

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Deep Learning-Based Autonomous Excavation: A Bucket-Trajectory Planning Algorithm

JAEMYUNG HUH¹, JANGHO BAE², DONGHYEON LEE¹, JAEWON KWAK¹, CHANYOUNG MOON¹, CHULWHAN IM¹, YEONHO KO¹, TAE KOO KANG³, AND DAEHIE HONG¹

¹School of Mechanical Engineering, Korea University, Seoul 02841, Republic of Korea

²Department of Mechanical and Automotive Engineering, KyungSung University, Republic of Korea

³Department of Human Intelligence and Robot Engineering, Sangmyung University, Cheonan, Republic of Korea

Corresponding authors: Daehie Hong (e-mail: dhong@korea.ac.kr), Tae Koo Kang (e-mail: tkkang@smu.ac.kr)

This work was supported by the Technology Innovation Program (or Industrial Strategic Technology Development Program) (20010776, Development of Collaborated Self Control System and Unmanned Excavator Technology for an Automated Earthworks) funded By the Ministry of Trade, Industry and Energy(MOTIE, Korea) (*Corresponding authors: Daehie Hong; Tae Koo Kang*)

ABSTRACT The increased risk to the safety of excavator personnel and difficulty in training them, combined with a manpower shortage, have led to an increased demand for machine automation. This study applies a long short-term memory algorithm for automating a bucket-tip trajectory planning AI system. Unlike other autonomous excavation techniques, the proposed approach in this study performs the bucket-trajectory planning of the excavator without prior knowledge of nonlinear bucket-soil interaction dynamics during excavation, which requires precise adjustment of parameters with the heuristic analysis of correlation between them. Based on data acquisition from excavation of excavator experts, this method uses the three-dimensional point cloud of terrain and bucket motion data in excavation process for training and application of the AI. Especially, we transform the point cloud, which comprises massive number of points and increases the computation complexity, into the much smaller number of values, which are sufficient for representing the three-dimensional shape of target terrain. To ensure safety against collision with underground obstacles, a collision avoidance algorithm is applied to prevent crashes during excavations along the given path, based on continuous monitoring of the pressure in the excavator's hydraulic cylinder. Comparative experiments reveal that the bucket-tip trajectory planning AI system with the collision avoidance algorithm generates a traceable trajectory for the machine controller, equipped in the excavator, and yields the desired excavation volume and lead time without collision, regardless of the topographic change caused through successive excavations.

INDEX TERMS Automated excavation, bucket-tip trajectory planning, collision avoidance algorithm, field robots, long short-term memory, point cloud data

I. INTRODUCTION

The global demand for excavators has soared owing to the machine's operational versatility, including tasks such as excavation, loading, and drilling. Thus, excavators are essential heavy machinery for construction sites. However, the supply of workforce required to operate this construction equipment has stagnated owing to the hazardous nature of the job. In particular, the time required to train new personnel on the machine and make them proficient is more than 5 years. Moreover, the population of skilled operators has been declining owing to ageing of the workforce (retirement).

Therefore, the construction industry is suffering from a shortage of workforce.

Thus, various researches from estimation of the soil–bucket interaction force to trajectory tracking and autonomous excavation have been conducted to address a shortage of personnel. In the field of excavation automation, several studies have investigated the interaction dynamics between soil and the tool or bucket. For example, certain researchers studied estimation of soil information based on the energy dissipated during soil–tool interaction by measuring the force and displacement of the bucket [1].

Dynamic models based on Newton–Euler equations [2] and soil–tool interactions [3] were used to predict the resistive forces exerted on the bucket during excavation. Furthermore, these predicted values were fed to the controller to plan the trajectory of the bucket [4], [5]. Additionally, other studies attempted to apply the fuzzy methods for identifying soil property parameters and the fundamental earthmoving equation for predicting soil resistive forces during excavation [6].

Despite these efforts to mathematically represent the soil–tool interaction, the non-linearity of soil dynamics complicates the bucket's motion control. Additionally, because the interaction dynamics depends on the bucket size and soil type, the controller needs to be manually tuned according to the work site and excavator model. However, such customization is time-consuming. Therefore, recent research on autonomous excavation has focused on the application of machine learning and deep learning, which do not require information on the complex dynamics between soil and bucket motion. Researchers [7] have attempted to replace existing control methods such as proportional–integral–derivative (PID) control with a reinforcement learning (RL)-based trajectory-tracking controller, which generates valve commands to supply hydraulic power to operate the excavator's arm. In [8] and [9], given the desired trajectory of an excavator's links, impedance and sliding mode controllers were leveraged to calculate desired cylinder forces. Subsequently, echo state networks used these forces and cylinder pressure values as inputs to control servo valves and force the bucket to follow a set course.

These examples of artificial intelligence (AI) focused on the trajectory-tracking performance to follow a set trajectory rather than the method used to generate the desired path. However, considering expert's manipulation for existing tracking methods is a difficult task, which focuses on expert's intuition on excavation state rather than calculation of excavation force. To automate the excavator, the automatic generation of the bucket-tip trajectory including the expert's experience of excavation must be prioritized. Accordingly, later studies developed AIs that can imitate manipulation of experts or generate the total path of the bucket by referring to in-situ data. Ref. [10] used the neural networks to determine the soil type and accordingly design a path for the bucket. Moreover, researchers [11] have developed an automatic bucket-filling algorithm for a wheel loader based on the time-delayed neural network that used measurement of encoders and pressure transducers to imitate the manipulation of experts in the “bucket filling” part of the entire excavation process. In [12], with the topology of soil surface as input, the heteroscedastic gaussian process was developed and applied to an electric manipulator for autonomous excavation. The method accurately determines the switching time from the drag phase to the scoop phase to obtain the desired excavation volume. In [13], the multi-layer perceptron was used to generate dataset of shaping forces, which were transformed into the point set of bucket trajectory using the dynamic

movement primitives. Furthermore, in [14], the deep reinforcement learning (DRL) was used to develop the bucket tip trajectory. Data for training the DRL were secured from a robotic simulation program. After learning, the DRL of the excavator can output each control action for excavation.

However, the control actions determined by the AIs in [10], [11], and [12] were extremely restrictive in certain parts of the excavation process, thereby limiting their application for different terrain shapes. Regarding trajectory planning based on force data set in [13], force configuration during excavation will differ in other fields with distinct soil hardness, even if the path is identical. Moreover, the configuration will require different tuning for transformation into the point's coordinate value against the different force set. The DRL has been frequently developed and applied in robotics, because the computing performance has advanced considerably in terms of memory and computing speed, as reported in [7], [14], [15], [16], and [17]. Generally, the DRL requires numerous trials and errors for training the control policy to output the Q value for each action. Because repetitive trials and errors are difficult to perform in practice owing to safety risks, several studies regarding excavators have used computer simulation to perform multiple trials and errors and store data of each excavation in the replay buffer [14], [15]. However, in virtual simulation, integration with soil dynamics, including the soil–tool interaction and the configuration of soil pile and particle, not only requires the knowledge of soil physical features, but also, depending on the size and number of soil particle, enhances computational complexity. Furthermore, in the DRL, the reward function, which yields a reward to the agent depending on the agent's action based on states, requires a heuristic approach for its configuration and needs to thoroughly consider the relation between the control action and the state of excavation.

To address limitations and complexity of prior researches, we develop a trajectory planning system for the bucket tip's position and orientation, based on the expert's excavation data, which is composed of the point cloud of target terrain before excavation and position values of the bucket tip during excavation. The bucket-tip trajectory-planning system consists of two AIs. The first AI is used to generate the region of excavation with consideration of safety against collision, and the second AI is utilized to locate waypoints inside that region. These two AIs include the long short-term memory (LSTM) algorithm as main layer. The LSTM is an advanced recurrent neural network (RNN), and suitable for training with one-dimensional (1D) long-sequence dataset. For configuration of the 1D input dataset, we transform the point cloud into 3 values, which represent the terrain features. In addition to the bucket-tip trajectory planning AI system, we also develop a collision avoidance algorithm to prevent the bucket from crashing into underground obstacles and adjust the excavation range with consideration of safety. With the aid of these algorithms, this study proposes robust excavation against topographic changes via successive excavation and with safe excavation against underground obstacle, without

TABLE 1. Specifications of experimental excavator

Description	Specification
Scale	1/8
Size (mm)	952 × 295 × 272
Hydraulic pressure of pump	4 MPa
Power (Boom/Arm/Bucket)	Hydraulic
Power (Swing)	Electric

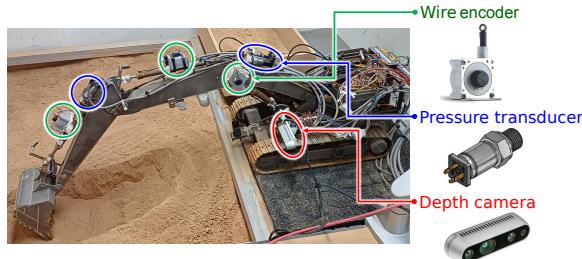


FIGURE 1. Excavator and sensors used for experimental verification

using knowledge of soil dynamics. To validate the robustness, we conduct performance comparison tests with experts in terrain changed in each trial, and the collision avoidance algorithm is demonstrated in simulated environment.

