Neural Network Pile Loading Controller Trained by Demonstration

Eric Halbach[1], Joni Kämäräinen[2] and Reza Ghabcheloo[1]

***Abstract*— This paper presents the development and testing of end-to-end Neural Network (NN) controllers for automated pile loading with a robotic wheel loader. NNs were trained using the Learning from Demonstration approach, i.e. by first recording sensor and control signals during manually-driven pile loading actions. Training made use of three input signals: boom angle, bucket angle and hydrostatic driving pressure; and three output signals: boom control, bucket control and the gas command. Most testing was conducted using NNs with 5 neurons in a single hidden layer, which were able to fill the bucket reasonably well. Qualitative comparisons were made to ascertain how the amount of training data and number of hidden neurons affects bucket filling performance, for NNs trained using both the Levenberg-Marquardt and Bayesian Regularization backpropagation algorithms. Different NNs trained with the same data were also compared. An additional pile transfer experiment compared the performance of an NN controller with a heuristic automated controller and manual human control. By estimating the total volume of material transferred using 3D laser scans, human control was found to have the highest performance, though the NN outperformed the heuristic controller. This indicated that end-to-end NN control trained by demonstration could offer improvement over current heuristic methods for automated pile loading.**

I. INTRODUCTION

The use of automated robotic machinery at mining and construction sites can help to increase safety and also productivity. Some parts of the work cycle are already being automated, such as hauling and dumping in underground mines, and hauling with dump trucks at open pit mines. The material excavation or loading phase of the work cycle is usually still controlled by a human, either directly or via teleoperation. This action is difficult to automate due to the unpredictable nature of ground material, which can contain rocks of unknown size, and which can behave differently depending on factors such as compaction and moisture.

This paper addresses the problem of bucket loading automation for wheel loaders, with the test platform being a computer-controlled compact Avant 635 (see Fig. 1). Automated control for bucket loading has been developed in the related work, with the general approach being to make use of heuristics to either follow a desired trajectory through the pile (possibly combined with detecting threshold forces)[1][2][3][4], or to apply specified forces to the compliant material [5]. Although automated loading appears close to being introduced in some industrial scenarios, it is still not generally applied.

*This work was supported by the Academy of Finland

[1]Automation and Mechanical Engineering, Faculty of Engineering and Natural Sciences, Tampere University, P.O. Box 1001, 33014, Finland eric.halbach@gmail.com

[2]Computing Sciences, Faculty of Information Technology and Communication Sciences, Tampere University, P.O. Box 1001, 33014, Finland

Fig. 1: Wheel loader and dirt piles used in our experiments.

One challenge in developing automated controllers using heuristics is that human experience and common sense needs to be coded into the form of an algorithm. This may not be trivial given the intricate coordination of the gas command and bucket actuation required to successfully load a bucket, which can change with different material properties. Another challenge is that humans and computers may find different types of sensing information to be useful. A human operating an excavation machine may use the sense of hearing, for example, to estimate the current load on the engine, while a computer could directly sense engine parameters.

Using recorded sensor data from human control, on the other hand, to develop a heuristic controller can be tedious if it requires analyzing large amounts of data from numerous sensors, and the tuning of parameters by trial and error. One example would be deciding when to start lifting the bucket and how quickly based on reaction forces encountered. Meanwhile, reaction forces and bucket control velocities may vary significantly from one recorded data set to the next.

This type of problem may be suited to machine learning techniques, which have the advantage of being able to process large amounts of data and sensor information to produce the desired output. Some machine learning approaches have recently been proposed for robotic excavation [6][7], but a full implementation of machine learning control of pile loading has so far not been found in the literature.

This paper presents a new method for automatic bucket loading with a robotic wheel loader using end-to-end Neural Network (NN) control. End-to-end means that control signals (e.g. joint velocities) are generated directly based on sensor information. The NN training procedure used here follows the Learning from Demonstration (LfD) approach [8], i.e. using training data collected during manually controlled actions.

Some advantages to this approach are that the experience of a human operator is still transferred to the computer, while implementation is made easier since sensor data need not be analyzed by humans to develop algorithms, but merely collected and annotated. No heuristics are used directly, although the sensor signals fed to the NN can be selected based on what is thought to be useful.

