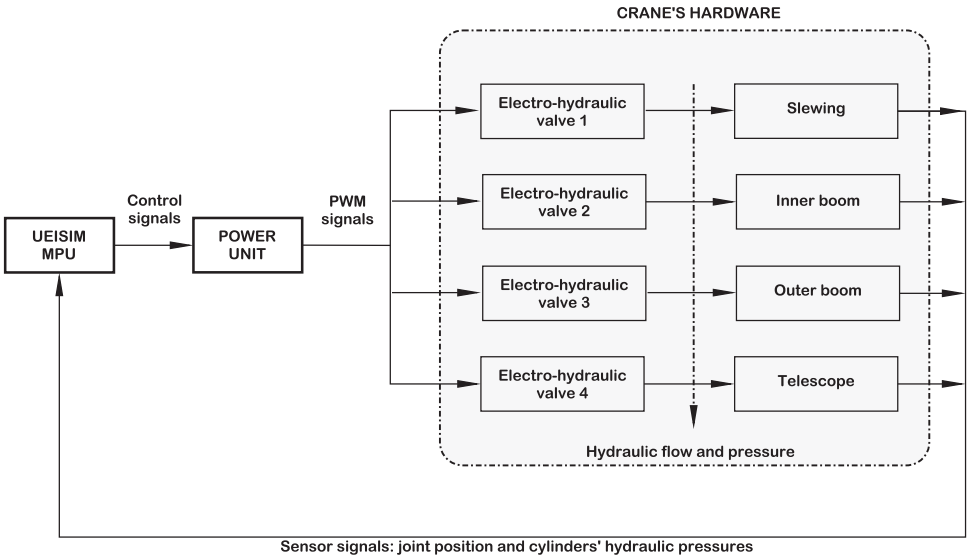

FIGURE 3 Hardware architecture used as part of the crane's control system.

refers to clear-cut, nearly no obstacles exists on these paths, apart of stumps and rocks that the vehicle can drive over.

4. *Scanning for logs*. Referring to Figure 4c, the system constantly scans for logs on the ground during vehicle navigation, using the stereo camera placed in front of the machine.
5. *Stopping the vehicle when logs have been found*. Once a log or groups of logs have been found, the machine shortly stops at a distance of 4 m to get a better visual recognition and positioning of the logs. This information is then translated to the machine's coordinate system, which is used to provide coordinates to the crane's motion control unit.
6. *Placing the load bunk for collection*. At this stage, the vehicle travels an average distance of 5 m passed the logs, and stops at a distance where the logs lay almost next to the load bunk. This facilitates the crane's tasks, by minimizing the crane's distance of travel. When several logs are found, the system stops at a distance where the crane will be able to reach all logs in a nearby area without the necessity to move the vehicle.

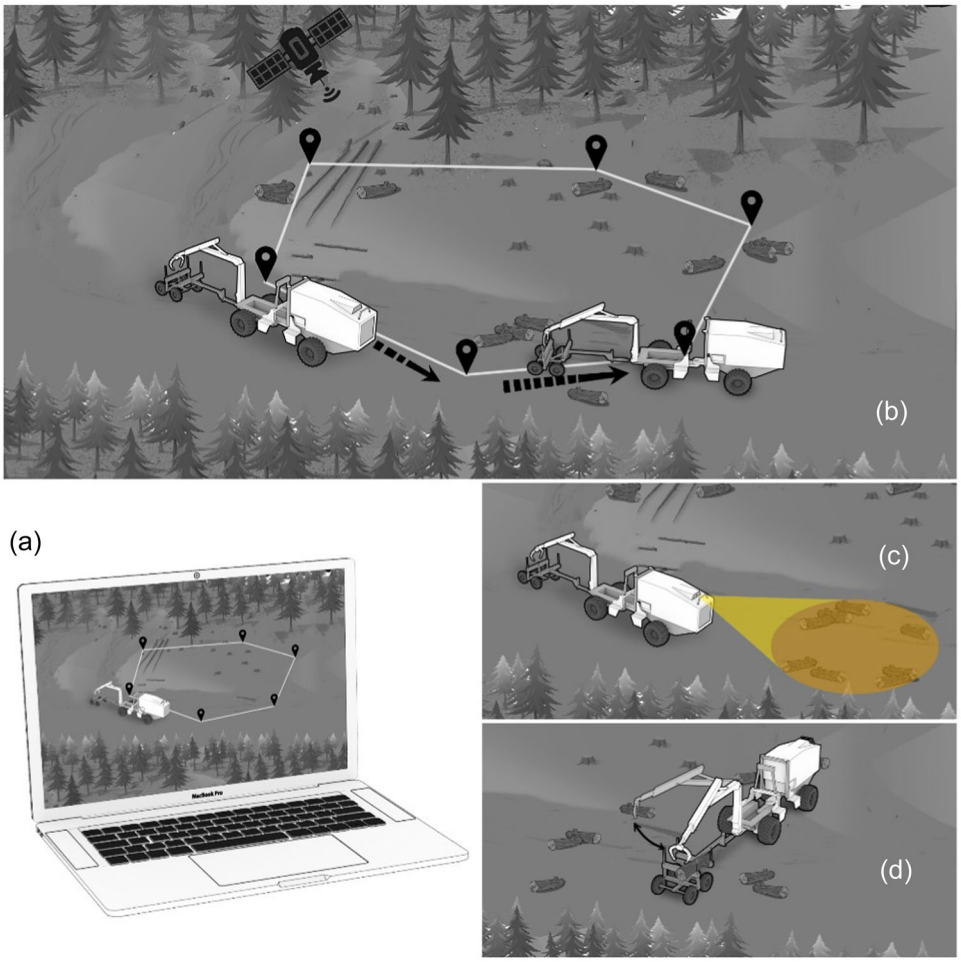

FIGURE 4 Sequence of steps that the AORO machine follows. (a) Planning the mission. (b) Autonomous navigation. (c) Scanning for logs. (d) Collecting logs. [Color figure can be viewed at wileyonlinelibrary.com]

7. *Lowering the vehicle*. At this stage, the vehicle height is lowered to its minimum range to have a lower center of gravity and better stability when the crane is moving and carrying logs.
8. *Collecting logs*. Referring to Figure 4d, at this stage, the crane's motion control system plans the necessary trajectories to reach the logs and load them into the load bunk. From this point, the sequence starts all over again from step 2 and ends when the last waypoint is reached or by manual means.

As this article refers to the very first results with this machine, and knowing that this project is undergoing development, there are certain limitations in the system's current functionality that are important to highlight:

1. The plan for the operation needs to be inserted by a human supervisor using coordinates in GPS format, indicating the waypoints that will be used for autonomous navigation. The possibility to extract and process the path information data from harvester machines is work in progress belonging to broader "forest digitalization" initiatives in Sweden, and it will take time before it reaches fruition Holmström (2020).
2. Currently, neither the grapple's rotation q_5 nor its opening q_6 are equipped with measurement capabilities. Therefore, in this initial tests, there is no motion control system in place to control the rotation q_5, and the opening q_6 is controlled via an open-loop command that merely opens and closes it. As a result, to effectively use the crane for grasping tasks, the logs need to be positioned on the ground at an angle relative to the main navigation path. Based on observations, following the sequence outlined earlier, attempting to grasp logs that are either parallel or perpendicular to the machine's orientation is more likely to result in failure. However, anything in between, owing to the grapple's wide opening, remains a viable possibility. It is worth mentioning that the control of the rotator has formerly been developed within our research (La Hera et al., 2009), and it is a matter of time until we have the hardware needed to include this part of the development in the AORO machine.
3. As neither the grapple's rotation nor opening are measured, there are not any log orientation capabilities in place for the experiments. Nevertheless, after the experimental tests presented in this paper took place, a solution has been published by the research group (Li & Lideskog, 2023) and by others (Fortin et al., 2022) that can be implemented to assist log grasping if the grapple's angle is known.

4. When the machine stops to collect logs, there is currently no camera on the crane to observe the logs on the ground. Therefore, the accuracy of the crane's work currently depends on the correctness of the log position estimation using the camera mounted at the front of the machine. As this affects the possibility to collect multiple logs in one go, performing such action is out of the scope of this article, and it still work in progress.
5. The sequence of steps is repeated until the last waypoint is reached or by a human operator that stops the system. Currently, the system does not have a method to estimate when the load bunk is full.
6. Currently, the system is only capable of loading the load bunk, but not the reverse task, because control of the grapple's rotation q_5 and a dedicate camera in the crane to provide visual information are needed to perform this action.

3.2 | Methods and algorithms

The algorithms required to perform the tasks described above have been divided into actions that are supervised and commanded through a task manager. The activity manager provides the required flags to switch between each action, such that two actions cannot happen simultaneously.

The main functions implemented on the AORO platform are the following:

1. *Autonomous navigation*, in which the machine is able to navigate by specifying waypoints in GPS format.
2. *Autonomous log recognition*, in which the machine is able to use its vision capabilities to observe logs laying on the ground, and defining their location according to the machine's reference frame.
3. *Autonomous crane motion control*, in which the machine is able to use its crane to collect the logs that have been recognized by the vision system.

The overview of the algorithms and the description of the task manager that sequentially switches among them is given below.

3.2.1 | Navigation control

The machine navigates between an initial and a final destination by following a trajectory that connects GPS waypoints. In this study, we adopted a straightforward approach based on the direction of the machine's front end. The algorithm calculates the difference between the bearing of the current waypoint and the heading of the machine's front end, and feeds this information into a P controller that determines the desired rate of steering angle. The controller then sends this information to a hydraulic valve that controls the cylinders that adjust the articulation angle. When the machine approaches a certain distance from the waypoint (based on the machine's steering radius), it switches to the next.

Robot localization

The robot localization node utilizes built-in functions and packages from ROS to create a real-time kinematics model of the machine using kinematic equations (see Figure 5). First, an XML file is composed where basic geometry, all links, and joints are described. Then, feeding sensor data as input to the localization node, rigid transformations define the position of each degree-of-freedom at each time step.

3.2.2 | Vision system

At the moment, the objective of visual perception is to enable fast-enough log detection, so that position coordinate outputs can be used in subsequent crane control. The visual perception runs in ROS and contains two nodes. The first is called DNN and comprises a deep neural network used for real-time object recognition. The second is referred to as Vision, comprising the data evaluation of detected objects where logs are extracted and positioned in absolute terms.

For the DNN node we use a modified version of the Legged Robotics ROS package called darknet_ros (YOLO ROS: Real-Time Object Detection for ROS, 2021), which comprises a real-time object detector named YOLOv3 built within a neural network framework named darknet (Redmon, 2021). Darknet-53 performs well for our purpose, at the time of testing working twice as fast with the same classification accuracy for comparable nets (Lawal, 2021). This is an important property given the necessity of real-time execution. At the time for full testing, YOLOv3 had shown substantial improvement compared with earlier versions (Redmon & Farhadi, 2018) and had ROS integration. The detector was previously tested in forestry applications (Li & Lideskog, 2021).