II. METHOD

A. EXPERIMENTAL EXCAVATOR HARDWARE SETUP

The proposed trajectory planning algorithm is applied to a 1/8-scale excavator (Table 1, Fig. 1). Boom, arm, and bucket are rotated through linear actuation of hydraulic cylinders, to which hydraulic pressure are supplied via the hydraulic pump. The rotation of pump and swing actuator is controlled via the electronic speed control using pulse width modulation (PWM) signals from the controller PC. Servo valves for operating hydraulic cylinders are also controlled by PWM signals. The velocity of the cylinder is controlled using the open/close ratio of servo valve, which is proportionate to the duration of the PWM “ON” signal, with each period spanning 20 ms.

To establish the automatic excavation system, we install wire encoders, pressure transducers, and the depth camera on the excavator (Fig. 1). Wire encoders can measure the stroke of the hydraulic cylinders, which will be transformed into rotation angles of boom, arm, and bucket joints. Two pressure transducers are installed at each end of a hydraulic cylinder, and the difference between two pressure values of each cylinder is proportional to the load on that cylinder. The depth camera is installed on the cabin to obtain the point cloud of terrain.

The excavator’s workspace can stretch to approximately 1 m from the boom joint, but, in this study, we limit its range at 0.95 m from the boom joint to prevent tipping of the excavator owing to heavy load during deep excavation.

B. EXPERIMENTAL CONTROLLER SETUP

The control system of the excavator comprises 3 computers, including the AI computer, the machine controller, and the host computer (Fig. 2).

The AI computer is equipped with an NVIDIA RTX 3090 GPU for rapid training and application of bucket trajectory planning based on the deep learning. Moreover, it collects the point cloud data via direct connection with the depth camera, thereby minimizing the data loss during transfer.

As the machine controller of excavator, we use the National Instruments’ compact RIO, which gathers sensor data from the operation of excavator and transmits the control input to manipulate machine links in real time. These input and output data flow through I/O modules installed on the compact RIO. The operation of compact RIO is controlled and managed through NI LABVIEW’s virtual instruments operating in the real-time (RT) module and the field-programmable gate array (FPGA).

The host computer is connected to the AI computer via the user datagram protocol communication method. The cable transmits the measured position and angle information of the bucket tip and receives the next position and angle data, which are calculated by the AI computer. Virtual instrument panels for RT and FPGA modules can be operated in the host computer. These panels employ the next position and angle values to calculate PWM “ON” duration signals for controlling 3 links of the excavator and transmit these control signals to the three servo valves via I/O modules of the compact RIO. Additionally, two joysticks are plugged to the host computer to manipulate the excavator.

C. TEST SIMULATOR

To secure the training dataset from the expert’s operation, we construct a personnel test simulator (Fig. 3), with the cockpit seat of the excavator. The soil in this test is decomposed granite soil. The excavator is installed on a workbench, and experts manipulate it, using joysticks of the cockpit seat. The simulator uses the same joystick manipulation method as real excavator. Experts use buttons of joysticks to initiate and complete the saving of data measurements. After each round of excavation, experts perform swing and soil loading operations on a box, which is located on a weighing scale to measure the weight of excavated soils. After the loading process, the test manager records the lead time for excavation and the excavated soil weight. In case of an emergency, such as unexpected manipulation and motion of the excavator, the test manager can push the kill switch to cut off electricity to the actuation system and sensors of the excavator.

D. POSITION CONTROL FOR EXCAVATION MOTION

The AI system generates waypoints along the bucket tip trajectory. After transmitting the point’s position data from the AI computer, the machine controller conducts the position control based on the point-to-point method. The information of a point is composed of the distance, depth, and angle of attack of the bucket tip, which are calculated and measured

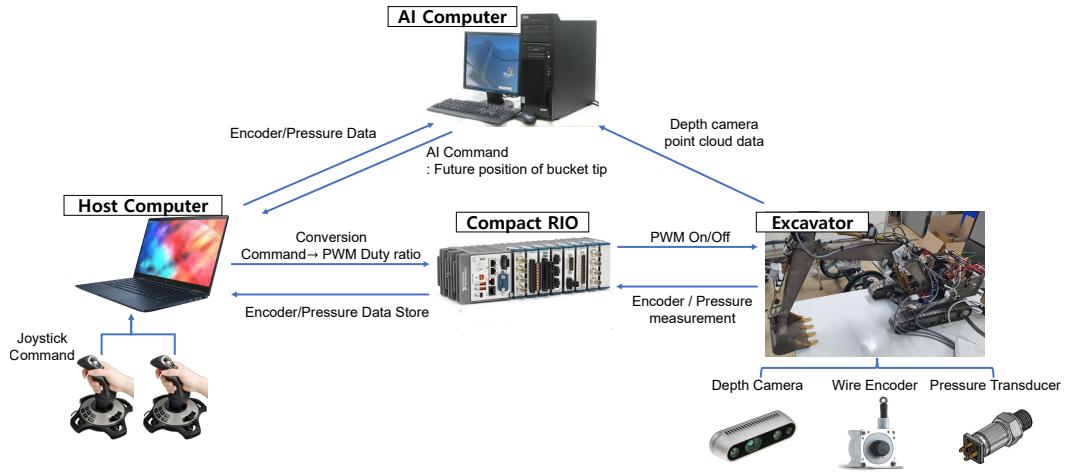


FIGURE 2. Excavator controllers: Host computer, AI computer, and CompactRIO

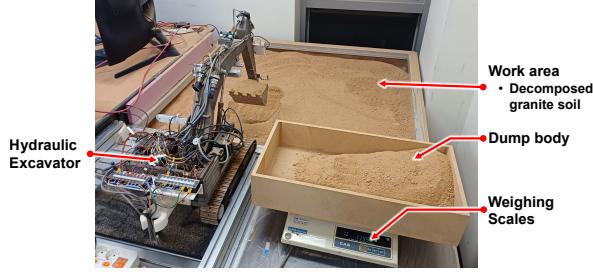


FIGURE 3. Expert Test Bed

TABLE 2. Denavit-Hartenberg parameters for excavator

Link	Link length	Link twist	Link offset	Joint angle
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

relatively to the revolute joint of the excavator's boom as origin. The machine controller transforms these values into variables of excavator's joints, using the Denavit-Hartenberg (DH) configuration of Table. 2 and Fig. 4 and inverse kinematics. The degree of freedom (DOF) of excavator's manipulator arm is 3, which is composed of 3 revolute joints, except for the swing joint fixed during excavation in this study. First, as (1), using angle of bucket tip, the sum of 3 joint angles is obtained. For a simple representation of θ_1 and θ_2 , we calculate the position of bucket joint using the angle and length between the bucket joint and the tip, as expressed in (2). Subsequently, in accordance with the law of cosines, as expressed in (3) and (4), θ_1 and θ_2 can be calculated with the length of arm and boom and the position of bucket joint. Thus, θ_3 is calculated based on (1) with θ_1 and θ_2 .

$$\theta_{123} = \theta_1 + \theta_2 + \theta_3 = \pi - \theta_{\text{bucket}} \quad (1)$$

$$p_x' = p_x - a_3 c_{123}, \quad p_y' = p_y - a_3 s_{123} \quad (2)$$

$$\theta_1 = \pi - \arctan \frac{p_y'}{p_x'} - \arccos \frac{a_1^2 + (p_x')^2 + (p_y')^2 - a_2^2}{2a_1 \sqrt{(p_x')^2 + (p_y')^2}} \quad (3)$$

$$\theta_2 = \pi - \arccos \frac{a_1^2 + a_2^2 - (p_x')^2 - (p_y')^2}{2a_1 a_2} \quad (4)$$

After the process of inverse kinematics, 3 joint variables are transformed into lengths of hydraulic cylinders of boom, arm, and bucket. In Figs. 5 and 6, hydraulic cylinders of boom and arm are on opposite sides across from angles of revolute joints, except for the bucket joint illustrated in Fig. 7, which is equipped with a 4-bar linkage. $L_{\text{cyl}1}$ and $L_{\text{cyl}2}$, which represent cylinder lengths of arm and boom, are calculated in accordance with the law of cosines and angles calculated previously in inverse kinematics, as shown in (5) and (6).