A. Contributions

The contributions made by the work presented in this paper are summarized as follows:

- A proposed shallow fully connected NN controller for a robotic wheel loader to automatically load material from a pile. The inputs to the NN controller are the boom angle, bucket angle and hydraulic driving pressure, and the outputs are control velocities for the boom and bucket joints, and the gas command. To the authors' best knowledge this is the first NN controller which is demonstrated real-time on a wheel loader.
- A procedure to train the proposed NN controller with data produced by a human operator (LfD approach).
- An ablation study of the components of the network and the training procedure, including how the number of training examples needed, training algorithm used (Levenberg-Marquardt, Bayesian Regularization and Gradient Descent) and number of hidden neurons affect the bucket filling performance.
- A comparison between the performance of an NN controller, a heuristic-based automated controller and manual human control for a pile transfer job.

The next section will further discuss related work in this area, followed by the development of the NN controller in Section III. Experimental results are then presented in Section IV, followed by the conclusion and areas for future work in Section V.

II. RELATED WORK

The application of machine learning to wheel loader automation, mentioned in Section I, includes the work of Dadhich et al., who provide an analysis of recorded data from manually controlled loading of gravel and rock piles with a full-sized wheel loader [6]. The analysis yields a regression model for understanding the relationship between the loader's lift and tilt cylinder lengths and the hydraulic forces applied by the cylinders. The work presented in this paper takes a similar approach by starting with collected data from manual control, and then extends it by developing and demonstrating a full NN controller for a wheel loader.

Fernando et al. propose using learning of control setpoints to control bucket fill for an admittance controller, which is able to autonomously load from muck piles that can contain large rocks [7]. This heuristic-based controller does not follow a desired trajectory but rather applies specified forces to the pile which helps to deal with rocks. The NN controller presented here differs since it replaces the entire control system for scooping actions.

Much previous work has been done in the application of NNs for robot control, however in other application areas such as robotic manipulator control and motion control of mobile robots [9] [10] [11], not to mention the current explosion of Deep Learning research in the automobile industry. An introduction to the application of NNs for control systems in general is provided by Hagan et al. [12]. Their description of NN predictive control is the technique followed in this paper. The increasing role of deep learning and end-to-end learning in robotics is discussed by Sünderhauf et al. [13].

In their recent works, Viereck et al. and Antonova et al. adopt NNs for robot control [14][15]. Viereck et al. propose a Convolutional Neural Network (CNN) architecture that maps a depth image and a small manipulator movement delta (3D translation and rotation) to a distance function to the nearest viable grasp in object grasping. The pre-programmed controller executes actions based on the CNN-based grasp approaching cost function until it converges to a position where a pre-programmed and fixed grasp action is launched. Antonova et al. use an NN to train a cost function for their controller parameters. The network is trained with short simulation data and provides a cost value based on whether the walking robot falls or not. An optimization procedure is executed and the NN-cost optimized controller parameters are used in simulated and real experiments. A similar work of a deep CNN-based cost function for autonomous driving was proposed by Drews et al. [16].

As a recent example of end-to-end control, James et al. present NN control for a picking task which was trained in simulation and applied to a real manipulator [17]. This paper also presents an end-to-end controller, though the application is for an outdoor heavy machine in a field environment, and training is done using sensor data (no vision) collected during manual control of the machine.

III. NEURAL NETWORK CONTROLLER

This section describes the experimental setup and procedure used to train and test the NN scooping controller.

A. Experimental Setup

The computer-controlled compact Avant 635 wheel loader used here has a bucket 1.04 m wide with a volume of approximately 0.14 m^3. The bucket can be positioned via three joints: a rotary boom joint, a telescopic joint inside the boom, and a rotary bucket joint.

Various sensors are installed onboard including a Global Navigation Satellite System (GNSS) positioning receiver, wheel odometers, Inertial Measurement Units (IMUs), a front-mounted laser scanner, and sensors for measuring joint angles and hydraulic pressures. More details about the system architecture are available in the authors' previous work [18].

Experiments are conducted at an outdoor testing site on campus which includes a paved area adjacent to a rocky soil area containing piles (see Fig. 1). The soil contains stones of various sizes, mostly under 10 cm long though with a few larger rocks up to 25 cm across. Experiments can be monitored from an on-site control room.