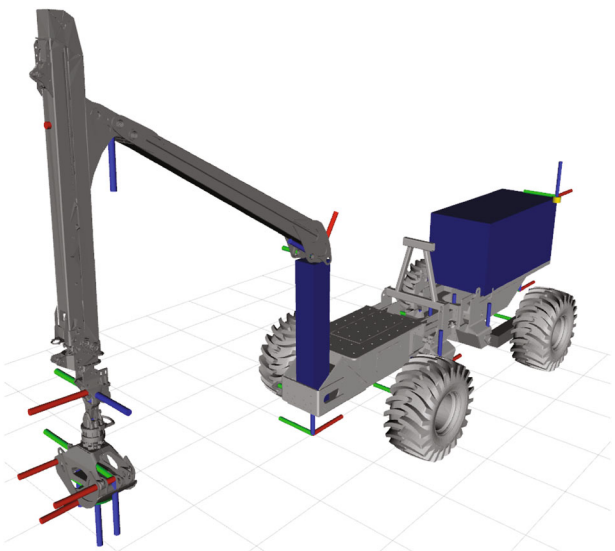

FIGURE 5 AORO robot joint and link definitions graphically depicted in Rviz (3D visualization tool for ROS applications). [Color figure can be viewed at wileyonlinelibrary.com]

FIGURE 6 Examples of labeled images from test site where data was recorded, and subsequent demonstration and experiments were performed. [Color figure can be viewed at wileyonlinelibrary.com]

DNN node: Data collection, labeling, and augmentation

To train the neural network in the DNN node, RGB data was collected at the forest area to be used for forwarding demonstrations and experimental validation.

Image data was collected using the stereo camera Zed2 with a resolution of 1280 × 720 pixels. To this end, we manually moved the camera by hand in an area of 75 × 75 m where logs had been strategically dispersed to represent different placements and orientations. The weather was sunny with a slight overcast, producing shaded parts on the grounds. No rain had fell for days and the grounds were dry. That enabled the opportunity to ensure data was collected both having the sun at the back and front facing the camera. Logs were also dispersed in both shaded and sunlit areas, as well as areas with and without vegetation. The logs consisted of birch trees with a mean diameter of 25 cm and a length of 2 m. Images were captured so that distance and angle to logs were differed to ensure images spanned the entire range of possible encounters for a machine in that area (see Figure 6). Only birch logs were used in all images, ensuring no confusion with other tree species (apart from standing trees). In total, 1700 images containing over 3500 annotated logs were captured and used as a training and validation data set.

A common step to increase the robustness of an object detector working on RGB imagery is to augment the training data set by adding copies of already existing data, but having them slightly modified in terms of, for brightness and contrast. That alleviates overfitting when training the detector and acts as a regularizer (Shorten & Khoshgoftaar, 2019). The original data set was augmented with modifications in image rotation, saturation, exposure, blur and noise, increasing the total data set size to 6300 images of which a major part of 95% was used for training and the rest for validation. The validation data set was selected randomly from the original raw data. After finalized training, testing procedures was done on film material from the same environment to ensure proper function.

DNN node: Preparing the data set for training

The data set was resized to 608 × 608 pixels, where black edges over and under the image ensured the correct ratio. This image size is specifically tailored to suit the training process for our choice of neural network.

The ROS package darknet_ros was also used for training, with a neural network having 53 convolution layers, named Darknet-53. Within

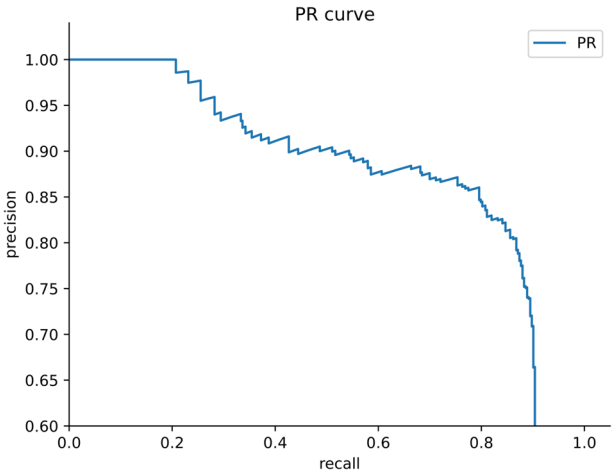

FIGURE 7 Precision/recall curve extracted from resulting training. [Color figure can be viewed at wileyonlinelibrary.com]

Darknet-53 the encapsulated YOLOv3 has a training process that reflects the original YOLOv3 algorithm. By feeding the training set into Yolo-ROS, we can get a so-called weights file trained on the prepared data. A weights file is a binary file containing the parameters for the 53-layered neural network. With transfer learning, we utilized a pre-trained model that was previously trained on logs as objects, building on a source data set of approximately 1000 images. This data set comprised annotated single birch logs scattered on a grassy field, and the pre-trained model was trained similarly using the darknet_ros package. Our final weights file achieved a mean average precision (mAP) of 80.51%, which is an evaluation metric (Salton & McGill, 1983) commonly used in the PASCAL Visual Object Class (VOC) challenge (Everingham et al., 2010). The input image size to the network were 608 × 608 pixels, momentum was set to 0.9 and initial learning rate to 0.001. Decay was set to 0.0005 and the training steps to 3200 and 3600. The resulting Precision/Recall curve is depicted in Figure 7.

3.2.3 | Vision node

The vision package encompasses the log positioning algorithm, which uses input from the stereo camera depth map and input from the

DNN as bounding box positions to calculate and publish log positions relative to the camera coordinate system.

Positioning detected objects in relation to the camera (and thus the machine itself) is conducted as follows. After target objects are recognized by the DNN node, the corresponding depth map of the area and corresponding bounding box 2D image coordinates are fed into the vision node. The depth information within the bounding box is used to calculate the actual position of the object. We use the mean value of depth data in an area of 8×8 pixels within the bounding box center to calculate the object depth, which significantly increases calculation robustness, but may add a few milliseconds to the calculation time. Furthermore, this is a viable method since the camera is directed downwards and to a great extent depicts terrain areas and not areas that otherwise would rapidly change.

Using the model of a pinhole camera Forsyth and Ponce (2003), the relationship between the coordinate of a 3D space point $[X, Y, Z]$ and the pixel coordinate of its image projection in 2D $[u, v]$ is given by

$$Z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \cdot [R \quad t] \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{1}$$

where R and t denotes rotation and translation, which relate the world coordinate system to the camera coordinate system, f_x and f_y are the camera's focal length in X- and Y-axis, and c_x and c_y are the center of the camera's aperture. Equation (1) gives that

$$\begin{bmatrix} X \\ Y \end{bmatrix} = Z \cdot \begin{bmatrix} (u - c_x)/f_x \\ (v - c_y)/f_y \end{bmatrix}. \tag{2}$$

If the pixel coordinate (u, v) and the depth information of the pixel Z is known, the actual spatial position represented by the corresponding pixel can be calculated according to the camera parameters f_x, f_y, c_x, and c_y, which are usually provided by the camera manufacturer.

3.2.4 | Active suspension height control

To set the vehicle height, a reference height is sent from the activity manager to the swing arm control system. This reference value is then used to control each swing arm hydraulic cylinder individually. To this end, a proportional controller is used to generate the control signals for the hydraulic valve of the swing arms' cylinders.

To improve the machine's navigation, the vehicle is equipped with passive suspension through float control valves. These valves can be activated on either the front or rear axle, but never on both simultaneously to prevent the vehicle from tipping over. When activated, the swing arm cylinders on the left and right sides are cross connected, that is, the inlet port on one cylinder is connected to the outlet port on the other, and vice versa. This causes the cylinders to move simultaneously in opposite directions, thus keeping the ground pressure equal on both sides, similar to a beam axle on a car. The float configuration is activated just before the machine moves for drivability and deactivated during the log loading operation for stability.

3.2.5 | Crane's motion control system

Controlling heavy-duty hydraulic manipulators is challenging. Some of the main reasons are the unpredictable loads and nonlinear dynamics, which include large amounts of residual vibrations resulting from oil compressibility and mechanical flexibility (Manring, 2005).

Our research group has been involved in the development of automation technology for forestry cranes for nearly two decades. Therefore, the motion control algorithms applied in the AORO platform are modifications of former work, some of which are presented in La Hera and Morales (2014), La Hera et al. (2021), La Hera and Ortíz Morales (2015), and references therein. In short, the reference tracking feedback controller is a nonlinear controller using sliding mode control, giving robustness to unmodeled dynamics and disturbance rejection. In addition, this control strategy allows to attenuate residual vibrations, a common problem from the crane's dynamics behavior. The validation of the performance of this type of control system has been presented in La Hera and Ortíz Morales (2015).

To plan motions, we use Dynamic Movement Primitives (DMP), following the machine learning algorithm detailed in La Hera et al. (2021). This approach is based on the concept of learning by demonstration, in which we manually demonstrate point-to-point motions to the crane, following quasi-parabolic paths, as those used by professional machine operators. This is done using joysticks. Consequently, the crane is able to use these demonstrations to dynamically plan any point-to-point motions from an initial to a desired position, mimicking the characteristics of the demonstrated motions.

Algorithm 1 Motion planning based on DMP

Input: Location of logs given in Cartesian World Coordinates [X,Y,X] and rotation ϕ

Output: Joint reference trajectories $q_{i_{ref}}(t)$ where $i = [1, 2, 3, 4, 5, 6]$

procedure MOTIONPLANNING
From bunk towards log:
  If *trigger_signal* = 1 **then**
    $[q_{1_{ref}}, q_{2_{ref}}, q_{3_{ref}}, q_{4_{ref}}] = DMP\_Algorithm(X_{target}, Y_{target}, Z_{target})$
    $[q_{5_{ref}}, q_{6_{ref}}] = CloseGrapple(\phi_{target})$
From the sides towards bunk:
  If (*target_reached* = 1) & (*grapple_closed* = 1) **then**
    $[q_{1_{ref}}, q_{2_{ref}}, q_{3_{ref}}, q_{4_{ref}}] = DMP\_Algorithm(X_{bunk}, Y_{bunk}, Z_{bunk})$
    $[q_{5_{ref}}, q_{6_{ref}}] = OpenGrapple(\phi(0))$
  finished_sequence = 1.

The algorithm to control crane motions is sequentially executed once the activity manager has sent a trigger command. This trigger command results from successfully recognizing a log on the ground and getting its Cartesian location, as well as stopping the machine at a

preplanned distance to collect the log. The input to the algorithm is the position of the logs in Cartesian Coordinates, according to the crane's reference frame. Consequently, the algorithm plans crane trajectories from the bunk to the side of the vehicle where the logs are located. The grapple opens during this motion. Once the target position is reached, the grapple closes and holds the logs. The reverse of this sequence is executed to load logs into the bunk. A pseudocode giving an idea of how the algorithm works is presented in algorithm 1.

3.2.6 | Activity manager

To control different activities and monitor progression toward achieving the mission, an activity manager was developed using Simulink Stateflow and integrated in the main Simulink software structure (The Mathworks, 1990). To control the start of the sequence and enable emergency stop, a manual switch between "Idle mode" and "Auto mode" is used. The activity manager structure in Auto mode is visualized as a flowchart in Figure 8.

