What Lies Beneath: Material Classification for Autonomous Excavators using Proprioceptive Force Sensing and Machine Learning

Heshan Fernando\textsuperscript{1,}\textsuperscript{*}, Joshua Marshall\textsuperscript{1,2}

\textit{Ingenuity Labs Research Institute, Queen’s University, 69 Union St W, Kingston, Ontario, K7L 3N6, Canada}

Abstract

The ability of robotic excavators to acquire meaningful knowledge about materials during digging can augment their autonomous functionality, as well as optimize downstream operations in construction and mining. Some material properties, such as rock sizes, can be determined visually, but these methods cannot see what lies beneath. In this work, a classification methodology that utilizes only proprioceptive force data acquired from an autonomous digging system and machine learning algorithms is proposed for excavation material identification. The consistent performance synonymous with autonomous digging systems allows for the use of basic features extracted from the force data for classification. A proof of concept of this novel approach to excavation material classification is demonstrated through a binary classification of rock and gravel materials. Force data were obtained from full-scale autonomous loading trials with a 14-tonne capacity load-haul-dump machine at a mining and construction test facility. Preliminary results achieved a classification accuracy of 90 \%.

Keywords: material classification, force sensing, machine learning, robotic

\textsuperscript{*}This project was funded in part by Epiroc AB (original data collection), NSERC Project RGPIN 2015-04025, and the NSERC Canadian Robotics Network (NCRN).

\textsuperscript{*}Corresponding author

Email addresses: heshan.fernando@queensu.ca (Heshan Fernando), joshua.marshall@queensu.ca (Joshua Marshall)

\textsuperscript{1}Department of Mechanical and Materials Engineering

\textsuperscript{2}Department of Electrical and Computer Engineering
excavation, loader automation

1. Introduction

Despite a large body of research and engineering work in robotic excavation [1], spanning decades, commercially viable systems for fully autonomous excavation are still in their infancy. Advances in control systems development towards autonomous digging and scooping of fragmented rock [2, 3] have enabled semi-autonomous load assist functions for robotic wheel loaders [4], such as the Epiroc ST14 Scooptram load-haul-dump (LHD) machine shown in Figure 1. These functions facilitate higher, more consistent, bucket fill factors when compared to manual loading operations. However, performance is strongly influenced by the conditions of interaction between the excavator and the material, which are unknown and changing throughout the excavation process. For example, material such as fragmented rock is not homogeneous in size and composition, can be wet and sticky, or dry, depending on local factors, and it inherently moves and exposes new hidden material that is not visible prior to executing the excavation process. A fully autonomous solution for robotic excavation must be able to identify and adapt to these variable conditions in order to achieve consistent bucket filling performance. Such a system would have obvious applications in construction [5], military [6], mining [7], and space exploration/development [8] where humans cannot be present for safety, logistical, and/or cost reasons.

One promising approach to autonomous excavation is a learning-based dig control strategy that has been developed through extensive field experimentation for robotic wheel loaders [3, 7, 9]. The dig controller is an admittance-based interaction control strategy, which utilizes force feedback to regulate bucket motion for scooping material. This control strategy has demonstrated consistent bucket filling performance in full-scale field experiments, but its parameters require different tuning when excavating material with different characteristics [3]. Recent research investigated augmenting the dig controller with an iterative learning algorithm to automatically tune control parameters for a desired bucket
fill weight in different material types, such as rock and gravel [7]. However, the algorithm requires multiple excavation passes to converge to the target bucket fill weight. Thus, iterative learning may not be suitable for some excavation scenarios, such as in mining, where material characteristics might change with every excavation pass. If the material type can be identified during the autonomous digging process, then the control parameters could be adapted faster for even more consistent bucket filling at every excavation pass. What is more, improved knowledge about the specific characteristics of the excavation media is directly useful to civil/mining engineers to optimize blast designs and downstream rock crushing systems.