B. Training by Demonstration

The first step in training the NNs was to collect data from manually driven pile loading actions (see Fig. 2). These were driven by the first author, and though they were not the actions of a professional operator, some experience had been accumulated over several years of working with Avant loaders. The general strategy followed during the loading actions can be summarized as follows:

- position Avant facing slope at a perpendicular angle, starting various distances from the base (∼1-5 m)
- position bucket low and level with ground (with constant telescopic boom extension ∼0.1 m)
- drive towards pile
- as contact is made and resistance stops forward motion, begin raising boom while maintaining gas throttle and trying to avoid wheel slippage
- when bucket mostly full and raised significant amount (∼0.5 m), curl bucket upwards to collect final material while releasing gas throttle.

This strategy was refined during the first few actions, which were less smooth and had more wheel slippage than the subsequent ones. In all, 23 data sets were collected. During one scooping action, the loading on the bucket was too great for the hydraulics and the engine died, yet this data was still included since the action was nearly complete. All actions resulted in bucket loads with some or significant overfill with respect to the geometric volume capacity.

Fig. 2: Manually driven scooping action for data collection.

Several sensor and control signals were recorded during the manually controlled actions. Those chosen for the NN training were kept to the minimum which were considered sufficient for controlling a scooping action. These include the joint angles and control velocities for the rotary boom and bucket joints, the gas command, and finally, the hydrostatic driving pressure, i.e. the hydraulic pressure driving the wheels. This pressure value exhibits a characteristic spike as the Avant makes contact with and pushes into a pile, thus it was thought to be a way that the machine could *feel* its way through a digging action.

Fig. 3 shows a sample of signals collected from one scooping action, with some of the main features labelled. The control signals for actuating the boom and bucket range from -1 to 1, while the gas command ranges from 0 to 1. The hydrostatic driving pressure, in bars, is here divided by 200 so that all signals can be plotted conveniently together.

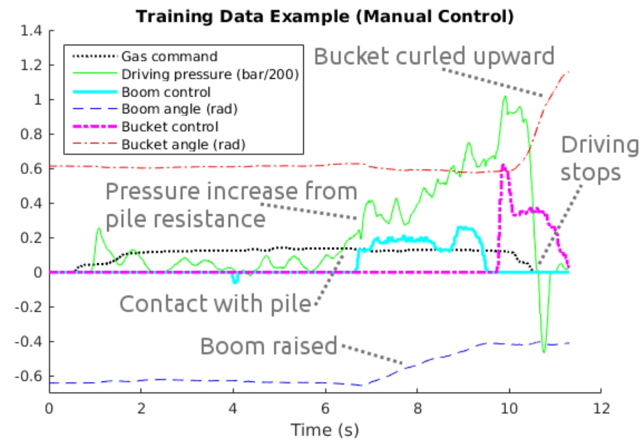


Fig. 3: Sample of recorded signals from one manually driven scooping action, with description of signal features.

C. Network Structure

Since the controller was trained with manually generated examples, it was necessary to adopt a shallow architecture to keep the number of network parameters small. The network structure is illustrated in Fig. 4 with its inputs and outputs. The standard NN used is very simple, with only a single hidden layer with 5 neurons and a fully connected output layer. During the experiments the network was trained using two different gradient methods, Levenberg-Marquardt (LM) and Bayesian Regularization (BR) [19][20]. LM is one of the most efficient training algorithms, while BR is less aggressive but provides robustness against over-fitting in the case of a small number of training samples. The NN was created in Matlab using the function *fitnet*, which uses the *tansig* transfer function in the hidden layer [21].

D. Initial Testing

The initial NN tested had 5 hidden neurons and was trained with 20 data sets, using the LM algorithm. The NN was integrated into the Avant by adding it to the Avant's Simulink Real-Time model and connecting the input and output signals. The general testing procedure, also followed during the experiments described in the next section, was to first prepare a scooping location at the test site by manually performing one loading and unloading action, in order to loosen the material and begin with easier conditions. The

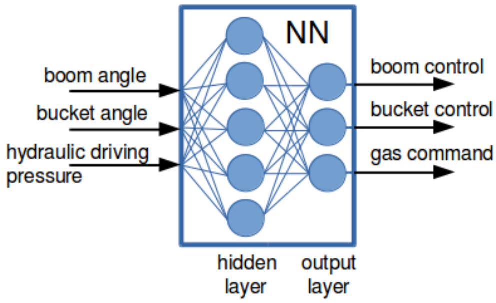


Fig. 4: NN structure with 5 neurons in hidden layer.

Avant was then manually driven a few metres in front of the scooping location, and the bucket positioned low and level with the ground, with a constant telescopic boom extension of ~0.1 m. The NN was then given control by switching the Avant to autonomous mode.