When activating the Auto mode via this manual switch, a check of the vehicle height is performed. If the height is outside a suitable transport level, the activity manager sends a set-height command to a pendulum arm control system (outside the activity manager see Figure 9) which then independently adjusts the pendulum arms until the set-height is reached. When the transport height is correct the "Log Search" activity starts. In "Log Search," the drive control system is activated, and the vehicle starts following the predefined fly-by waypoints. As previously stated, note that it is assumed that the drive path follows a strip-road from the harvester track and that all logs to be picked up can be reached from this path.

During navigation, stereo camera information of the surrounding is gathered via the perception system and used to continuously store average values of the identified log positions. In addition, based on the GNSS-position of the vehicle given by the localization system, the distance between the crane reference frame and the closest log is calculated. When this distance is below a threshold $\delta > \epsilon > 0$, the closest log is set as the target log whereas the tracking system is deactivated and thus the vehicle stops. To improve the positioning accuracy, the detected log positions are then updated at standstill. After a predefined hold-time the tracking system is activated and thus the vehicle starts to drive. For collecting the logs, the crane's reference frame should be positioned at a certain range and ahead of the target log. This is achieved by using a circle with origin at the target position and a predefined radius. The position of the crane foot in relation to the target log is then monitored to ensure that the crane foot first enters the circle. Then, as the crane foot exits this circle, the tracking system is deactivated and thus, the vehicle stops.

At this stage, a validity check is performed by evaluating that the target log position is realistic in the local crane coordinate system (used for the log collection). If not, the system goes back to the log search. Otherwise, a check for log collection height is performed. To improve the load capability and machine stability, a low set-height is sent to the pendulum arm control system, which independently adjusts the pendulum arms until the set-height is reached. Then the log collection activity is performed by sending the target log position relative the crane foot to the crane control system, which generates local crane trajectories (including the grapple). These trajectories are

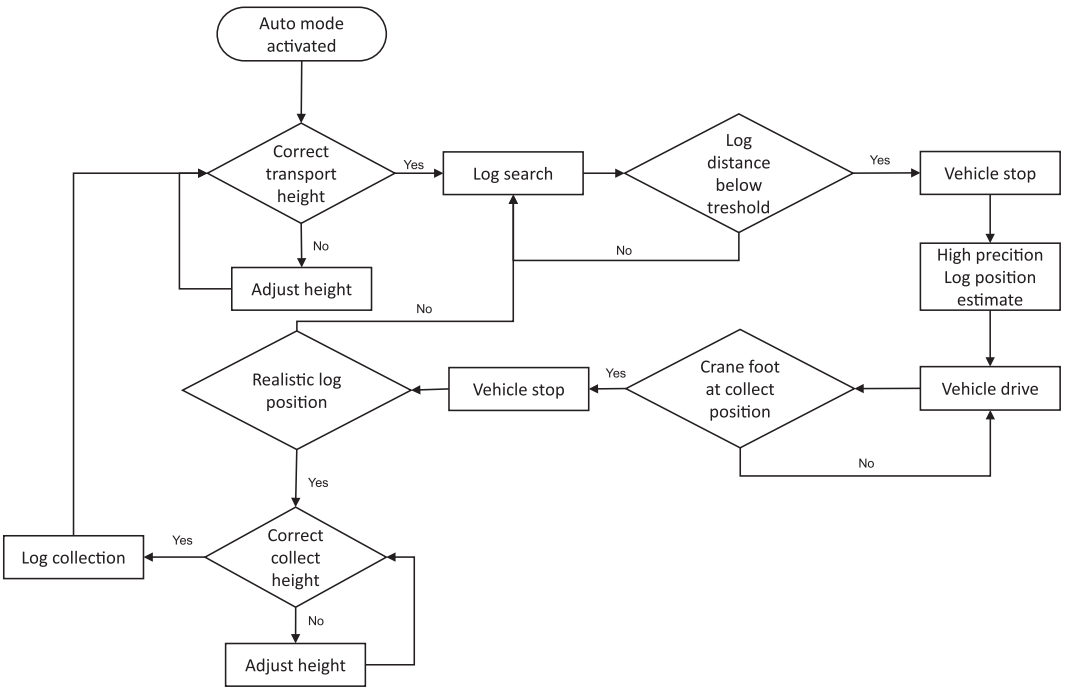

FIGURE 8 Activity manager flowchart in auto mode.

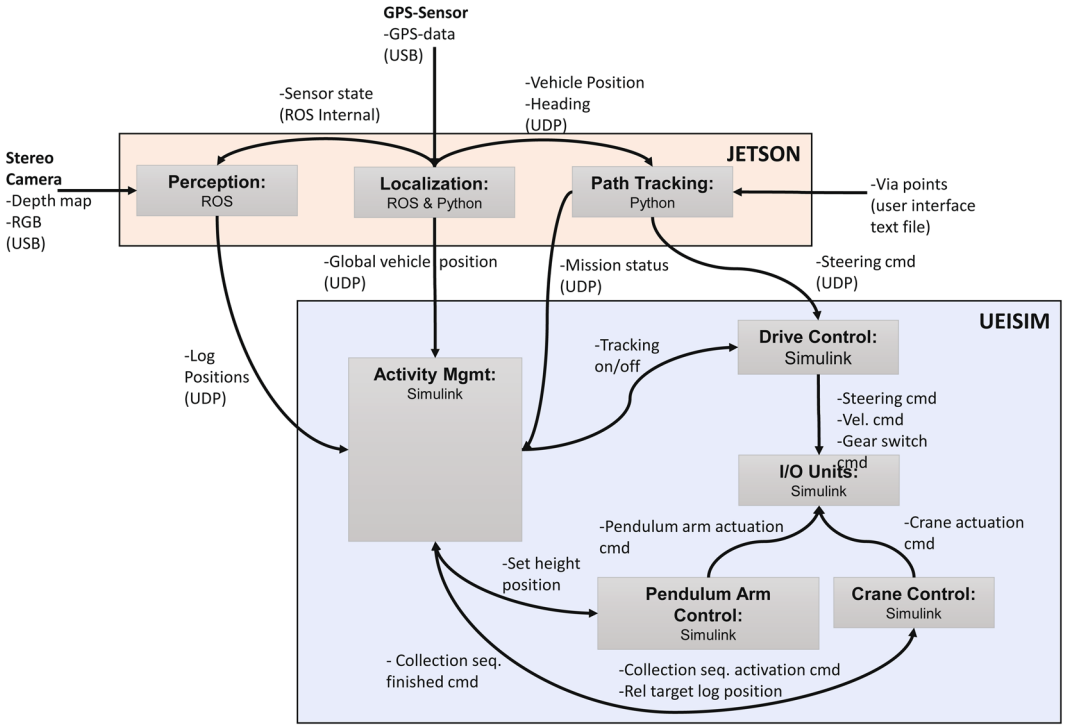

FIGURE 9 Activity manager software architecture. [Color figure can be viewed at wileyonlinelibrary.com]

used to control the actual crane movements. When the collection activity is finished, a command is sent to the activity manager which then returns to check the transport height. This state-transition behavior then loops until the mission has been completed, i.e. when the final waypoint has been reached, whereas the system is set into Idle mode.

4 | TESTING SCENARIO

The tests we present below have the intention to present our development in two stages. In the first stage, the intention is to evaluate the performance of each individual component separately, that is, autonomous navigation, log recognition, and crane motion control. In the second stage, the intention is to evaluate whether the task manager is capable to combine these methods to perform a fully autonomous forwarding task.

4.1 | Tests for the individual system's functionality components

4.1.1 | Testing the autonomous navigation control system's performance

Tests were performed to verify the machine's ability to autonomously navigate by defining waypoints in GPS format. These tests consisted on defining a nearly rectangular path defined by four waypoints (see Figure 12), and letting the machine traverse this route for four consecutive laps. To test the robustness and converge to the same path from different initial conditions, the machine's initial position was changed for each new test sequence.

4.1.2 | Testing the crane's motion control system's accuracy

Tests were performed to verify the performance of the crane's motion control system and to calculate an estimation of its accuracy. These tests consisted on performing automatic motions from an initial configuration toward different desired goal locations in the world coordinate system (Cartesian coordinates), to calculate the deviation at the final position. The initial configuration was the center of the trail where logs are piled up (see Figure 10). The goal location referred to the desired Cartesian coordinate target where the crane's tip was meant to reach. Referring to Figure 10, four desired test locations were selected to cover the four quadrants in the x–y axis, having the coordinate system at the base of the crane.

To calculate an estimation of the positioning accuracy, the deviation of the crane's tip position to the desired target was calculated by Euclidean distance. To calculate the tip position, we used the forward kinematics presented in La Hera et al. (2021). To get a reliable estimation, several repetitions of the same motion to each target location were performed. Consequently, the mean value for all the deviation errors of the four target locations was used to get a final estimation of the crane's positioning accuracy.

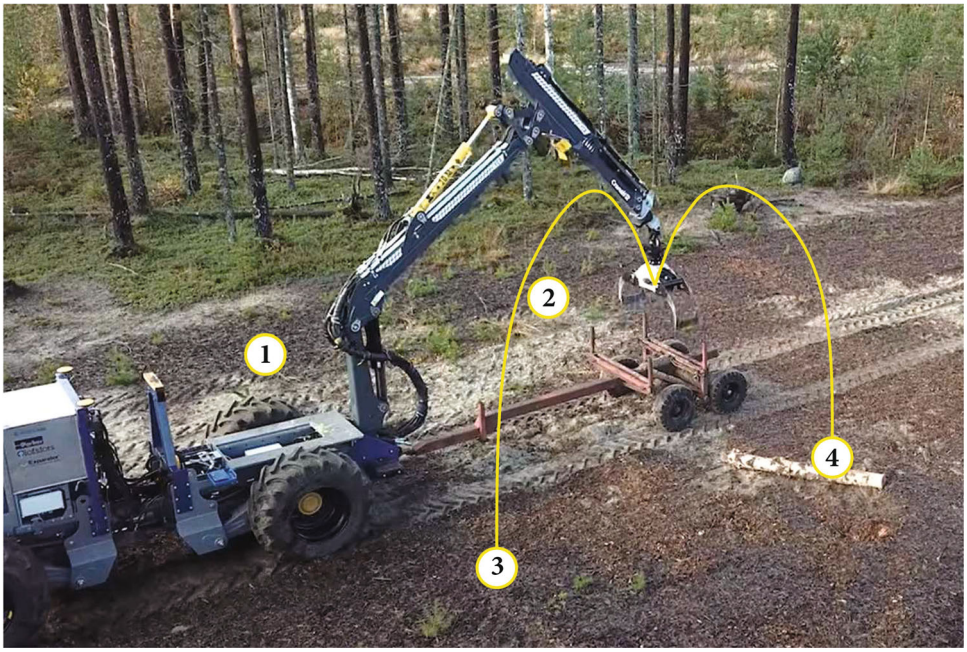

FIGURE 10 Crane's desired motions from it initial configuration at the center of the trail, toward four desired locations in the Cartesian Space. [Color figure can be viewed at wileyonlinelibrary.com]

4.1.3 | Testing the logs' location algorithm accuracy

Tests were performed to calculate the accuracy of the log recognition algorithm. These tests consisted in estimating the location of logs that were manually placed on the ground at specified known locations. To this end, the test consisted on placing two logs, 5 m apart from each other, according to the reference frame presented in Figure 11. The logs were placed according to the following characteristics:

1. In the first instance, the logs where placed at 0°, specified by the reference system observed in Figure 11. In this position the machine has the highest visibility to the whole log, as the direction of travel is perpendicular to the logs.
2. In the second instance, the logs were placed at 45° of rotation. In this position the machine observes the logs at an angle.
3. In the third case, the logs were placed at 90° of rotation. In this position the machine has the least observation of the logs, as the direction of travel is parallel to the logs.