This paper proposes a material identification methodology for autonomous excavators that uses proprioceptive sensor data obtained from the dig controller’s force feedback signal and machine learning (ML) algorithms for classification. The idea is to classify material types by categorizing data extracted from the dig controller’s force feedback signal. If the material can be correctly identified, then the dig controller parameters could be automatically adapted
(e.g., using a lookup table or local optimization). The advantage of the proposed methodology is the consistency in machine-material interaction that is achieved through automation via the dig controller. This consistency in the force data enables the use of basic signal processing and straightforward learning algorithms for classification (e.g., compared to operator data which may be highly variant within a particular class of material). Consequently, no operational complexity is added to existing excavators, and the method is applicable in dark, dusty and harsh environments such as those encountered at night, in underground mining, and in planetary development activities. Furthermore, the proposed method is independent of lighting conditions that challenge material identification techniques using exteroceptive sensing such as cameras and laser scanners that cannot see what lies beneath. In addition to enabling adaptation for autonomous excavation control, identifying the material of every bucket scoop has potential to significantly improve downstream operations such as sorting and blending (e.g., in mining). In this work, a proof of concept of the proposed methodology is validated for a binary classification of rock and gravel material types. Force data is obtained from full-scale autonomous excavation trials with an Epiroc ST14 LHD machine at a mining/construction machine test facility located near Kvarntorp, Sweden.

1.1. Further Related Work

There exist early works on autonomous excavation that focus on parametric modeling of soil-tool interactions [10, 11] for control development, and real-time identification of soil parameters for control optimization [12, 13, 14]. However, these models are not applicable to fragmented rock, which is significantly different from soil and other homogenous material. Due to difficulties in explicitly modelling heterogenous excavation materials, this work proposes to use data-driven machine learning methods for identifying different material classes, which can then be used for autonomous excavation control optimization. The following subsections describe similar approaches in relevant robotics applications.
1.1.1. Vision-based Material Identification

Others have used exteroceptive sensing vision and LiDAR-based systems for estimating size distribution of blasted rock [15, 16, 17]. While rock size distribution does have an impact on digging, vision-based systems only see the surface and are usually not feasible in underground scenarios. Cameras, in particular, suffer from challenges with poor lighting underground. Force sensing can sense beneath the surface, and is a key measurement signal in several recently developed autonomous excavation algorithms [7, 18, 19]. In these algorithms, the force signal has been used for feedback control [7], disturbance observation [19], and regression of actuator commands [18]. As far as we know, no one else has used force sensing (or another proprioceptive sensing approach) for material identification or classification in excavation.

1.1.2. Machine Learning for Classification

Machine learning (ML) is the study of algorithms that can learn and make predictions (and generalize) on empirical data [20]. Loosely speaking, there are two main types of types learning algorithms: supervised and unsupervised. Supervised learning minimizes an error/cost based on differences between targets and outputs (correct responses), while unsupervised is about clustering data when prior training data is not available. In the context of robotic excavation, unsupervised learning methods are preferred as all possible classes of materials may not be known at the time of training. In this work, both types of learning are investigated to compare performances when classifying rock and gravel materials.

In the robotics context, others have used proprioceptive sensor data and ML methods for robotic applications related to excavation and/or terrain classification. For example, [21] investigated the identification of wheel-loader work cycles in construction tasks. Some researchers have proposed the use of ML techniques for terrain classification from the direct interactions of mobile equipment with changing terrains; e.g., see [22, 23, 24, 25]. In [26], proprioceptive sensing was used to estimate slippage and immobilization of a mobile robot in
planetary exploration applications. Perhaps most similar to our application, 
[27] used tactile sensors to classify interactions between a robot hand and its environment. There are similar works in this area (e.g., [28, 29, 30, 31, 32]) that propose the use of ML to relate complex tactile data to object and material classes. In this paper, we aim to use a similar approach in that we propose the novel application of ML and proprioceptive sensing to classify the excavation media in industrial excavation activities. This is facilitated through the combination of these methods with an autonomous excavation technology.