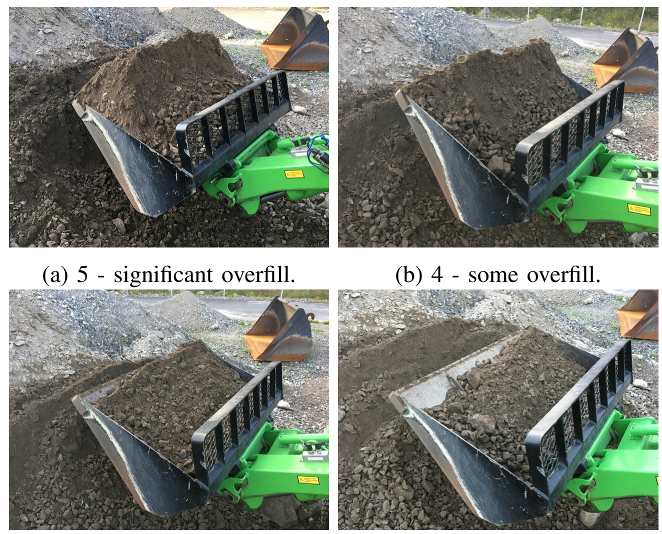
The Avant drove somewhat slower than a human typically would, but successfully loaded material. After a loading test, the Avant was switched back to manual mode, with the bucket being unloaded at the same location and the Avant then reversed to a starting location, with the bucket again positioned low and level with the ground. 13 tests were initially conducted, with the Avant successfully loading material and filling the bucket relatively well each time (see example in Fig. 5). Additional tests were conducted using this initial NN, and others trained with all 23 data sets using both the LM and BR algorithms. All performed about the same and consistently loaded material well. Some failure modes occurred occasionally, however, which are described in the next section.

Fig. 5: Scooping sequence with Neural Network controller.

IV. EXPERIMENTS

This section presents four experiments which evaluated and compared the performance of different NNs that were trained. Experiments 1-3 involved 26 different NN controllers and 97 individual automated scooping actions. Most NNs were tested 3-5 times, though a few only once or twice if they failed to come close to a proper scooping action. Half of the NNs were trained with LM and half with BR. Two additional NNs were trained using Gradient Descent backpropagation, however these failed to produce proper scooping actions and this method was no longer pursued.

Comparisons were made by evaluating bucket filling performance using a qualitative scale from 5 (fullest) to 0 (empty), since no direct load measurement capability was available. This scale is illustrated in Fig. 6. A level of 3 corresponds to the bucket being full with respect to its geometric shape, while 4 has some overfill and 5 significant overfill. A level of 2 is less than full, while 1 would be very little material and 0 empty. Some of these scores were noted during testing, but photos and videos were also used to check each result.



(a) 5 - significant overfill. (b) 4 - some overfill.

(c) 3 - approx. full, no overfill. (d) 2 - less than full.

Fig. 6: Qualitative scale (5 to 0) for evaluation of bucket fill performance. Not shown: 1 - little material, and 0 - empty.

A. *Failure Modes*

During the experiments, some of the NN controlled scooping actions exhibited various failure modes which can be categorized below. The number of occurrences out of the 97 total tests in Experiments 1-3 is also mentioned in parentheses, for LM and BR. In contrast to failure mode 1, material was still usually collected in the bucket when modes 2-5 occurred:

1) digging motion started by bucket before reaching pile (not counted - see below)
2) slow, continued lifting of boom after load extracted (not counted - see below)
3) driving up slope after load extracted (LM-7, BR-0)
4) engine stall when extracting load (LM-4, BR-3)
5) non-smooth driving, bucket actuation and/or excessive wheel skid (LM-2, BR-1).

Failure mode 1 was found to occur when the starting position of the bucket was too high off the ground, making the NN seem to think digging had already started. This was difficult to preclude, because when manually positioning the bucket, the exact joint angles are not known. The bucket was generally positioned low and level, but with some elevation to avoid excessive scraping along the ground. When this occurred, however, the result was neglected since the error was likely due to the starting position of the bucket and not indicative of the NN's scooping performance. During Experiments 1-3 this occurred about 10 times, but these are not included in the 97 total tests.

Failure mode 2 was likely due to boom raising control signals being present in some of the training data after the main scooping action had ended, causing some NNs to generate small boom raising signals after extraction. This was not considered a major failure, however, since in an implemented system, the NN could be disengaged after the bucket is raised a certain amount.