For each of these cases, the machine started autonomously driving from a distance of 40 meters away from the first log (see Figure 11 for reference).

The log recognition software provides 15 estimations per second according to the machine's main reference system. Consequently, this data is transformed into different reference systems to, for example, provide logs' location coordinates to the crane control system, or to navigate the vehicle to better position itself when collecting logs. In these particular tests, the data was transformed into the world coordinate observed in Figure 11.

The purpose of the estimation is to give the coordinates of the center of the logs, as this is the value that is transferred to the crane's motion control system. To get a reliable estimation, several repetitions were performed to obtain an averaged value of the recognition software accuracy. Consequently, we measure the deviation of the estimation to the values measured by hand using Euclidean distance. The deviation from true values gives us an estimation of the recognition software accuracy.

Precision measurement

To assess the entire system's precision with an external measuring tool, we used a mobile two-camera setup. For each position a log was recognized and targeted by the system, two images were taken. We chose to program a short pause in the crane's movement toward the log, stopping for a second on top of the log just before activating the grapple and grabbing the log. That enabled the cameras to record an image frame each in the machine rear end's longitudinal and transversal direction. With knowing the grapple true size, each image was given a measurement scale and thus a measure from the grapple's ideal grabbing point and the log's center point was determined. No rounding of measures has been done in the resulting values; however, at least a ±2 cm error is to be expected.

4.2 | Testing the complete forwarding task

The experimental scenario refers to the location where the AORO machine was unveiled to the public, which is also the place where we performed experimental tests The world's first self-propelled forestry machine (2021). It is a region in the northern county of

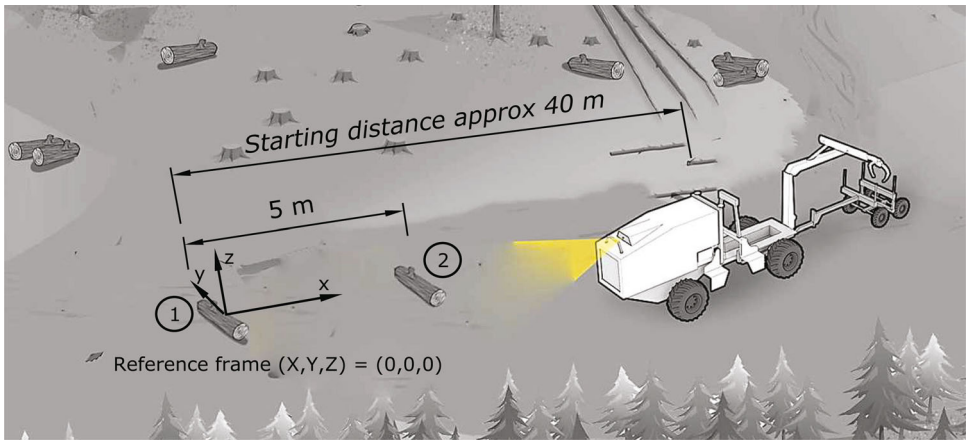

FIGURE 11 The setup to test the log recognition accuracy. Two logs are placed on the ground, 5 m apart. The machine starts autonomously driving from 40 m, giving estimations for both logs as it goes. [Color figure can be viewed at wileyonlinelibrary.com]

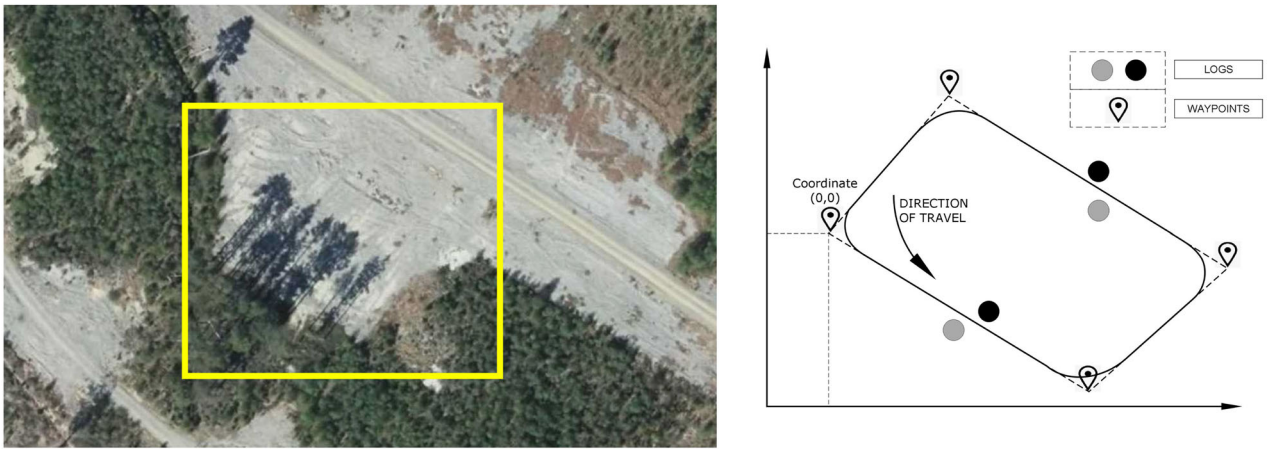

FIGURE 12 Left: Forest site where the experiments were carried out. The square points to the particular area for the experiment. Right: The mission planned for testing the machine's ability to carry on with fully autonomous forwarding. Note that this is just an illustration and not the exact vehicle path. [Color figure can be viewed at wileyonlinelibrary.com]

Sweden belonging to one of our industry partner, with the closes city being Hörnefors. The specific coordinates of the site in the WGS84 decimal (lat, lon) system is 63.657853, 19.895562. Referring to the left side of Figure 12, the testing area had a length of 40 m and width of 30 m.

The experimental testing was designed according to the following mission:

1. Referring to the area marked in yellow in the left side of Figure 12, four GPS waypoints sufficed to define the mission path (see right side of Figure 12). These points were inserted manually and communicated to the main machine's computer wirelessly.
2. As shown in the right side of Figure 12, logs were placed manually on the longer sides of the routes, either to the right or to the left of the main path. As explained in Section 3.2.2, only birch logs were used in these tests, because this is a common species for commercial harvesting in Sweden and possibly easy to recognize.
3. Each time the machine passed through the longest sides and collected the logs, they were immediately replaced by other logs for the next time the machine would pass through.
4. Referring to Figure 10, a trailer was attached to emulate a forwarder machine. The dimensions of the trailer were added to the recognition software, to have the coordinates required for the crane to release the logs it picked up from the ground.

5 | EXPERIMENTAL RESULTS

5.1 | Results for the individual system's functionality components

5.1.1 | Navigation control

Referring to the explanation of the mission plan presented in Figure 12, the initial experiment verifies the machine's ability

to navigate by defining waypoints. To this end, the machine was started from different initial conditions, to validate the exponential convergence of the motion to the same path.

Figure 13 shows three of the results from these tests, having the square and circular markers pointing at the initial and final locations of the machine respectively. In all these cases, we let the machine run for two laps to verify that the motion converges toward the same path connecting the waypoints. Figure 14 shows the heading error for the same tests. While the machine is aiming toward one of the waypoints, the heading error should be zero. In addition, each peak represents a turn, where the setpoint for the heading is abruptly changed by approximately 90 degrees, to head toward the next waypoint. The settling of the heading is about 9 degrees/meter. In Figure 15, we show the results of six test cases superimposed on the same image, which clearly shows that the machine's autonomous navigation control system is able to exponentially converge toward the same path, despite starting at different initial conditions.

5.1.2 | Logs' recognition

Referring to the tests described in Section 4.1.3, Figure 16 shows results from the log recognition algorithm. From left to right, the three plots in Figure 16 refer to the cases where the logs are placed (a) straight to the camera, having the highest visibility of the log, (b) at 45° angle, where the camera observes the log at an angle, and (c) at 90° angle, where the camera would have the least visibility of the log.

As the system is able to provide 30 estimations per second, our software uses the averaged value to give a final estimated result. Figure 17 shows results using a histogram for the log marked as two in Figure 11. From left to right, the histograms in Figure 17 represent the cases (a), (b), and (c) described above. The bars on this histogram represent the Euclidean distance for every observation in respect to the origin. In addition, the dark gray bar shows the Euclidean distance where the log was actually placed.

According to manual measurements and calculations from GNSS, the log was placed at an approximate distance of 5.2 m from the origin (see Figure 16). It is noticeable that the observations for cases

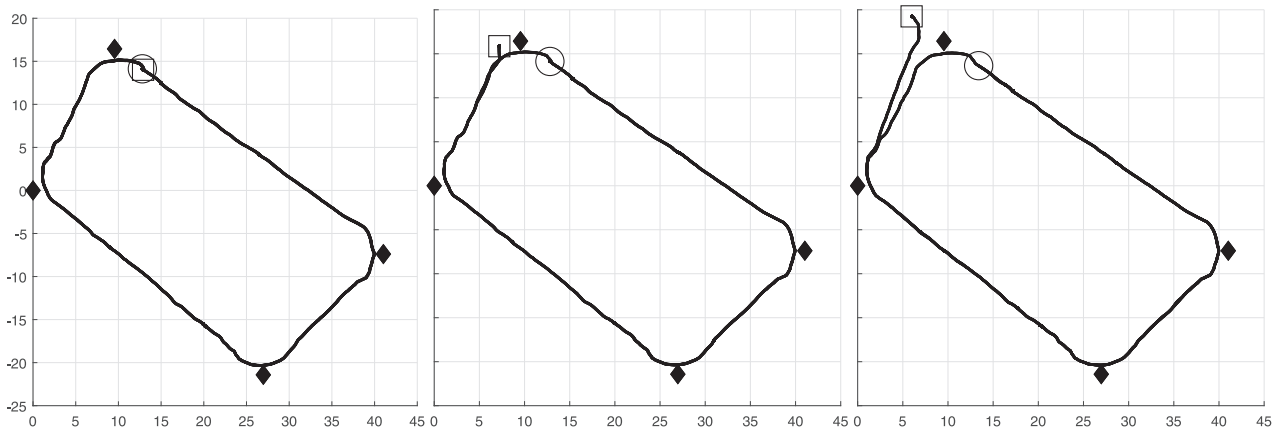

FIGURE 13 Examples of running the navigation control system from three different initial conditions. The square markers point at the initial conditions, while the circular marker point at to the final position. The four via points are plotted with back diamond shaped markers. The units in these graphs are in meters.