$$\pi - \theta_1 + \theta_{c1} + \theta_{c2} = \arccos \frac{L_{O_0a}^2 + L_{O_0b}^2 - L_{\text{cyl}1}^2}{2L_{O_0a}L_{O_0b}} \quad (5)$$

$$\begin{aligned} \theta_{c4} &= \pi - \theta_{c3} - (\theta_{c5} - \theta_2) \\ &= \arccos \frac{L_{O_1c}^2 + L_{O_1d}^2 - L_{\text{cyl}2}^2}{2L_{O_1c}L_{O_1d}} \end{aligned} \quad (6)$$

To determine the length of the bucket cylinder $L_{\text{cyl}3}$ from θ_3 , which is also calculated in inverse kinematics, we initially calculate the sum of θ_{c9} and θ_{c10} , as shown in (7). Thereafter, the 4-bar linkage near the bucket joint is divided into 2 triangular sections : $\triangle O_2gf$ and $\triangle O_2hf$. Thus, θ_{c9} and θ_{c10} are determined based on the law of sines and cosines in those two triangles. Moreover, by calculating these two angles, the length of L_{O_2f} is determined, and the length of bucket cylinder ($L_{\text{cyl}3}$) is employed in this process, as shown in (10). By implementing the inverse sequence of this process, $L_{\text{cyl}3}$ is obtained from θ_3 .

$$\pi - \theta_{c8} - \theta_{c11} - \theta_3 = \theta_{c10} + \theta_{c9} \quad (7)$$

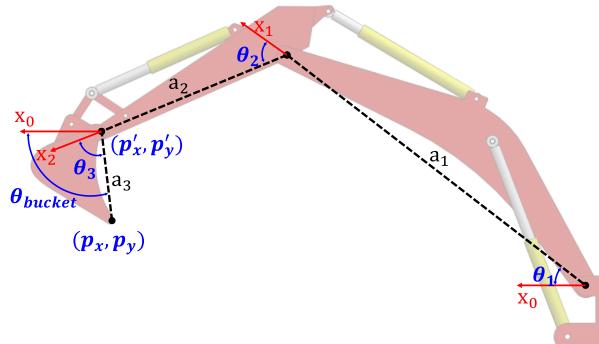


FIGURE 4. 3-DOF robotic system of excavator

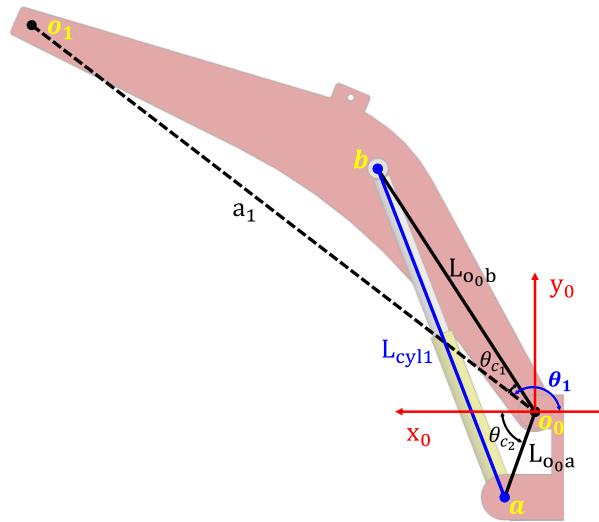


FIGURE 5. Configuration of boom joint and link

$$\theta_{c_9} = \arcsin \frac{L_{gf} \sin \theta_{c_7}}{L_{O_2f}} \quad (8)$$

$$\theta_{c_{10}} = \arccos \frac{L_{O_2h}^2 + L_{O_2f}^2 - L_{fh}^2}{2L_{O_2h}L_{O_2f}} \quad (9)$$

$$L_{O_2f} = \sqrt{L_{O_2g}^2 + L_{gf}^2 - 2L_{O_2g}L_{gf} \cos \theta_{c_7}} \quad (10)$$

$$\theta_{c_7} = \pi - \arccos \frac{L_{gf}^2 + L_{eg}^2 - L_{cyl3}^2}{2L_{gf}L_{eg}}$$

As mentioned earlier, the PWM “ON” duration signal is based on the difference between the target and current length of cylinder. This difference in each cylinder is used as input to PID controllers, which output the PWM signals. In summary of this chapter, the control system, including the AI system, is displayed in Fig. 8

E. LONG SHORT-TERM MEMORY

The RNN is suitable for processing 1D sequence dataset. It focuses on the relation inside elements of the input vector and their sequence using directional connection between RNN cells in the hidden layer. In each step of the RNN, the hidden state of current cell receives the current input and the hidden

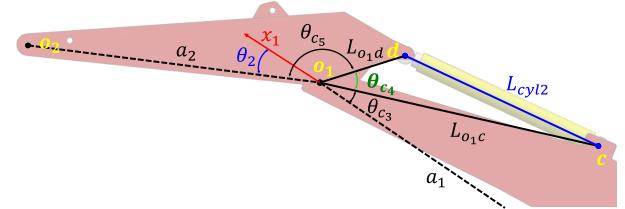


FIGURE 6. Configuration of arm joint and link

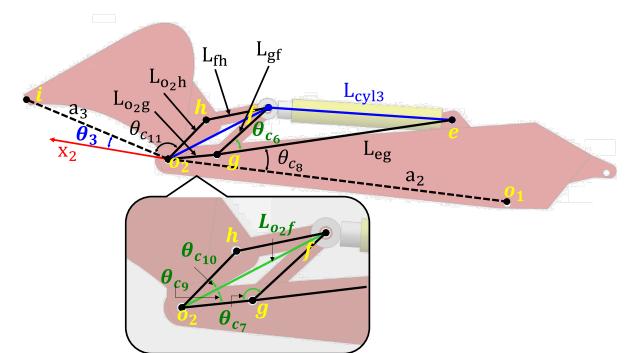


FIGURE 7. Configuration of bucket joint and link

state of prior cell, and is calculated as output. In this process, parameters, including weights and biases for calculation of hidden state, are shared across the sequence. Thus, the RNN is advantageous for determining features from the sequence with less computational complexity.

However, the RNN’s learning result is considerably influenced by the weight between cells, because gradient of loss function is proportional to iterative multiplication of weight between two cells, as reported in [18]. For example, if the weight value is larger than 1, the gradient will be exponentially large during back-propagation through Time (BPTT). In contrast, if the weight is smaller than 1 and the length of input sequence is large, then the gradient can vanish during BPTT.

Owing to the instability of learning process of the RNN, we employed the LSTM, which is an improved RNN, as foundation of the AI system for the bucket-trajectory planning in this study. The LSTM’s memory cell replaces the RNN’s cell and generates the hidden state \$h_t\$, called the short-term memory, and the cell state \$C_t\$, called the long-term memory, using the current input, the cell state and the hidden state of prior step as input, as depicted in Fig. 9. During the update of states, the memory cell utilizes sigmoid gates and a tangent layer. The sigmoid gates are composed of forget, input, and output gates. The forget gate determines values that will be forgotten or transmitted from the prior cell state. The input gate determines values that will be selected for the cell state update, among calculation results of the tanh layer, which uses a combination of the current input and the prior hidden state as input. The output gate determines values that will become the current hidden state after calculating

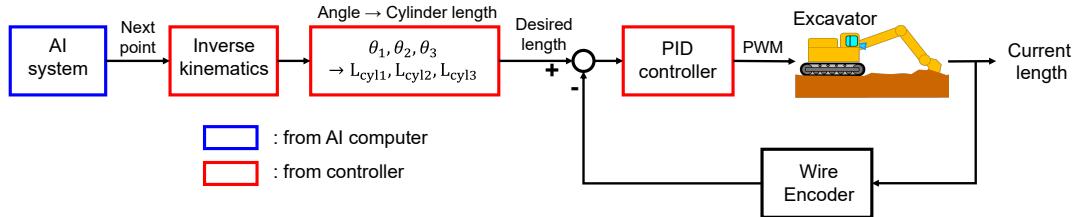


FIGURE 8. Block diagram of excavator control system

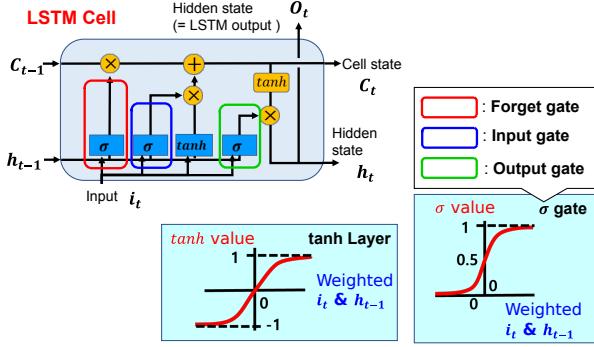


FIGURE 9. Structure of long short-term memory (LSTM)

the $tanh$ function using the current (updated) cell state as input. During this process, the prior cell state is transmitted in its original form. This strategy ensures that the LSTM is not affected by vanishing or exploding of the gradient. Thus, through parameter sharing and stable gradient for training networks, the LSTM is an appropriate algorithm to generate the output vector from the 1D long-input dataset.