When analyzing recorded signals from the initial testing, it was found that spikes in the gas command occasionally occurred near the end of the scooping action. This caused the problem of driving up the slope after extraction, and tended to happen only with LM NNs. The reason for this, however, is not well understood, and highlights a challenge in using NNs as controllers: unpredictable behaviours can arise, and some hard-coded safety limits may be needed to prevent accidents. This particular problem could again be avoided by disengaging the NN after the bucket has been lifted a certain amount.

Engine stall sometimes occurred when digging too deep. This could perhaps be mitigated by collecting more training data with different soil properties. Non-smooth driving and bucket actuation tended to happen with too little training data being used (less than 5 sets), and also with too many (16) hidden neurons.

B. Possible Error Sources

During the experiments, an attempt was made to begin each automated scooping action with initial conditions as similar as possible. For each test, scooping took place at the same pile location, within 1 m to the right or left. Before collecting the training data, and before each experiment, an initial scooping action was made at the location to loosen the material if it appeared to have become compacted.

Each of the four experiments was conducted within a few hours on a particular day, therefore ground material conditions stayed about the same during an experiment. The 4 different experiment days had similar dry weather, without much moisture in the soil. Nevertheless, possible sources of error which could have affected the bucket fill include:

- starting bucket position
- approach angle into pile
- change in shape of bottom pile contour over successive scooping actions
- change in ground material properties over successive scooping actions.

C. Experiment 1: Repeatability of NN Training

This experiment was done to test the performance of NNs with 5 hidden neurons, trained separately with all 23 data sets. 3 NNs were trained using LM and 3 using BR. This test was done because it was noticed that during the training process, there was often a big variation in the number of epochs needed before training stopped, even when the same data was used in the same order. This number ranged from less than 100 to several hundred. All NNs tested here, however, had the order of the training data randomly shuffled.

The average qualitative bucket fill is plotted in Fig. 7. The third BR NN had noticeably lower performance, however all others were approximately the same.

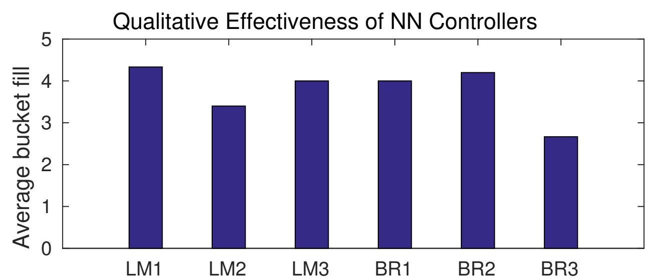


Fig. 7: Effect on bucket fill of retraining with the same data.

D. Experiment 2: Effect of Amount of Training Data

NN controllers with 5 hidden neurons (both LM and BR) were tested which were trained with 23, 20, 15, 10, 5 and 3 sets of training data, selected randomly from the available pool of 23 sets. The results are summarized in Fig. 8, and show a trend in which the bucket filling performance decreases as less training data is used. Both methods (LM and BR) have similar results.

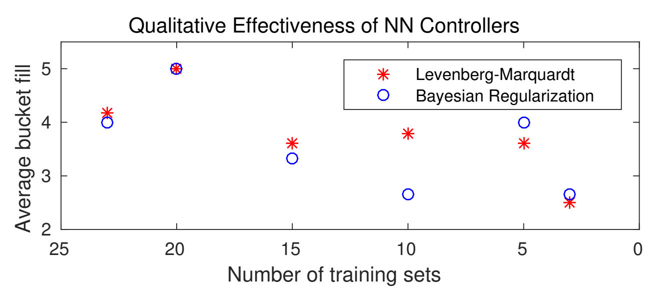


Fig. 8: Effect of amount of training data on bucket fill.

E. Experiment 3: Effect of Number of Neurons

In this experiment, LM and BR NNs were tested which were trained with all 23 data sets, with different numbers of neurons in the hidden layer, including 2, 4, 8, 10 and 16. The results are shown in Fig. 9. It appears that 8-10 hidden neurons achieved the best performance with both methods, though they still performed well with only 2 hidden neurons. Although the LM 16-neuron NN achieved a high average bucket fill, it came at the cost of one engine stall and some jerky motion and wheel skidding during the 3 trials. The BR

16-neuron NN was only tested once as it did not result in a full scooping action. The poor performance with 16 hidden neurons may be a result of over-fitting to the training data.