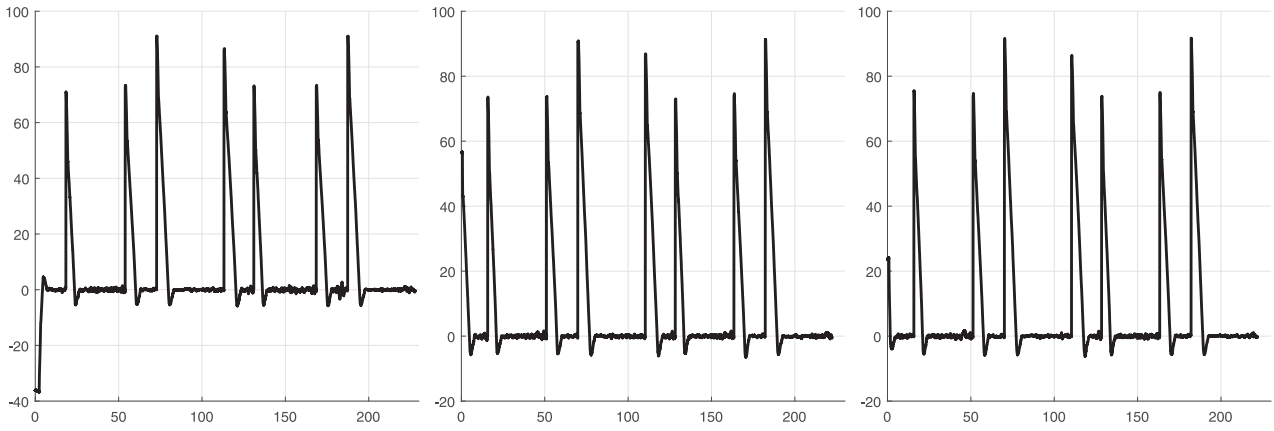

FIGURE 14 Heading error for the trajectories shown in Figure 13 (error in degrees vs. distance traveled in meters).

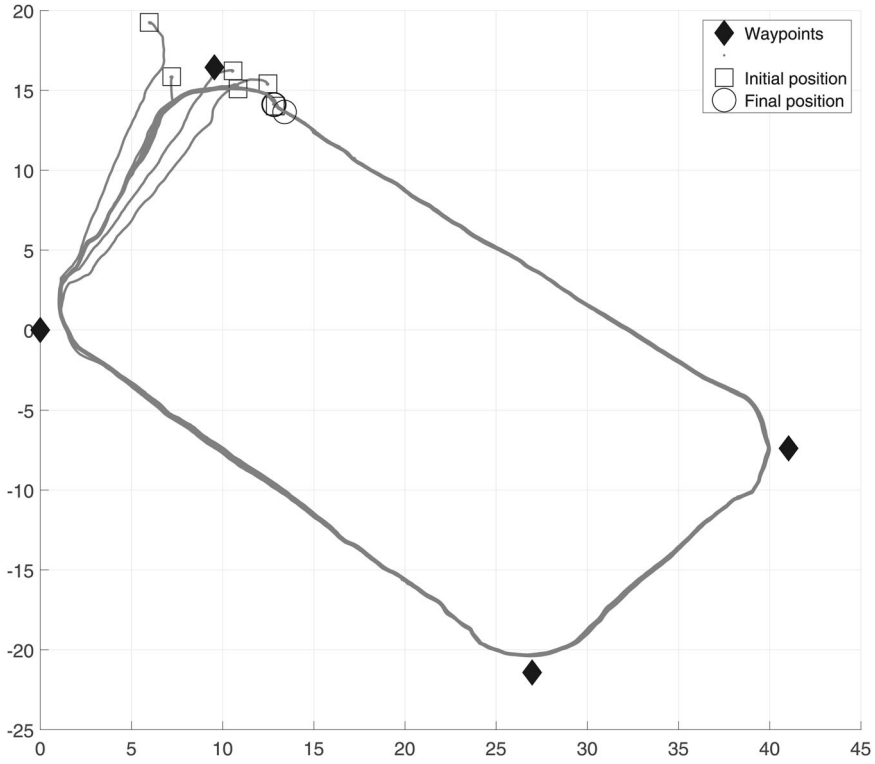

FIGURE 15 Results of the navigation control system for six different test cases. The square markers point at the initial conditions, while the circular marker point at to the final position. The four via points are plotted with back diamond shaped markers. The units in these graphs are in meters.

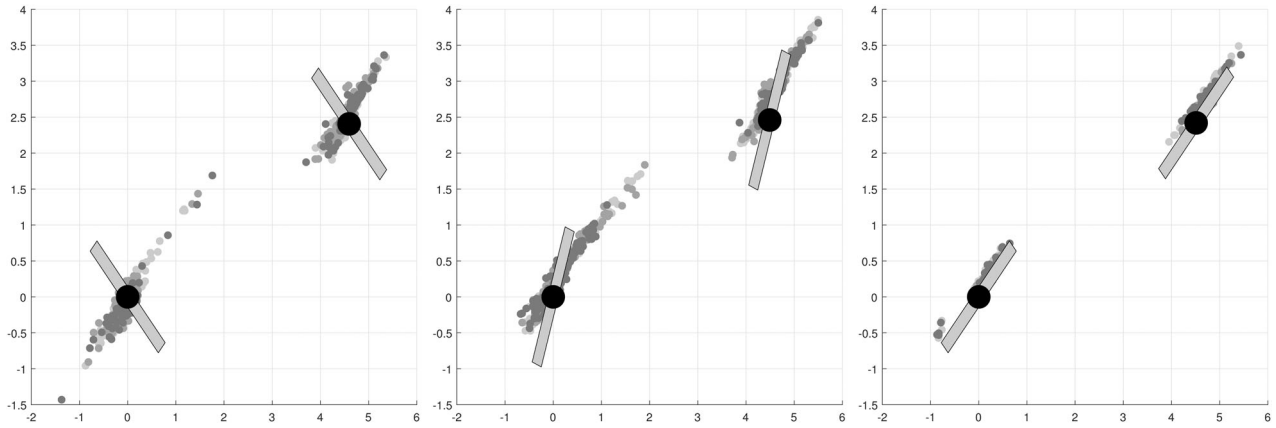

FIGURE 16 Results of the log recognition software. The gray dots are the estimated logs' locations using our algorithm. The black circle represents the logs' centers. The rectangular shapes represent the position of how the logs were placed for the tests. The units in these graphs are in meters.

(a) and (b) follow a normal distribution having a maximum peak around the vicinity of 5.2 m. For the first case, measurements have an error of 3 cm, while for the second case there is an error of approximately 10 cm. However, for the case where logs are placed at a 90° angle, the error increases to near 30 cm. Nevertheless, due to the size of the crane's grapple, the machine will still be able to grab these logs.

Similar results for the log marked as one in Figure 11 can be observed in Figure 18. The location of this log represents the origin for our calculations in this particular test case. For this test, we see that for cases (a) and (b) the error is also within a decimeter range, while it increases to 30 cm for case (c). Our results conclude that the log recognition algorithm is able to provide estimations within 10 cm of error for the cases where the camera has good visibility of the logs. The error increases to 30 cm for the worst case where camera has the minimum visibility of logs.

5.1.3 | Crane motion control

Referring to the tests described in Section 4.1.2, Figure 10 shows an example of the crane paths that our motion planning algorithm

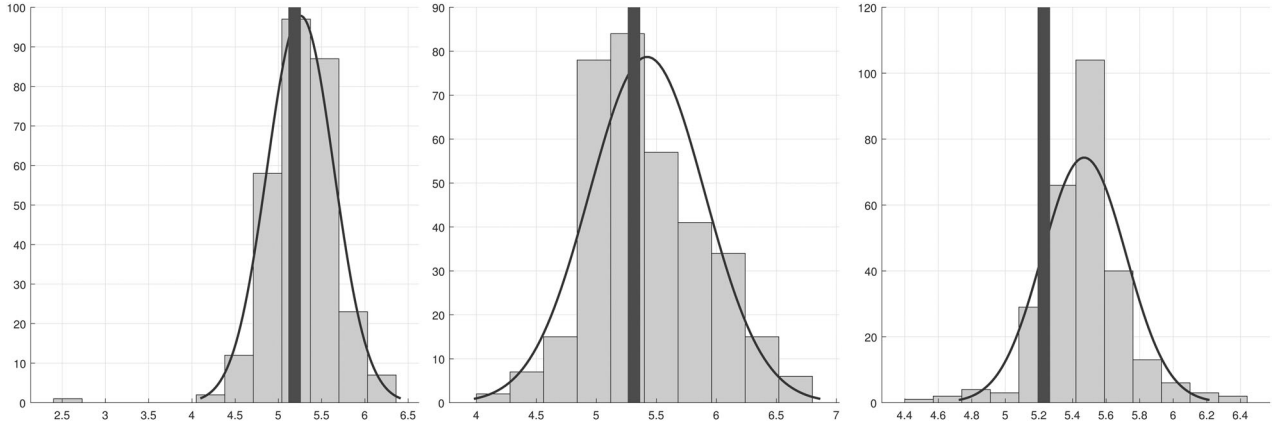

FIGURE 17 Results of the log recognition software using a histogram for log 2. The light gray bars represent the estimations, which clearly show a normal distribution in the vicinity of 5.2 m. The dark gray bar represents the position where the log is actually placed.

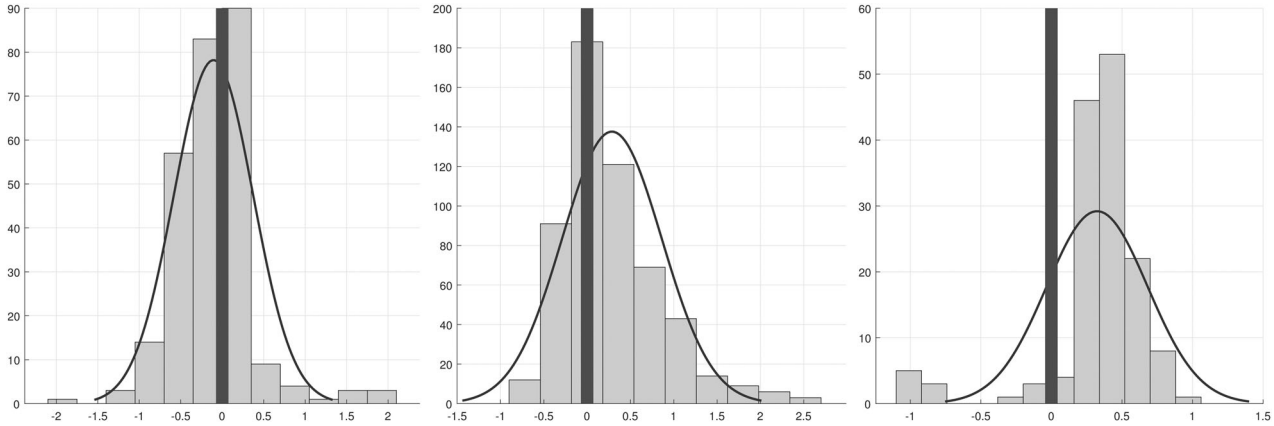

FIGURE 18 Results of the log recognition software using a histogram for log 1. The light gray bars represent the estimations, which clearly show a normal distribution in the vicinity of the origin. The dark gray bar represents the position where the log is actually placed.

defines to move the crane toward the desired goals. For this particular test case the Cartesian coordinates for the goal positions were defined as follows: Point 1 = [4.27, 1.86, 1], Point 2 = [4.27, −1.86, 1], Point 3 = , Point 4 = [6, −1.86, 1].