F. BUCKET-TIP TRAJECTORY-PLANNING AI SYSTEM

The bucket-tip trajectory planning AI system is used as the framework to generate the base trajectory of bucket and produce the angle of bucket along the generated trajectory. This system is created based on ideas of the instance segmentation method, which includes the generation algorithm for the region-of-interest (ROI) box near the target object and the drawing algorithm of segmentation mask to separate it from the background image. Similarly, the first AI, called the region-of-excavation (ROE) generation AI, generates the ROE, which is a minimum excavation boundary from penetration to bucket curl, to include all waypoints of bucket tip to penetrate ground without crashing into the excavator cabin, as presented in Fig. 10. It also defines the number of waypoints in the ROE. The second AI, called the waypoint generation AI, generates the trajectory of bucket tip in the workspace specified by the ROE generation AI and calculates the angle of bucket in each step, as illustrated in Fig. 11. These AIs possess the same structure and consist of multiple layers, including the bidirectional LSTM that considers effects of forward and backward sequences of input data and two fully-connected layers, as depicted in Fig. 12. However,

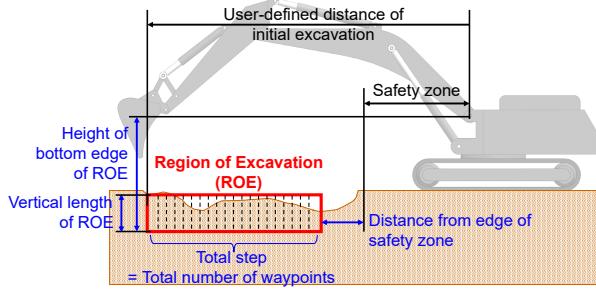


FIGURE 10. Concept of ROE generation AI

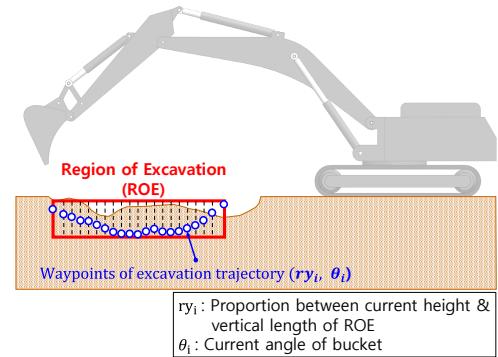


FIGURE 11. Concept of waypoint generation AI

inputs and outputs of these AIs are different.

Among the data of ROE generation AI in Fig. 12, polynomials of the terrain equation are the most important parameters for the ROE creation of bucket tip. We initially obtain point cloud data of the soil surface using the depth camera before the start of each excavation round. The data in this measurement include $1,270 \times 840$ points, and owing to the large data size, features of terrain cannot be feasibly obtained for generating the bucket-tip trajectory. To address this problem, we set the ROI of point cloud to reduce the size of point cloud, based on the width of bucket and the starting point of excavation. Subsequently, we project points in the defined ROI on the y-z graph and perform the polynomial regression of the y-z scatter diagram to secure 3 coefficients of the 2nd order equation, which represents a slope or a curvature of terrain, after the polynomial regression, as depicted in Fig. 13. After the polynomial regression, the ROE

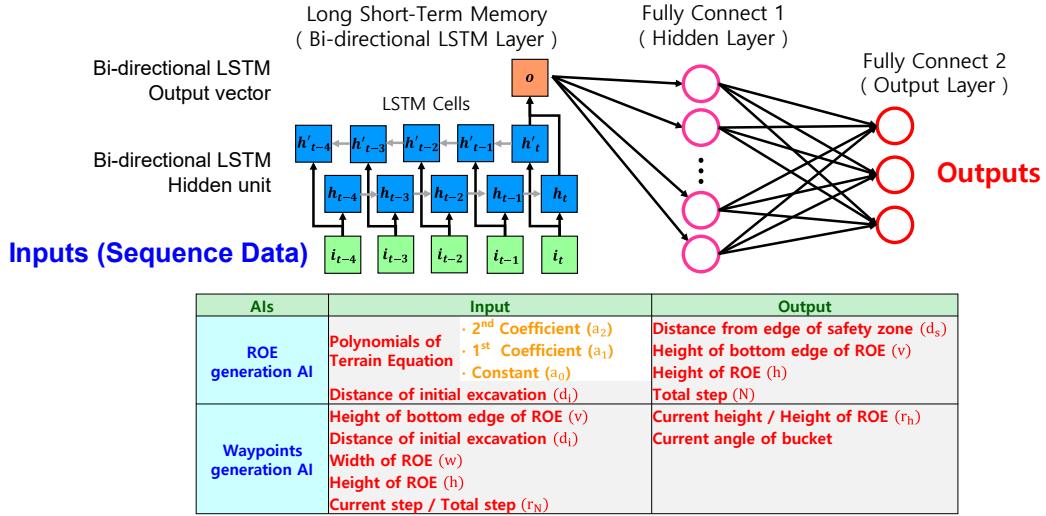


FIGURE 12. Structure of Al's in bucket-tip trajectory-planning AI system

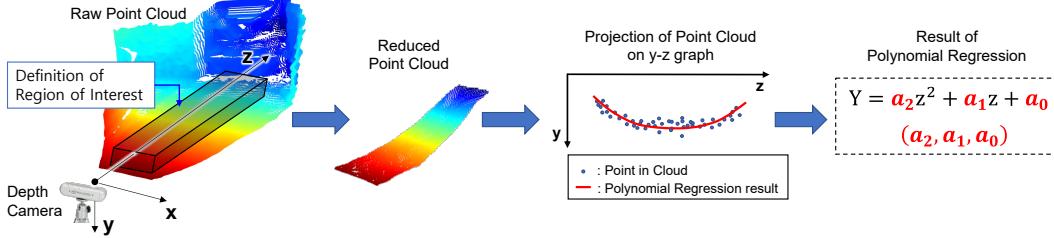


FIGURE 13. Transformation from point cloud to polynomials

generation AI uses these polynomials (a_2 , a_1 , a_0) and the distance from the boom joint to the initial excavation point d_i , which is defined by users, as inputs. The ROE generation AI outputs features of the ROE, including the distance between the edge of ROE and the edge of safety zone d_s , the height of ROE's bottom edge v , the vertical length of ROE h , and the number of excavation waypoints N in the ROE. Following the completion of ROE generation AI, the width of ROE w is determined using (11).

$$w = d_i - d_s - s \quad (11)$$

where s represents the range of the safety zone from the boom joint, which is determined with considering the crash safety.

Subsequently, the waypoint generation AI uses the geometric information of ROE and the distance of initial excavation point as inputs, including the height of ROE's bottom edge v , the distance of initial excavation point d_i , the ROE's vertical length h , and the ROE's width w , and the current step number divided by the total number of waypoints r_N , which is updated in every loop of the bucket-tip trajectory generation process. Using these values, the waypoint generation AI outputs the proportion r_h between the current height

of bucket tip and the vertical length of ROE, and the angle of bucket in each step. After the application of waypoint generation AI, the position of bucket tip is calculated as (12) and (13).

$$p_x = d_i - wr_N \quad (12)$$

$$p_y = v + hr_h \quad (13)$$

where p_x and p_y represent the position of bucket tip, as shown in Fig. 4. These values and the angle of bucket tip are transmitted to the controller to operate the excavator.

G. COLLISION AVOIDANCE ALGORITHM

The bucket-tip trajectory planning AI system primarily generates the trajectory of excavation after training based on the expert's trajectory. However, as the bucket sweeps through the ground, any unexpected variances, including underground rocks or buried pipes, must be considered. This variance under the soil surface cannot be identified using the depth camera or the trajectory planning AI, leading to unsatisfactory results, including machine stalling or rollover of the excavator during excavation due to the overload of the bucket.