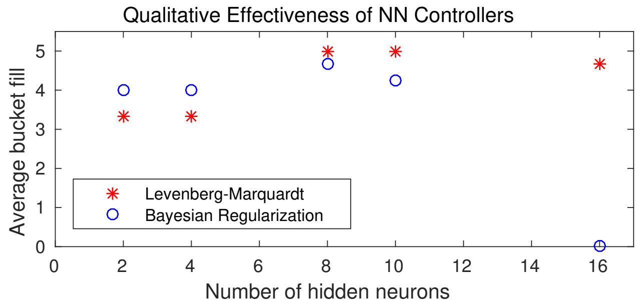


Fig. 9: Effect of number of hidden neurons on bucket fill.

F. Experiment 4: Quantitative Comparison of Scooping Control Methods

To evaluate the NN control vs. a more traditional scooping automation method, an NN with 5 hidden neurons, using LM and trained with all 23 data sets, was compared with a heuristic-based automated bucket controller which was developed. The heuristic controller, implemented in the Avant's Simulink model, was developed after analyzing the training data, and tried to replicate the manually-controlled strategy which was outlined in Section III. I.e. after the initial driving pressure rise is detected on pile contact, the boom is lifted while forward driving continues. When the boom reaches a certain height, the bucket is curled upwards and driving stops. This follows a similar heuristic as Sarata et al. [2].

When developing the heuristic controller, it was difficult to estimate what control signal magnitudes should be used for driving and moving the boom and bucket. Initially there was a problem of digging too deep, resulting in engine stall, and about 5 cycles of parameter tuning were required.

To compare the controllers, a pile transfer job of 5 load-haul-dump cycles was repeated, with hauling and dumping controlled manually. Manually controlled scooping was also included for further comparison. Scooping actions were distributed across a $\sim$3 m-wide section of the slope, with the Avant oriented to make perpendicular scooping approaches. After each job, the dump pile material was approximately replaced at the same location on the slope, to try and re-create the experiment as closely as possible.

Point clouds were collected of the dump pile area before and after each job, so that the volume of the end pile could be estimated, and hence the average bucket load using each method (see Fig. 10). The job time and work rate are here not considered. As Table I shows, the highest performance was achieved by manually controlling the Avant, however the NN controller outperformed the automated heuristic controller.

V. CONCLUSION AND FUTURE WORK

Automated scooping control for a robotic compact wheel loader was developed using an end-to-end NN trained by demonstration. Experiments 2 and 3 investigated the effect of amount of training data and number of hidden neurons on the bucket filling performance, and may help to provide general guidelines for choosing NN design and training parameters.

Fig. 10: Pile-transfer test for bucket filling controller evaluation: (*left*) workspace with scooping area at left and dump pile at right; (*right*) surface model of dump pile after one job, used to estimate volume.

Control Method	Total Pile Volume (m^3)	Avg. Bucket Load (m^3)
Manual	1.120	0.224
NN	0.908	0.182
Heuristic	0.795	0.159

TABLE I: Results of 5-scoop pile transfer test using three scooping control methods.

Using the most training data available (23 sets) generally resulted in controllers which could successfully dig, though performance varied from one controller to another. Some NNs pushed harder into the pile than others, and some had the problem of constantly raising the bucket or trying to drive up the slope after digging.

Performance generally decreased with less training data, though 5 data sets was still sufficient. It appeared that 8-10 hidden neurons had the best performance, though 2 was also sufficient. Overall the LM and BR backpropagation algorithms resulted in about the same performance.

The pile transfer evaluation resulted in the highest average bucket load under human control. Although NNs may not yet be on par with human control for loading jobs, the NN controller here did outperform the heuristic controller which was developed. Given that the NN controller was easier to implement, without manual tuning of parameters, the end-to-end NN approach, with training by demonstration, was shown to be a promising method for automated pile loading compared with strategies based on heuristics.

For future work, collecting more training data in general would be desirable, also with different human operators and different material properties, to see if an NN controller can learn to dig effectively in different situations. More sophisticated NNs with multiple hidden layers could be tested, while adding more sensor inputs to the NN could also be attempted, such as wheel odometry, IMU accelerations, additional hydraulic pressures and the estimated distance to pile if available via vision or laser scanners. NN scooping controllers can also be incorporated into the fully-automated pile transfer system which has been demonstrated in the authors' previous work, so far using only a heuristic controller for scooping [18].