The motions were performed multiple times to get an estimate of the crane's motion control system's accuracy. These were performed both with the grapple empty and with a load of 200 kg. As an example, Figure 19 shows data of the Cartesian coordinates of the crane tip's location for 20 tests at each testing point, that is, 80 data points in total. The tip location was calculated by the equations of forward kinematics having values for each joint given by sensor measurements. On the left of Figure 19, we show a 3D representation of the results of all tests, while the right side shows a zoomed out view of each individual point for the *X*-*Y* axis. In all the cases, the data with a square represent the motion with an empty grapple, while the data points marked with an *x* represent the grapple with a load of 200 kg.

To show the accuracy of the crane's motion control system, we use the Euclidean distance of a data point to its desired location as a measurement of error, because this value tells how much the crane deviates from the goal location. Figure 20 presents a histogram showing these values for all data points, that is, the data for all four desired locations are piled up into a single histogram plot. The left plot is the Euclidean distance calculated in 3D, that is, using the data in the $[x, y, z]$ axis. The middle plot is the Euclidean distance calculated in 2D, that is, using the data in the $[x, y]$ axis. The right plot is the error in height, that is, z axis.

Results show that the Cartesian coordinate positioning error in 3D follows a normal distribution, having an average value of 8 cm, that is, the crane reaches the vicinity of the desired location with an average error of 8 cm. However, we see that the error in the $[x, y]$ axis is smaller on average, that is, 4 cm. This implies that the highest error in the crane positioning is the height of the tip, which deviates by an average of 6 cm in respect to the desired height. This phenomena can

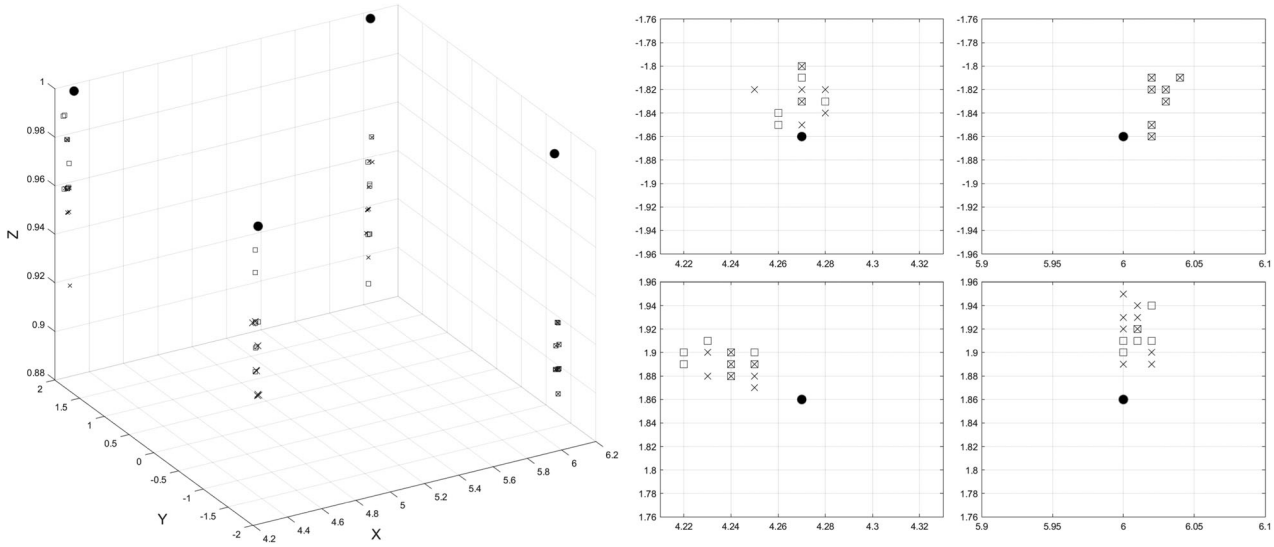

FIGURE 19 Example of results for each testing points. The square data points represent the grapple without load, while the *x* marks data points with a loaded grapple. The units for the axis in these graphs are in meters.

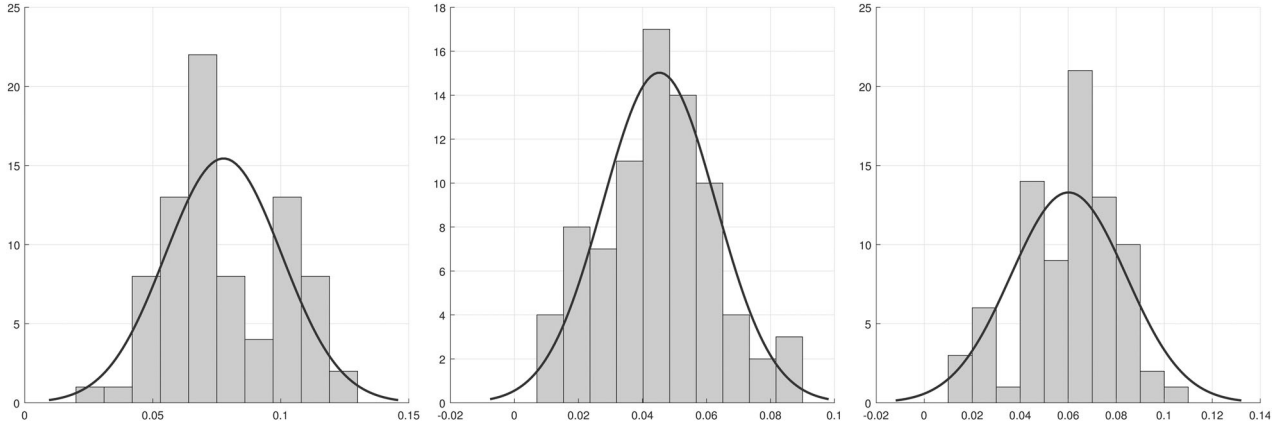

FIGURE 20 Cartesian positioning error for the crane's motion control system. From left to right, the histograms represent the error measured by the Euclidean distance to the desired location for 3D ([*x*,*y*,*z*] axis), 2D ([*x*,*y*] axis), and height (*z* axis), respectively.

be observed in the left of Figure 19, which shows that the longer the crane needs to reach the higher the deviation in the *z*-axis.

5.2 | Testing the complete forwarding task

According to the layout presented in Figure 12, a total of 24 logs were laid out within the reach of the crane along the machine's path. Table 1 shows results with a trial of all these runs. Overall, 23 out of 24 logs were detected correctly. Successful attempts were classified as those where the log was successfully loaded onto the trailer.

The reason for the failed attempts were attributed to poor image quality that lead to inaccurate log recognition, leading the crane movements to a target location unable to grasp the logs near its center point. Although there was no specific experiments to determine the tolerance to grabbing logs, we could visually determine that any log located within the maximum opening range of the grapple could be successfully grasped. The grapple has 140 cm opening range in q_6 (see Figure 2), which implies that any logs located outside this range would cause a failed attempt. As our log recognition software provided an estimation error below this value, no further analysis was performed in this regard.

6 | DISCUSSION

Forest industry plays an important role in the global economy and has significant influences in our lives and the environment that we live in. Therefore, considering improvements in technologies for forest logging operations is vital to secure wood supply with environmentally sound methods.

TABLE 1 Error measured in the *X* and *Y* directions in mm (*Y* = machine rear longitudinal direction).

Log no.	X Error	Y Error	Z Error	Euclidian Distance Error	Successful attempt	Reason for failure
1	33.7	146.6	55.2	160.3	Yes	
2	−175.9	215.8	−132.3	308.2	Yes	
3	−710.8	−120.4	181.1	743.3	No	Inaccurate estimation
4	−646.0	−23.6	−224.6	684.3	No	Inaccurate estimation
5	−295.6	−19.9	−82.2	307.5	Yes	
6	−370.2	226.7	−0.4	434.1	Yes	
7	−202.7	−82.8	−9.3	219.1	Yes	
8	−268.2	56.6	−79.2	285.3	Yes	
9	−306.5	−181.6	−28.0	357.4	Yes	
10	−292.6	222.7	−69.5	374.2	Yes	
11	−100.9	88.6	70.1	151.5	Yes	
12	−308.0	174.0	−42.2	356.3	Yes	
13	45.8	40.1	−43.3	74.7	Yes	
14	−363.6	190.5	−72.8	416.9	Yes	
15	33.2	−10.9	−7.7	35.7	Yes	
16	−271.3	298.0	−122.6	421.2	YES	
17	–	–	–	–	No	CV did not detect a log
18	−120.4	70.9	−84.8	163.4	Yes	
19	−191.1	−101.1	54.2	222.9	Yes	
20	74.1	−8.1	−94.6	120.4	Yes	
21	−402.1	−244.0	−69.0	475.4	Yes	
22	−125.0	19.2	−198.4	235.3	Yes	
23	−196.1	97.9	−36.3	222.1	Yes	
24	−47.8	73.3	−150.3	173.9	Yes	
Mean error	232.6	113.0	79.5	289.3		

With the rapid advancements of technologies related to digitalization, automation, and robotics, a transition toward more resourceful forest operations is slowly coming to fruition. Early industry adoptions of these technologies have been well received, as they provide improvements in various aspects of the work and drive the need for further advancements.

Within forestry, Scandinavian countries are well known for their high-performance machinery and the research to automate them. However, despite years of developments, projects featuring full scale demonstrations with unmanned machines are rare to find in scientific literature. This makes the development shown in this article one of a kind, because it shows the practical application of combining different research areas to successfully perform fully autonomous forestry work in real world conditions using a dedicated platform. To the best of our knowledge, this is the first time that such demonstrations have taken place in the public domain, and the success on the preliminary results motivates the vision that 1 day fully unmanned forestry machines will be able to part take the work in forestry, as envisioned by this industry. However, there is much work to be done to achieve this goal.

The development presented in this article has centered around the introduction of an unmanned forestry machine that has been purposely built as a research platform to test advanced automation for forest operations' work. Apart of the construction of the machine, that underwent 5 years, we showed successful application of three key areas, that is, autonomous navigation, log recognition via computer vision, and autonomous crane manipulation. Each of these subjects possesses its own challenge, but the initial results combining these technologies show that the accuracy of our current development is reliable enough to perform basic forwarding tasks.