To address these problems, we apply the collision avoid-

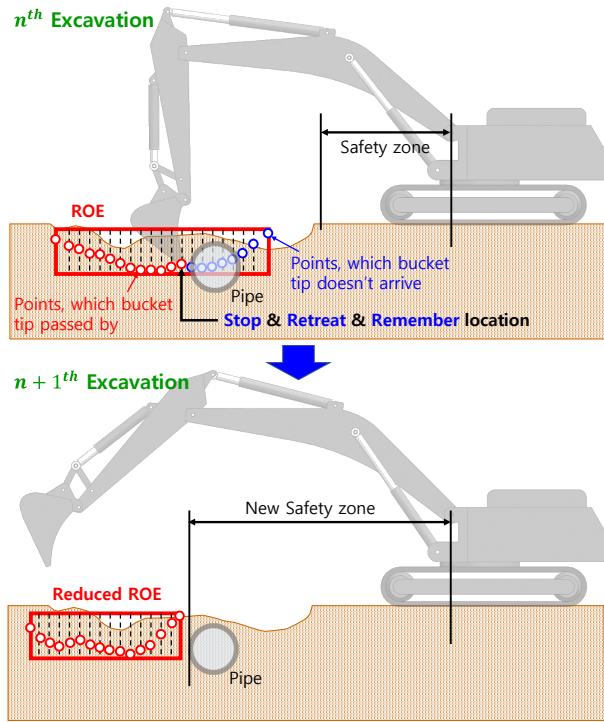


FIGURE 14. Process of collision avoidance algorithm

ance algorithm based on monitoring of the cylinder's pressure in each loop, as illustrated in Fig. 14. When the bucket tip approaches an underground obstacle, the pressure of the cylinder increases and exceeds the defined criteria of pressure. Thereafter, the collision avoidance algorithm suspends excavation and retracts the bucket in the opposite direction of path obtained via the bucket-tip trajectory planning AI system. This strategy is implemented because it minimizes chances of the bucket getting trapped, as the bucket tip moves back along the path through which it advanced. After a few reverse moves, the collision avoidance algorithm pulls up the bucket and finishes the excavation. Before the bucket is retracted, the collision avoidance algorithm saves the location information of place wherein one of cylinder pressures exceeds the pressure criteria. This saved value is applied to determine the width of ROE in the subsequent excavation, by updating the range of safety zone to protect the bucket from crashing. If the ROE width after the ROE generation AI is considerably large to be crashed into underground obstacles, then the ROE width will be calculated again, with consideration of the new safety zone, which is set up in front of underground obstacles. If this is not the case, then the ROE configuration of ROE generation AI will be transmitted into the waypoint generation AI without modification.

H. ACQUISITION OF TRAINING DATA

We secured the data for AIs from experts' excavation, following the sequence illustrated in Fig. 15. During expert's excavation, 4 successive excavations are performed in each

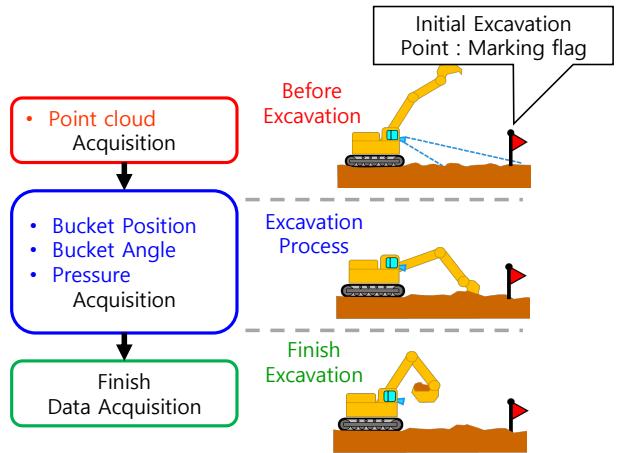


FIGURE 15. Process of expert data acquisition

cycle to render AIs robust against changes in the terrain's slope. In this study, we define the distance from the initial excavation point to the boom joint as 0.95 m wherein the marking flag is located for experts to identify where to initiate the penetration. Experts' excavation is performed under the same condition and same height as that of our AI system to compare their performances. When the bucket approaches the marking flag, the data, including the point cloud, the bucket position and the angle, and the pressure of cylinder, are saved during experts' excavations from the penetration to the bucket curl. Among the data of experts, we select 180 trials of experts' excavation data as qualified training data, wherein the soil weight ranges from 2.8 kg to 3.15 kg and the lead time ranges from 0.5 sec to 1.1 sec. For supplementary data, we conduct data augmentation by inserting random noise into the dataset. Moreover, we limit the magnitude of random noise such that results remain unaffected.

For the training data of ROE generation AI, point clouds are pre-processed with the size reduction, the projection on 2D graph, and the polynomial regression to extract polynomials, following the process depicted in Fig. 13. The distance of initial excavation d_i is determined by the first position of bucket tip in each trial. The ROE configuration is determined by calculation of the least region, which includes all waypoints of each excavation.

The training data of waypoint generation AI includes the ROE configuration, which is calculated during the data generation process of ROE generation AI, with the distance of initial excavation. Additionally, the proportion between the bucket displacement and the ROE width is calculated in accordance with (14).

$$r_N = \frac{d_i - p_x}{w} \quad (14)$$

where r_N denotes the proportion value and corresponds with the current step divided with the total step in Fig. 12, w represents the ROE width, and p_x represents the distance of bucket tip in each step of excavation. Among outputs of

TABLE 3. Hyperparameters of bucket trajectory-planning AIs

Hyperparameters	Value and Info.
Units of LSTM layer	2(bidirectional) \times 128
Units of 1 st FC layer	64
Units of 2 nd FC layer	4 (waypoint generation AI: 2)
Activation of FC	Linear
Loss	MSE
Optimizer	ADAM
Metric	MSE
Epoch	30
Batch size	10 (Waypoint generation AI: 100)

training data set, the proportion value between the vertical displacement of bucket and the height of ROE is calculated as r_h in (15).

$$r_h = \frac{p_y - v}{h} \quad (15)$$

where r_h corresponds with the current height of bucket divided by the ROE's vertical length in Fig. 12, v and h represent the height of ROE's bottom edge and the vertical length of the ROE, and p_y denotes the current depth of bucket tip. Additionally, the scaling factor is applied to the angle of bucket tip to adjust the magnitude to a comparable range with other values.

I. HYPERPARAMETERS

The hyperparameters set for 2 AIs includes the cost function, the size of mini-batch, epochs, and the node size of each layer. These hyperparameters are listed in Table 3.

J. ALGORITHM STRUCTURE

The algorithm is composed of 3 main algorithms, as shown in Fig. 16: the ROE generation AI, the waypoint generation AI, and the collision avoidance algorithm.

Before the excavation, the depth camera obtains the point cloud within the defined ROI, and subsequently, the polynomial regression is applied to secure polynomial values as input to the ROE generation AI. The ROE generation AI outputs the ROE configuration with the number of waypoints, and the waypoint generation AI generates the depth and the angle of bucket tip as location data of waypoint. Including the distance calculated in (12), the waypoint's location is saved in a list variable. When the waypoint of final step is saved, the excavation starts.

During the excavation, pressure values of boom, arm, and bucket cylinder are continuously monitored by the collision avoidance algorithm to determine if they exceed the safety criteria. Based on the result of collision avoidance algorithm, the excavator continues the current excavation or cancels it and adjusts the ROE width for the next excavation, while identifying the location of underground obstacles.

Algorithm 1 Bucket tip-trajectory planning AI system

Input: Pressure, Point cloud

Output: Future position, Future angle

Initialisation :

- 1: Obtain point cloud in defined ROI
 - 2: Calculate polynomials of terrain equation
 - 3: Generate ROE box by AI, considering safety zone
 - 4: Enlist bucket-tip trajectories by AI
- Loop process*
- 5: **if** State1 **then**
 - 6: **if** $i ==$ Size of trajectory list **then**
 - 7: State1 = False
 - 8: **end if**
 - 9: **if** (Current pressure > Pressure threshold) **then**
 - 10: State1, State2 = False, True
 - 11: Save current location as new safety zone
 - 12: **end if**
 - 13: Transmit i^{th} vector set of list to controller
 - 14: $i += 1$
 - 15: **end if**
 - 16: **if** State2 **then**
 - 17: Transmit i^{th} vector set of list to controller
 - 18: $i -= 1$
 - 19: **if** ($i ==$ Threshold of retreat) **then**
 - 20: State2 = False
 - 21: **end if**
 - 22: **end if**
 - 23: **return** Futureposition, FutureAngle

III. EXPERIMENTAL RESULTS

The primary purpose of bucket-tip trajectory planning AI system is to achieve robust excavation of various terrains. To validate our AI system, we first perform 4 successive excavations. Subsequently, we introduce comparison tests with expert's trials to validate that the AI-led excavation is comparable to that of the expert in terms of efficiency. During the excavation, we constrain the distance of excavation starting point, which is approximately 0.95 m from the boom joint for safety, and focus on the excavation volume and the lead time of excavation as performance factors.