REFERENCES

[1] P. J. A. Lever and F.-Y. Wang, "Intelligent Excavator Control System for Lunar Mining System," *Journal of Aerospace Engineering*, vol. 8, no. 1, pp. 16–24, 1995.

[2] S. Sarata, H. Osumi, Y. Kawai, and F. Tomita, "Trajectory Arrangement Based on Resistance Force and Shape of Pile at Scooping Motion," in *IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, U.S.A., 2004.

[3] H. Almqvist, "Automatic bucket fill," Master's thesis, Linköping University, 2009.

[4] R. Filla, M. Obermayr, and B. Frank, "A study to compare trajectory generation algorithms for automatic bucket filling in wheel loaders," in *3rd Commercial Vehicle Technology Symposium (CVT)*, Kaiserslautern, Germany, 2014.

[5] A. A. Dobson, J. A. Marshall, and J. Larsson, "Admittance Control for Robotic Loading: Underground Field Trials with an LHD," in *Field and Service Robotics*, Toronto, Canada, 2015.

[6] S. Dadhich, U. Bodin, F. Sandin, and U. Andersson, "Machine Learning Approach to Automatic Bucket Loading," in *24th Mediterranean Conference on Control and Automation (MED)*, Athens, Greece, 2016.

[7] H. A. Fernando, J. A. Marshall, H. Almqvist, and J. Larsson, "Towards Controlling Bucket Fill Factor in Robotic Excavation by Learning Admittance Control Setpoints," in *11th Conference on Field and Service Robotics (FSR)*, Zurich, Switzerland, 2017.

[8] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, 2009.

[9] S. Hu, M. H. Ang Jr., and H. Krishnan, "Neural Network Controller for Constrained Robot Manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, U.S.A., 2000.

[10] L. Jin, S. Li, J. Yu, and J. He, "Robot manipulator control using neural networks: A survey," *Neurocomputing*, vol. 285, pp. 23–34, 2018.

[11] M. L. Corradini, G. Ippoliti, and S. Longhi, "Neural Network Based Control of Mobile Robots: Development and Experimental Validation," *Journal of Robotic Systems*, vol. 20, no. 10, 2003.

[12] M. T. Hagan, H. B. Demuth, and O. De Jesús, "An introduction to the use of neural networks in control systems," *International Journal of Robust and Nonlinear Control*, vol. 12, pp. 959–985, 2002.

[13] N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, D. Fox, J. Leitner, B. Upcroft, P. Abbeel, W. Burgard, M. Milford, and P. Corke, "The limits and potentials of deep learning for robotics," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 405–420, 2018.

[14] U. Viereck, A. ten Pas, K. Saenko, and R. Platt, "Learning a visuomotor controller for real world robotic grasping using simulated depth images," in *1st Conference on Robot Learning (CoRL)*, Mountain View, U.S.A., 2017.

[15] R. Antonova, A. Rai, and C. G. Atkeson, "Deep Kernels for Optimizing Locomotion Controllers," in *1st Conference on Robot Learning (CoRL)*, Mountain View, U.S.A., 2017.

[16] P. Drews, G. Williams, B. Goldfain, E. A. Theodorou, and J. M. Rehg, "Agressive Deep Driving: Combining Convolutional Neural Networks and Model Predictive Control," in *1st Conference on Robot Learning (CoRL)*, Mountain View, U.S.A., 2017.

[17] S. James, A. J. Davison, and E. Johns, "Transferring End-to-End Visuomotor Control from Simulation to Real World for a Multi-Stage Task," in *1st Conference on Robot Learning (CoRL)*, Mountain View, U.S.A., 2017.

[18] E. Halbach, A. Kolu, and R. Ghabcheloo, "Automated Pile Transfer Work Cycles with a Robotic Wheel Loader," in *17th International Conference on Computing in Civil and Building Engineering (ICCCBE)*, Tampere, Finland, 2018.

[19] H. Yu and B. M. Wilamowski, *Levenberg-Marquardt Training*, 2011, vol. 5, ch. 12, pp. 1–15.

[20] D. J. C. MacKay, "A practical Bayesian framework for backpropagation networks," *Neural Computation*, vol. 4, no. 3, pp. 448–472, 1992.

[21] MATLAB, *Release 2015b*. Natick, Massachusetts, U.S.A.: The MathWorks, Inc., 2015. [Online]. Available: http://www.mathworks.com