6.1 | Discussion about results

To perform autonomous navigation, we have currently proposed using GPS waypoints, as this has shown to be reliable in related industries, for example, agriculture and mining. Our main reasoning comes from the fact that forest harvesting operations are preplanned before hand in Scandinavia. Therefore, having information about the routes for forwarding tasks is preliminary known and confirmed while the harvester is working in the forest. This allows our algorithms to have the information needed to plan routes that connect GPS waypoints.

Using our algorithm for navigation control, we were able to demonstrate performance properties, such as exponential stability and repeatability, as the machine was able to exponentially converge toward the same path in multiple rounds, despite starting at different initial conditions and weather conditions. Although it is difficult to show the weather influence in an article, our machine has been able to perform similarly in different weather conditions and terrains. The results in this article have been from a time where the terrain was muddy due to persistent rain fall, making it difficult to drive without sliding.

A major concern in this project was the subject of computer vision, and in particular the effectiveness of the log recognition software. As this information is needed for subsequent log manipulation via crane motion control, the accuracy on the estimation of the logs' pose and location is highly important. Results show that our system is currently able to estimate logs with an averaged deviation error between 3 and 10 cm, when the camera has good visibility of the logs. This error worsens if the logs ahead of the machine are placed nearly parallel to the machine's direction of travel, because this does not let the camera observe the logs' shape properly. This result is quantitatively sufficient for our development, because logs that have been cut-to-length are much bigger in magnitude in relation to the estimation error.

Unlike industrial robot manipulators, highly accurate control of forestry cranes is challenging, due to size, weight, and hydraulic manipulator dynamics. In heavy duty hydraulic applications, we often need to make a compromise between dynamic performance and control accuracy. In these regards, our crane motion control system provides the ability to perform smooth motions with reliable accuracy. According to results, the expected deviation from the target goal shows to be within a decimeter. However, 50% of the deviation comes from inaccurate height control (z-axis), and it is attributed to the difficulty to control height due to the length and weight of the crane, as explained in former work. Nevertheless, for forestry cranes of this magnitude (9 meters reach), the deviation error is almost negligible, and in fact it works in our advantage, because it increases the certainty that the grapple will grab logs. Therefore, considering the size of the grapple, and the size of the logs that these cranes collect, our control system does not need higher accuracy.

Our current results have several strengths. First, we were able to demonstrate autonomous navigation in a terrain that is common in forestry. Second, our results show that object recognition using machine learning can provide logs' location estimation with high accuracy for subsequent manipulation through the machine's crane. Third, our results demonstrate the feasibility of collecting logs autonomously using the crane, relying only on estimations done with the help of the camera installed in front of the machine. Four, we obtained successful results performing a fully autonomous forwarding task in an scenario that is common in clear-cut. Therefore, our system setup with minimum sensing equipment already presents the ability to carry on with forwarding tasks, because of the accuracy of algorithms developed during this project.

Despite the positive results, some limitations to our current development should be addressed. First, although we attempt to show results in real forestry conditions, our experimental environment is still idealized. So far, we have concentrated on collecting single logs, due to the lack of a dedicated vision system for the crane, and the lack of sensor's to measure the grapple's motions. Most forwarding tasks involve more complex manipulation and decision making than the ones we are showcasing in these initial results. Second, one of our objectives was to demonstrate autonomous navigation in a clear-cut terrain. To achieve this goal, our system uses DGNSS, which is available in Sweden, covering the whole nation. However, other countries might not count with similar coverage, making the AORO platform currently restricted to areas having access to DGNSS reception. In addition, a clear-cut terrain tends to have obstacles that are easy to drive over without complex obstacle avoidance methods. However, higher complexity on terrains does exist in many parts of Scandinavia and the world, and should be properly tackled with obstacle avoidance algorithms. Third, the crane's motion control system currently depends on good log location and pose estimation to successfully collect logs. As long as the crane is brought to a vicinity of a log positioned within the grapple's opening range, grasping is successful. However, to perform highly accurate collection of logs demands the need for a dedicated camera that is able to help on visually-guided motion control of the crane. This would help advancing toward more complex manipulation where we need to incorporate the rotation of the grapple q_5 as well, for example, for unloading or grabbing multiple logs.

6.2 | General discussion

Automating the work of machines represents a paradigm shift in forestry. Machines in the market are already starting to exhibit automation features to facilitate the work for the operator. Upcoming developments are promising to relieved some of the work from the machines' operators hands. With the advancements in the area of robotics and automation, it is clear that this trend will continue toward removing the necessity of on-board operators. At that point, the human operator will remotely supervise the work, as it is currently done in other industries. However, there is much to be done to build an infrastructure in forestry that can meet the requirements for automation, before this vision in forestry can become a reality. In addition, development of automation technology for forestry machine is recently starting to take place.

Although we are at the early stages of making the AORO platform a system that can autonomously perform forwarding tasks, our current development demonstrates that we are reaching a point in which algorithms currently existing in robotics are robust enough to handle real world applications. Our initial results demonstrate the beginning of an unmanned machine with the ability of handling simplified work already, despite being still an ideal testing scenario.

Some of the added benefits of our development include the ability to separate the key areas of our developments in computer vision, autonomous navigation, and autonomous crane motion control. Individually, they are the essence for market products featuring semi-autonomous functions that can relief some work from machine operators' hands in the near future. However, we refrain from discussing such topic further, as this has been explained in some of our former work.

In conclusion, this article has provided an initial view of results from the AORO platform aiming at performing fully autonomous forwarding tasks. Results show that our algorithms in the key areas of computer vision, autonomous navigation, and crane motion control, meet our expected requirements in terms of accuracy and reliability. Consequently, our activity manager software is able to combine these functions to successfully perform idealized forwarding tasks in real world conditions. These results highlight the possibility that in the future, further advancements will be able to tackle forwarding tasks with comparable performance to human operators.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support of the Kempe Foundation and The Swedish Energy Agency through the projects JCK-1713 and HAFSBIT no. 48003-1, respectively. This study was partly financed by the Swedish Foundation for Strategic Environmental Research MISTRA (program Mistra Digital Forest). The authors would also like to acknowledge the support of the company "The Swedish Cluster of Forest Technology" (In Swedish: skogstekniska klustret) during the development of this project. We would also like to thank Martin Sedin, Hugo Eriksson and Krzysztof Polowy, for their help performing the experimental studies.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in Tokyo Opensource Robotics Kyokai Association at https://github.com/tork-a/darknet-release. These data were derived from the following resources available in the public domain: - YOLO ROS: Real-Time Object Detection for ROS, https://github.com/leggedrobotics/darknet_ros/releases.

ORCID

Pedro La Hera http://orcid.org/0000-0002-5032-5087
Torbjörn Lindbäck http://orcid.org/0000-0002-6209-9355

REFERENCES

Ali, W., Georgsson, F. & Hellstrom, T. (2008) Visual tree detection for autonomous navigation in forest environment. In: *2008 IEEE Intelligent Vehicles Symposium, Eindhoven, Netherlands*. IEEE, pp. 560–565.

AORO. (2021) *Arctic off-road robotics lab*. Available at: https://www.skogstekniskaklustret.se/projekt/aoro [Accessed November 2021].

Automated harvesting with robots in the forest. (2020) Available from: https://web.fpinnovations.ca/automated-harvesting-with-robots-in-the-forest/ [Accessed November 2021].

Bellicoso, C.D., Gehring, C., Hwangbo, J., Fankhauser, P. & Hutter, M. (2016) Perception-less terrain adaptation through whole body control and hierarchical optimization. In: *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), Cancun, Mexico*. IEEE, pp. 558–564.

CINTOC. (2020) *We develop future technology for a circular bioeconomy*. Available from: http://cintoc.se/

CRANAB AB. (2021) *AB, FC8 model forwarder crane*. Available from: https://www.cranab.com/products/forwarder-cranes/fc8 [Accessed November 2021].

da Silva, D.Q., dosSantos, F.N., Filipe, V., Sousa, A.J. & Oliveira, P.M. (2022) Edge AI-based tree trunk detection for forestry monitoring robotics. *Robotics*, 11(6), 136.

Daniel Tenezaca, B., Canchignia, C., Aguilar, W. & Mendoza, D. (2020) Implementation of dubin curves-based RRT* using an aerial image for the determination of obstacles and path planning to avoid them during displacement of the mobile robot. In: *Developments and Advances in Defense and Security: Proceedings of MICRADS 2019*. Singapore: Springer, pp. 205–215.

EATON. (2019) *CMA advanced mobile valves*. Available from: https://www.eaton.com/us/en-us/catalog/valves/cma-advanced-mobile-valves.resources.html

Ebeaver, A.B. (2011) *The radio-controlled bio-energy harvester forest ebeaver*. Available from: http://ebeaver.se/Default.aspx [Accessed November 2021].

Eriksson, M. & Lindroos, O. (2014) Productivity of harvesters and forwarders in CTL operations in Northern Sweden based on large follow-up datasets. *International Journal of Forest Engineering*, 25(3), 179–200.

Everingham, M., Van Gool, L., Williams, C.K., Winn, J. & Zisserman, A. (2010) The Pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2), 303–338.

Fethi, D., Nemra, A., Louadj, K. & Hamerlain, M. (2018) Simultaneous localization, mapping, and path planning for unmanned vehicle using optimal control. *Advances in Mechanical Engineering*, 10(1), 1687814017736653.

Forsyth, D.A. & Ponce, J. (2003) A modern approach. *Computer Vision: A Modern Approach*, 17, 21–48.

Fortin, J.-M., Gamache, O., Grondin, V., Pomerleau, F. & Giguère, P. (2022) Instance segmentation for autonomous log grasping in forestry operations. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 6064–6071.

Gingras, J.-F. & Charette, F. (2017) Fp innovations forestry 4.0 initiative. In: *Bangor: 2017 Council on Forest Engineering Annual Meeting*. Retrieved in 2019, July 23, from http://blog.fpinnovations.ca/blog/2017/06/20/forestry-4-0-featured-as-part-of-cif-ifc-e-lecture-series/

Halme, A & Vainio, M. (1998) Forestry robotics-why, what and when. In: *Autonomous robotic systems, Lecture notes in vontrol and information sciences*, 236 vols. London: Springer. Available from: https://doi.org/10.1007/BFb0030804, pp. 149–162.

Hansson, A. & Servin, M. (2010) Semi-autonomous shared control of large-scale manipulator arms. *Control Engineering Practice*, 18(9), 1069–1076.

Hansson, L.J., Forsmark, V., Flisberg, P., Rönnqvist, M., Mörk, A. & Jönsson, P. (2022) A decision support tool for forwarding operations with sequence-dependent loading. *Canadian Journal of Forest Research*, 52(12), 1513–1526.

Holmström, J. (2020) Digital transformation of the Swedish forestry value chain: Key bottlenecks and pathways forward. Retrieved from Mistra website.

Hosseini, A., Lindroos, O. & Wadbro, E. (2019) A holistic optimization framework for forest machine trail network design accounting for multiple objectives and machines. *Canadian Journal of Forest Research*, 49(2), 111–120.