A. PERFORMANCE VALIDATION OF TRAINED AI

Performances of two AIs are validated during training. The training loss and accuracy are the key performance parameters in the validation. The loss and metric to optimize and evaluate the AIs are calculated with the mean squared error (MSE) expressed in (16).

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2, \quad (16)$$

where \hat{Y}_i is the estimated value of ROE configuration in the ROE generation AI and the estimated value of the rate of height and the angle of bucket tip in the waypoint generation AI. Y_i denotes the truth value, which is the result of experts' work, regarding the input data. The MSE of each AI is

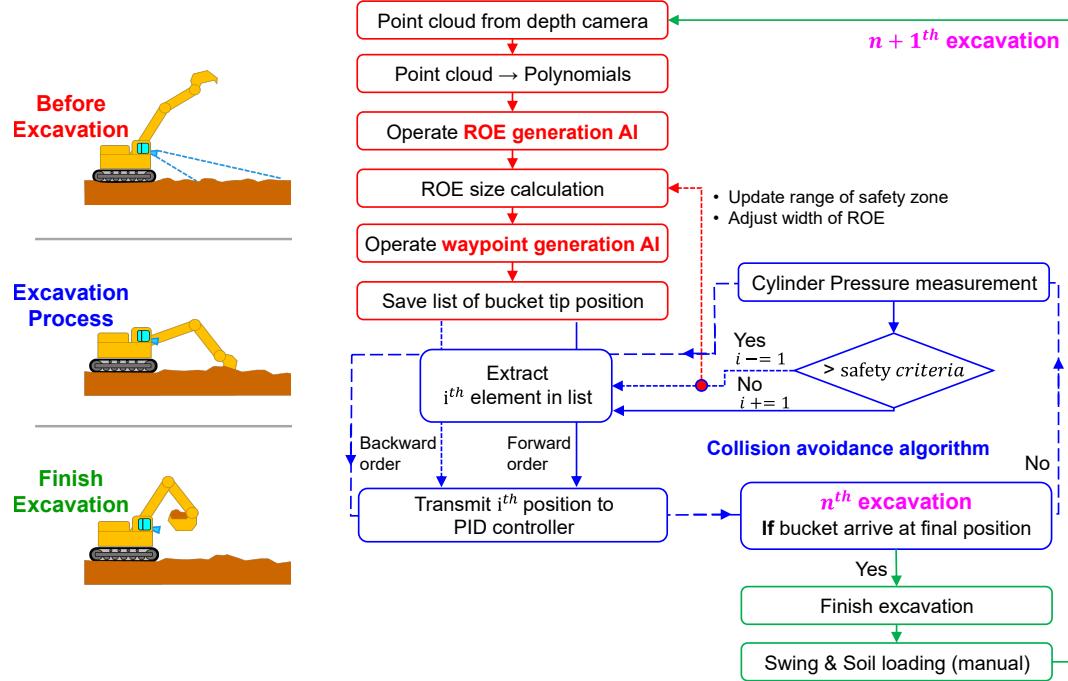


FIGURE 16. Process of AI system

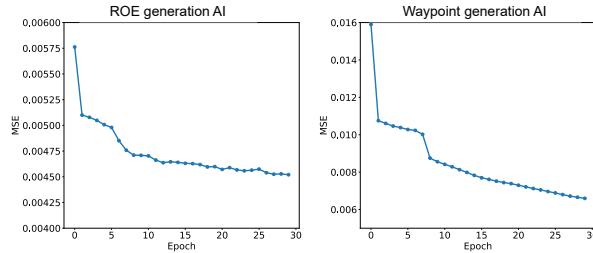


FIGURE 17. Training loss of two AIs

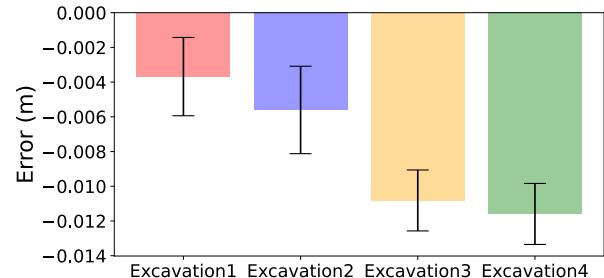


FIGURE 18. Tracking performance of each excavation : tracking error

illustrated in Fig. 17, and the performance saturates well around 30 epochs for training both AI algorithms.

B. TRAJECTORY TRACKING PERFORMANCE DURING SUCCESSIVE EXCAVATIONS

This experiment is performed to verify the quality of bucket-tip trajectory generated by the AI system, comparing the bucket course and the trajectory obtained via AIs in Figs. 19. Essentially, we check if the bucket tip trajectory developed by the AI system can be followed by the PID controller, although certain unfavorable factors are present, including delay from the opening of an electronic valve to the transmission of hydraulic power during the continuous update of new target point of the bucket tip from the AI system with 100 ms period. Moreover, a more effort is required to optimize gains of PID controllers during the excavation. Considering this aspect and the tracking performance of an excavator in a prior study [15], whose average position tracking error is

0.078 m, the limit of tracking error in this study is under 0.015 m. If the tracking error between the measured track and the AI-based path, which are shown in Fig. 19, satisfies this criteria, the bucket-tip trajectory planning AI system is proven to be able to develop a traceable track for the bucket tip. The result of tracking experiment is illustrated in Fig. 18, and Table 4 with the 99% confidence interval range, showing that tracking errors of 1st, 2nd, 3rd, and 4th excavation are under 0.015 m on average. That is, our algorithm can develop a traceable path.

C. EFFICIENCY OF THE EXPERTS AND THE AI

The excavation performance of bucket-tip trajectory planning AI system is compared with that of experts in terms of efficiency. We use the weight of excavated soil and the lead time from the starting location of excavation to the location at which the bucket is lifted, as indicators of performance.

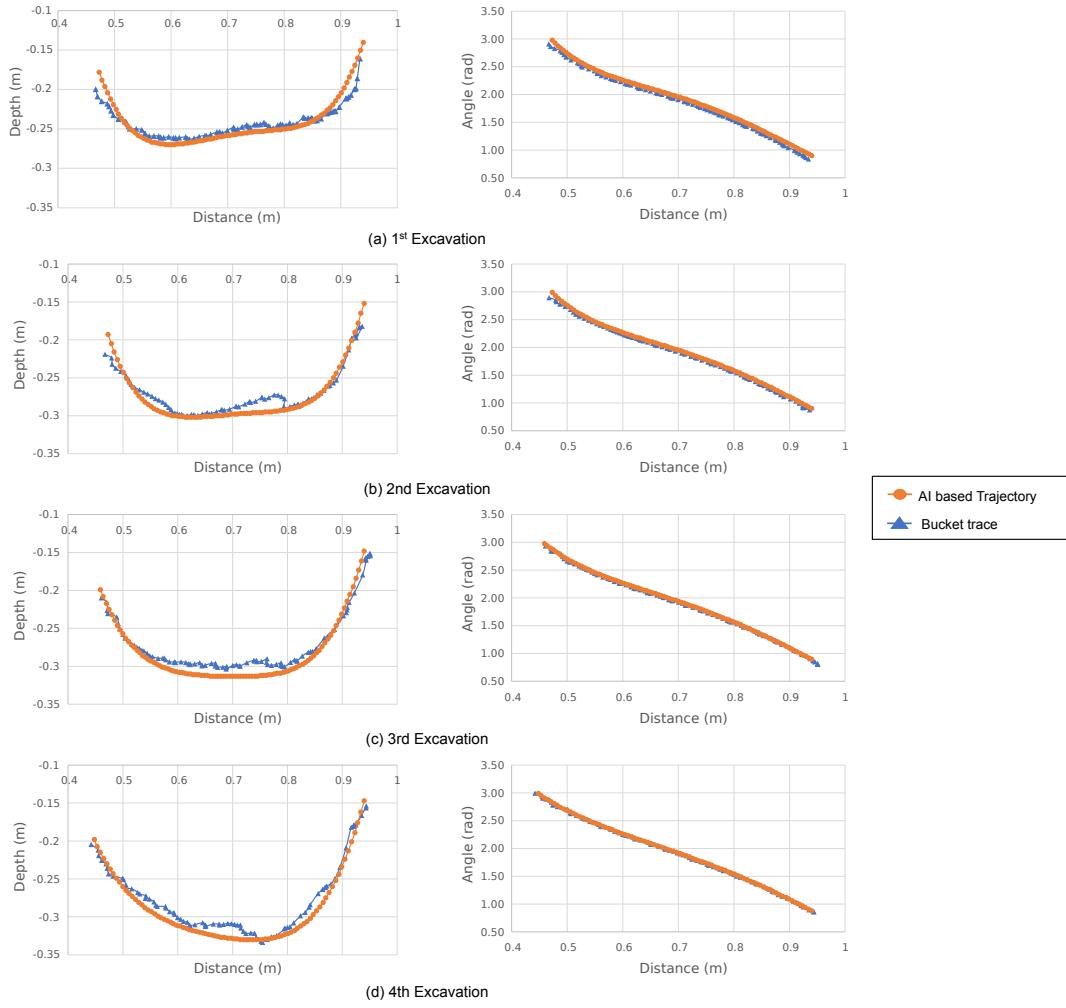


FIGURE 19. Comparison between AI based trajectory and trace of bucket

TABLE 4. Confidence interval range of tracking error

Excavation No.	99% Confidence interval range (m)
Excavation1	-0.006 ~ -0.0014
Excavation2	-0.0082 ~ -0.003
Excavation3	-0.0126 ~ -0.009
Excavation4	-0.0134 ~ -0.0098