Hyyti, H., Kalmari, J. & Visala, A. (2013) Real-time detection of young spruce using color and texture features on an autonomous forest machine. In: *The 2013 International Joint Conference on Neural Networks (IJCNN), Dallas, TX, USA*. IEEE, pp. 1–8.

Jelavic, E., Jud, D., Egli, P. & Hutter, M. (2022) Robotic precision harvesting: mapping, localization, planning and control for a legged tree harvester. *Field Robotics*, 2, 1386–1431.

Johns, R.L., Wermelinger, M., Mascaro, R., Jud, D., Gramazio, F., Kohler, M., Chli, M. & Hutter, M. (2020) Autonomous dry stone. *Construction Robotics*, 4(3), 127–140.

Keefe, R.F., Wempe, A.M., Becker, R.M., Zimbelman, E.G., Nagler, E.S., Gilbert, S.L. & Caudill, C.C. (2019) Positioning methods and the use of location and activity data in forests. *Forests*, 10(5), 458.

Komatsu Forest AB. (2017) *Komatsu smartflow—new hydraulic system for forwarder*. Available from: https://www.forestry.com/editorial/equipments/komatsu-smartflow-hydraulic-system-forwarder/ [Accessed 2017].

La Hera, P. & Morales, D.O. (2014) Non-linear dynamics modelling description for simulating the behaviour of forestry cranes. *International Journal of Modelling, Identification and Control*, 21(2), 125–138.

La Hera, P., Morales, D.O. & Mendoza-Trejo, O. (2021) A study case of dynamic motion primitives as a motion planning method to automate the work of forestry cranes. *Computers and Electronics in Agriculture*, 183, 106037.

La Hera, P. & Ortíz Morales, D. (2015) Model-based development of control systems for forestry cranes. *Journal of Control Science and Engineering*. 2015, 256951. Available from: https://doi.org/10.1155/2015/256951

La Hera, P.M., Mettin, U., Westerberg, S. & Shiriaev, A.S. (2009) Modeling and control of hydraulic rotary actuators used in forestry cranes. In: *2009 IEEE International Conference on Robotics and Automation, Kobe, Japan*. IEEE, pp. 1315–1320.

Lawal, M.O. (2021) Tomato detection based on modified yolov3 framework. *Scientific Reports*, 11(1), 1–11.

Li, S. & Lideskog, H. (2021) Implementation of a system for real-time detection and localization of terrain objects on harvested forest land. *Forests*, 12(9), 1142.

Li, S. & Lideskog, H. (2023) Realization of autonomous detection, positioning and angle estimation of harvested logs. *Croatian Journal of Forest Engineering*, 44(2), 369–383.

Lideskog, H., Karlberg, M. & Bergsten, U. (2015) Development of a research vehicle platform to improve productivity and value-extraction in forestry. *Procedia CIRP*, 38, 68–73.

Lindroos, O., LaHera, P. & Häggström, C. (2017) Drivers of advances in mechanized timber harvesting-a selective review of technological innovation. *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, 38(2), 243–258.

Lindroos, O., Mendoza-Trejo, O., LaHera, P. & OrtizMorales, D. (2019) Advances in using robots in forestry operations. In: University of Southern Queensland, Australia & Billingsley, J. (Eds.) *Burleigh Dodds Series in Agricultural Science*. Burleigh Dodds Science Publishing, pp. 233–260.

Liu, M., Han, Z., Chen, Y., Liu, Z. & Han, Y. (2021) Tree species classification of lidar data based on 3d deep learning. *Measurement*, 177, 109301.

Lundbäck, M., Häggström, C. & Nordfjell, T. (2021) Worldwide trends in methods for harvesting and extracting industrial roundwood. *International Journal of Forest Engineering*, 32(3), 202–215.

Lundmark, H., Josefsson, T. & Östlund, L. (2017) The introduction of modern forest management and clear-cutting in Sweden: Ridö state forest 1832-2014. *European Journal of Forest Research*, 136, 1–17.

Manner, J., Mörk, A. & Englund, M. (2019) Comparing forwarder boom-control systems based on an automatically recorded follow-up dataset. *Silva Fenn*, 532, 10161.

Manring, N.D. (2005) *Hydraulic control systems*, 1st edition, New York, NY: John Wiley & Sons.

Morales, D.O., LaHera, P., Westerberg, S., Freidovich, L.B. & Shiriaev, A.S. (2014) Path-constrained motion analysis: an algorithm to understand human performance on hydraulic manipulators. *IEEE Transactions on Human-Machine Systems*, 45(2), 187–199.

Münzer, M.E. (2004) *Resolved motion control of mobile hydraulic cranes, Ph.D. thesis*. Aalborg University, Institute of Energy Technology.

Nguyen, H.T., LopezCaceres, M.L., Moritake, K., Kentsch, S., Shu, H. & Diez, Y. (2021) Individual sick fir tree (*Abies mariesii*) identification in insect infested forests by means of UAV images and deep learning. *Remote Sensing*, 13(2), 260.

Nordfjell, T., Öhman, E., Lindroos, O. & Ager, B. (2019) The technical development of forwarders in Sweden between 1962 and 2012 and of sales between 1975 and 2017. *International Journal of Forest Engineering*, 30(1), 1–13.

Nurminen, T., Korpunen, H. & Uusitalo, J. (2006) Time consumption analysis of the mechanized cut-to-length harvesting system. *Silva Fennica*, 40(2), 346. Available from: https://doi.org/10.14214/sf.346

Oliveira, L.F., Moreira, A.P. & Silva, M.F. (2021) Advances in forest robotics: A state-of-the-art survey. *Robotics*, 10(2), 53.

OrtizMorales, D., Westerberg, S., LaHera, P.X., Mettin, U., Freidovich, L. & Shiriaev, A.S. (2014) Increasing the level of automation in the forestry logging process with crane trajectory planning and control. *Journal of Field Robotics*, 31(3), 343–363.

Pagnussat, M., Hauge, T., Silva Lopes, E.d., Martins de Almeida, R.M., Naldony, A. (2020) Bimanual motor skill in recruitment of forest harvest machine operators. *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, 41(1), 25–33.

Park, Y., Shiriaev, A., Westerberg, S. & Lee, S. (2011). 3D log recognition and pose estimation for robotic forestry machine. In: *2011 IEEE International Conference on Robotics and Automation, Shanghai*. IEEE, pp. 5323–5328.

Phiquepal, C., Orthey, A., Viennot, N. & Toussaint, M. (2022) Path-tree optimization in discrete partially observable environments using rapidly-exploring belief-space graphs. *IEEE Robotics and Automation Letters*, 7(4), 10160–10167.

Purfürst, F.T. (2010) Learning curves of harvester operators. *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, 31(2), 89–97.

Rånman, H. (2015) Utveckling av kontaktorgan till terrängdrönare.

Rakkatec, A.B. (2021) Unmanned ground vehicles for the most demanding conditions. Available from: https://rakkatec.com [Accessed November 2021].

Redmon, J. & Farhadi, A. (2018) YOLOv3: an incremental improvement (cite arxiv:1804.02767Comment: Tech Report).

Redmon, J. Darknet: Open Source Neural Networks in C. (2021) *Darknet: open source neural networks in C*. Available from: https://pjreddie.com/darknet/ [Accessed November 2021].

Reitz, J., Schluse, M. & Roßmann, J. (2019) Industry 4.0 beyond the factory: An application to forestry. In: *Tagungsband des 4. Kongresses Montage Handhabung Industrieroboter*. Vieweg, Berlin, Heidelberg: Springer, pp. 107–116.

Roslan, Z., Awang, Z., Husen, M.N., Ismail, R. & Hamzah, R. (2020) Deep learning for tree crown detection in tropical forest. In: *2020 14th*

International Conference on Ubiquitous Information Management and Communication (IMCOM), Taichung, Taiwan. IEEE, pp. 1–7.

Salton, G. & McGill, M.J. (1983) *Introduction to modern information retrieval*. McGraw-Hill Book Company. Printed in the United States of America.

Shorten, C. & Khoshgoftaar, T.M. (2019) A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1–48.

Sihvo, S., Virjonen, P., Nevalainen, P. & Heikkonen, J. (2018) Tree detection around forest harvester based on onboard lidar measurements. In: *2018 Baltic Geodetic Congress (BGC Geomatics), Olsztyn, Poland*. IEEE, pp. 364–367.

Spong, M., Hutchinson, S. & Vidyasagar, M. (2006) *Robot modeling and control*. NJ: John Wiley and Sons.

Technion Onlinesince. (2017) *Crane control systems*. Available from: https://technion.fi/crane-control-systems/ [Accessed 2023].

The death of the forest Beast. (2006) Available from: https://www.forest-monitor.com/en/death-forest-beast/ [Accessed November 2021].

The Mathworks. MATLAB/Simulink (1990) Available from: http://www.mathworks.com

The world's first self-propelled forestry machine. (2021) Available from: https://gamingsym.in/the-worlds-first-self-propelled-forestry-machine-removes-man-from-the-equation/ [Accessed November 2021].

United Electronic Industries (UEI). (2021) *Rack mountable Simulink/RTW target, ideal for HIL applications*. Available from: https://www.ueidaq.com [Accessed November 2021].

Visser, R. & Obi, O.F. (2021) Automation and robotics in forest harvesting operations: identifying near-term opportunities. *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, 42(1), 13–24.

Wells, L.A. & Chung, W. (2023) Real-time computer vision for tree stem detection and tracking. *Forests*, 14(2), 267.

Westerberg, S. (2014) *Semi-automating forestry machines: motion planning, system integration, and human-machine interaction*. Ph.D. thesis, UmeÅ Universitet.

Wu, J., Yang, Q., Bao, G., Gao, F., et al. (2009) Algorithm of path navigation line for robot in forestry environment based on machine vision. *Nongye Jixie Xuebao= Transactions of the Chinese Society for Agricultural Machinery*, 40(7), 176–179.

Yarak, K., Witayangkurn, A., Kritiyutanont, K., Arunplod, C. & Shibasaki, R. (2021) Oil palm tree detection and health classification on high-resolution imagery using deep learning. *Agriculture*, 11(2), 183.

YOLO ROS: Real-Time Object Detection for ROS. (2021) Available from: https://github.com/leggedrobotics/darknet-ros [Accessed November 2021].

Zoto, J., Musci, M.A., Khaliq, A., Chiaberge, M. & Aicardi, I. (2020) Automatic path planning for unmanned ground vehicle using UAV imagery. In: *Advances in Service and Industrial Robotics: Proceedings of the 28th International Conference on Robotics in Alpe-Adria-Danube Region (RAAD 2019) 28*. Cham: Springer, pp. 223–230.

How to cite this article: La Hera, P., Mendoza-Trejo, O., Lindroos, O., Lideskog, H., Lindbäck, T., Latif, S. et al. (2024) Exploring the feasibility of autonomous forestry operations: results from the first experimental unmanned machine. *Journal of Field Robotics*, 1–24. https://doi.org/10.1002/rob.22300