Over 80 excavation trials are performed by experts and the AI system, and the mean and standard deviation of each result are calculated. In Fig. 20, and Tables. 5 and 6, the AI system and experts excavate more than 2 kg of soil and require 8 ~ 9 sec for excavation. However, the bucket-tip trajectory planning AI system exhibits steadier work efficiency compared to experts, as validated using standard deviations of soil weight and lead time, as presented in 99% confidence interval ranges in Figs. 21 and 22.

TABLE 5. Average and standard deviation of weight of excavated soil

unit : g		AI	Expert1	Expert2	Expert3	Expert4
1 st	avg	2950	2737	2786	2813	2509
	std	124	215	128	205	197
2 nd	avg	2937	2605	2784	2814	2638
	std	168	194	144	242	182
3 rd	avg	2998	2654	2835	2630	2560
	std	109	348	143	299	268
4 th	avg	2677	2569	2816	2663	2491
	std	172	349	160	246	266

D. APPLICATION OF COLLISION AVOIDANCE ALGORITHM

We also verify the application of collision avoidance algorithm based on pressure monitoring. We install a pipe representing an underground obstacle, as shown in Fig. 23, and performed successive excavations. When pressure values of three cylinders rise and exceed the safety criteria, the

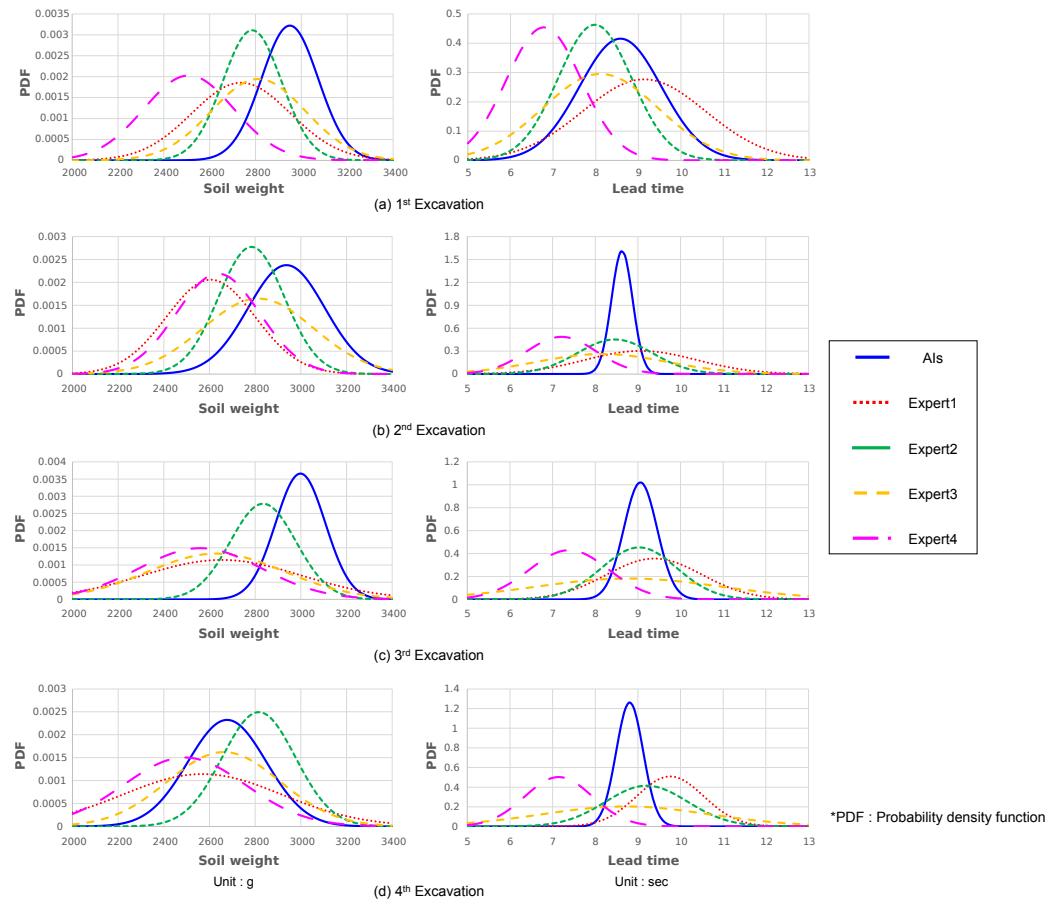


FIGURE 20. Normal distribution of excavation performance for excavation by experts and AI system

TABLE 6. Average and standard deviation of lead time for excavation

unit : sec	AI	Expert1	Expert2	Expert3	Expert4	
1 st	avg std	8.6 0.96	9.1 1.4	8 0.86	8.1 1.4	6.8 0.88
2 nd	avg std	8.6 0.25	9.1 1.3	8.5 0.88	8.2 1.5	7.2 0.82
3 rd	avg std	9.1 0.39	9.4 1.1	9 0.88	8.8 2.2	7.4 0.93
4 th	avg std	8.8 0.32	9.8 0.78	9.2 0.96	8.7 2	7.1 0.79

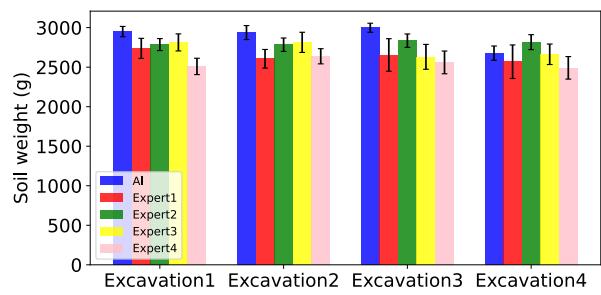


FIGURE 21. 99% Confidence interval range : Soil weight

collision avoidance algorithm halts excavation and retreats the bucket, similar to the 2nd excavation illustrated in Fig. 24. The ROE width decreases, after updating the range of safety zone to cover the location of detected underground pipe. After the reduction of ROE width, the trajectory of 3rd excavation (Fig. 24) is also shortened, without crashing into the underground pipe.

IV. CONCLUSION AND FUTURE WORKS

The autonomous excavation operated by the bucket-tip trajectory planning AI system and the collision avoidance algorithm based on pressure monitoring shows results comparable with those of experts in terms of the excavation volume and lead time. Although the geography of the targeted terrain changes owing to successive excavation, the LSTM-based AI system robustly generates a traceable trajectory without considering the complicated soil dynamics. The collision avoidance algorithm adjusts the ROE's width

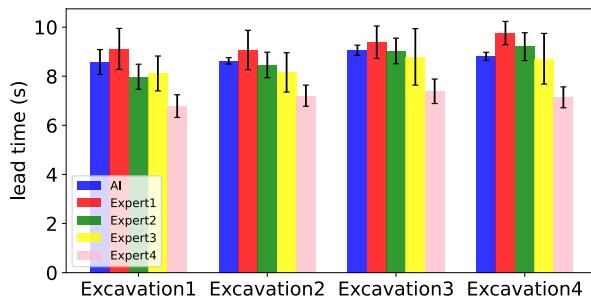


FIGURE 22. 99% Confidence interval range : Lead time

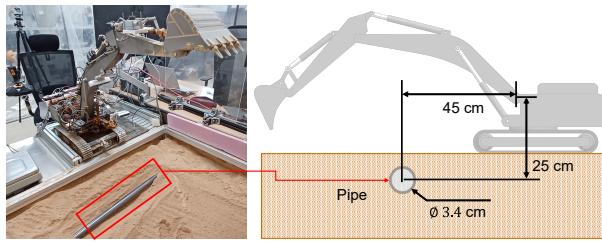


FIGURE 23. Underground obstacle(Pipe), installed in front of excavator

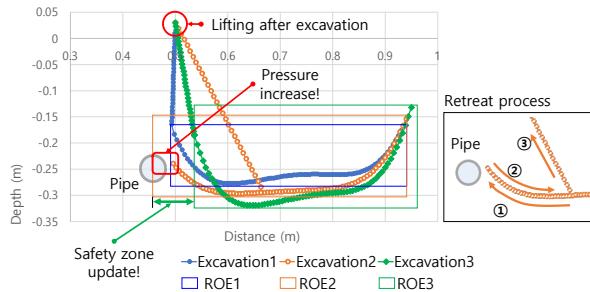


FIGURE 24. Excavation under activation of collision avoidance algorithm

for the waypoint generation AI as a result of monitoring the pressure, thereby preventing the bucket from crashing into underground obstacles.

However, certain challenges must be overcome to apply the bucket-tip trajectory-planning AI system in real-world excavation. When soil was frozen or wet, penetration into soil as initial part of excavation was not smooth. Consequently, pitch occurred in cabin during penetration process. In future works, for safety, we will resolve excavation of soil with different hardness, thereby improving our proposed AI system. Furthermore, we will extend the application of our AI system from one section to field including several sections. For this purpose, we will investigate the planning of excavation tasks based on the topographical data of the specified field, referencing [19], and integration with the proposed AI system in this study.

REFERENCES

- [1] S. Vahed, X. Song, J. Dai, H. Lam, L. Seneviratne, and K. Althoefer, "Soil estimation based on dissipation energy during autonomous excavation," IFAC Proceedings Volumes, vol. 41, no. 2, pp. 13 821–13 826, 2008.
- [2] A. Koivo, M. Thoma, E. Kocaoglan, and J. Andrade-Cetto, "Modeling and control of excavator dynamics during digging operation," Journal of aerospace engineering, vol. 9, no. 1, pp. 10–18, 1996.
- [3] B. P. Patel and J. M. Prajapati, "Soil-tool interaction as a review for digging operation of mini hydraulic excavator," International Journal of Engineering Science and Technology, vol. 3, no. 2, pp. 894–901, 2011.
- [4] O. Luengo, S. Singh, and H. Cannon, "Modeling and identification of soil-tool interaction in automated excavation," in Proceedings. 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications (Cat. No. 98CH36190), vol. 3. IEEE, 1998, pp. 1900–1906.
- [5] H. Cannon and S. Singh, "Models for automated earthmoving," in Experimental Robotics VI. Springer, 2000, pp. 163–172.
- [6] Y. Zhao, J. Wang, Y. Zhang, and C. Luo, "A novel method of soil parameter identification and force prediction for automatic excavation," IEEE Access, vol. 8, pp. 11 197–11 207, 2020.
- [7] P. Egli and M. Hutter, "A general approach for the automation of hydraulic excavator arms using reinforcement learning," IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 5679–5686, 2022.
- [8] J. Park, D. Cho, S. Kim, Y. B. Kim, P. Y. Kim, and H. J. Kim, "Utilizing online learning based on echo-state networks for the control of a hydraulic excavator," Mechatronics, vol. 24, no. 8, pp. 986–1000, 2014.
- [9] J. Park, B. Lee, S. Kang, P. Y. Kim, and H. J. Kim, "Online learning control of hydraulic excavators based on echo-state networks," IEEE Transactions on Automation Science and Engineering, vol. 14, no. 1, pp. 249–259, 2016.
- [10] S. Lee, D. Hong, H. Park, and J. Bae, "Optimal path generation for excavator with neural networks based soil models," in 2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems. IEEE, 2008, pp. 632–637.
- [11] S. Dadhich, F. Sandin, U. Bodin, U. Andersson, and T. Martinsson, "Field test of neural-network based automatic bucket-filling algorithm for wheel-loaders," Automation in Construction, vol. 97, pp. 1–12, 2019.
- [12] R. J. Sandzimier and H. H. Asada, "A data-driven approach to prediction and optimal bucket-filling control for autonomous excavators," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2682–2689, 2020.
- [13] B. Son, C. Kim, C. Kim, and D. Lee, "Expert-emulating excavation trajectory planning for autonomous robotic industrial excavator," in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020, pp. 2656–2662.
- [14] P. Egli, D. Gaschen, S. Kerscher, D. Jud, and M. Hutter, "Soil-adaptive excavation using reinforcement learning," IEEE Robotics and Automation Letters, vol. 7, no. 4, pp. 9778–9785, 2022.
- [15] P. Egli and M. Hutter, "Towards rl-based hydraulic excavator automation," in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020, pp. 2692–2697.
- [16] L. Tai, J. Zhang, M. Liu, J. Boedecker, and W. Burgard, "A survey of deep network solutions for learning control in robotics: From reinforcement to imitation," arXiv preprint arXiv:1612.07139, 2016.
- [17] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," The International journal of robotics research, vol. 37, no. 4–5, pp. 421–436, 2018.
- [18] R. Grosse, "Lecture 15: Exploding and vanishing gradients," University of Toronto Computer Science, 2017.
- [19] T. Osa and M. Aizawa, "Deep reinforcement learning with adversarial training for automated excavation using depth images," IEEE Access, vol. 10, pp. 4523–4535, 2022.



JAEEMYUNG HUH received the B.S. degree from the school of mechanical engineering, Korea University, Seoul, South Korea, in 2012, where he is currently pursuing the Ph.D. degree in mechanical engineering.

His research interests include algorithms design of automation in excavation process, machine learning, and mechanism design of automated guided vehicle system and gripper system.



CHULWHAN IM received the B.S. degree from the School of Mechanical Engineering, Korea university, Seoul, Korea, in 2022, where he is currently pursuing the M.S. degree in mechanical engineering.

His research interests include field robotics, vision, and autonomous excavator.



JANGHO BAE is a professor in the department of mechanical and automotive engineering at Kyung-sung University, Busan, Korea. He earned his B.S., M.S., and Ph. D. in the Department of Mechanical Engineering at Korea University, Seoul, Korea.

His research interests include design and control of hydraulic robotic system, physical human-robot interaction, human-robot collaboration system, application of artificial intelligence in field robotics and autonomous car system.



YEONHO KO received the B.S. degree in mechanical design engineering from Chungnam National University, Daejeon, Korea, in 2015. He is currently pursuing the Ph.D. degree in mechanical engineering at Korea University, Seoul, Korea.

He is also a Researcher. His research interests include design and control of wearable robotic systems, physical human-robot interactions and collaboration systems, and construction field robotics.



DONGHYEON LEE received the B.S. degree from the School of Mechanical Engineering, Korea University, Seoul, South Korea, in 2021, where he is currently pursuing the M.S. degree in mechanical engineering.

His research interests include human-robot interaction, field robotics, and machine learning.



TAE KOO KANG received his B. S. in Applied Electrical Engineering, M. S. in visual image processing, and Ph. D. in Electrical Engineering from Korea University, Seoul, Republic of Korea, in 2001, 2004, and 2012 respectively. He was a research professor at Korea University, Seoul, Republic of Korea from 2012 to 2014 and an assistant professor in Information and Telecommunication Engineering, Cheonan, Republic of Korea in 2015 and 2016. He is now an Assistant Professor in Department of Human Intelligence and Robot Engineering, Sangmyung University, Cheonan, Republic of Korea.

His research interests include computer vision, robotics, artificial intelligence, and machine learning.



JAEWON KWAK received the B.S. degree from the School of Mechanical Engineering, Korea University, Seoul, South Korea, in 2021, where he is currently pursuing the M.S. degree in mechanical engineering.

His research interests include control of hydraulic excavator and computer vision.



DAEHIE HONG received the B.S. and M.S. degree from the school of mechanical engineering, Korea University, Seoul, South Korea, in 1985 and 1987, respectively, and the Ph.D. degree from the Department of Mechanical and Aerospace Engineering, University of California at Davis, Davis, CA, USA, in 1994.

He is currently a professor with the school of Mechanical Engineering, Korea University. His research interests include manufacturing automation, precision machine design and control, field robotics, and autonomous car systems.



CHANYOUNG MOON received the B.S. degree in mechanical engineering from Korea University, Seoul, Korea in 2019. He is currently pursuing the M.S. degree in mechanical engineering at Korea University, Seoul, Korea.

His research interests include automation in excavation process, field robotics, and medical robot.