1.2. About this Paper

As a first step in the development of the proposed material classification methodology, a binary classifier for rock and gravel materials is developed and validated using proprioceptive force data obtained from full-scale autonomous excavation trials. The aim of this work is to show the sufficiency of the proprioceptive force data obtained from an autonomous excavator to classify different excavation materials. Section 2 provides an overview of the methodology for binary classification of rock and gravel materials. Section 3 describes the autonomous excavation experimental setup used to acquire force data set for testing the classifier. Section 4 describes the processing and feature selection from the force signals. Section 5 provides background information on the learning algorithms that are used for classification in this work. Section 6 presents the classification results and Section 7 offers some concluding remarks, including limitations of the current work and future directions.

2. Binary Classification of Rock and Gravel

Excavation materials can be classified into numerous categories (e.g., based on type, mechanics, rock size or fragmentation). In this work, only a binary classification of rock and gravel material types is investigated to test the sufficiency of the proposed material classification methodology. A block diagram of the methodology to test the binary classification is shown in Figure 2 which
addresses the relevant machine learning operations such as signal processing, feature selection, classification and evaluation. Force data for validation were collected from full-scale autonomous excavation trials in real mining environments, which are described in Section 3.

First, force signals obtained from autonomous excavation trials with a rock pile and a gravel pile are analyzed and processed to find properties (i.e., features) that can be used as inputs to the classifiers. The goal of signal processing and feature selection is to find features that are very similar (i.e., invariant) for materials in the same category and very different (i.e., distinguishable) for materials in different categories [20]. The engineering objective is to find an ideal set of features that yields a representation to make the job of the classifier trivial. Of course, this is a difficult task for complex systems, but finding a good set of invariant and distinguishable features enables the use of simple classifiers. In contrast, omnipotent classifiers (e.g., deep learning) do not require the help of a sophisticated feature extractor [20]. However, deep learning methods require extremely large amounts of data for training, which is impractical for excavation material classification because material characteristics could change before sufficient training data becomes available.

Second, the features are formatted into a labelled data set for training and testing machine learning algorithms. The data set consists of input features $X$ and target classes $t$. $X$ is an $n \times m$ matrix, where $m$ is the number of features in each feature vector, and $n$ is the total number of feature vectors; $t$ is a $n \times 1$ vector containing the target class labels, “rock” or “gravel”, for the feature vectors. The feasibility of classifying “rock” and “gravel” materials using this data set is evaluated using two supervised learning algorithms, $k$-nearest neighbours (KNN) and artificial neural networks (ANN), and one unsupervised learning algorithm, $k$-means. There are many other potential classification algorithms such Support Vector Machines, Decision Trees, and Naive Bayes; however, KNN, ANN and $k$-means were selected because of their simplicity, popularity, and demonstrated suitability in classification applications—particularly in robotics (e.g., [28] [22] [26] [29]). The three algorithms are described with further detail in
Supervised learning algorithms train a classification model how to map input features to an output category by presenting the model with example input-output pairs. Thus, supervised learning requires the data set to be divided into training data and testing data. Typically, a ratio of 80/20 is recommended for dividing the data into training/testing sets [20]. The testing set contains new data, unseen by the model during training, to validate the model’s performance.

In contrast, unsupervised learning algorithms do not have an explicit teacher. Instead, the classifier performs clustering, or grouping, of similar input patterns based on a cost function or distance metric [20]. Thus, in this work, unsupervised learning is always performed on the complete data set, as shown in Figure 2.

Features can have different ranges of values, so it is good practice to scale the input data when implementing machine learning algorithms. Many learning algorithms utilize distance metrics and gradient descent for learning, so scaling
the inputs prevents features with larger numerical values from dominating these calculations. If the input data is not scaled, the learned models can become biased to major features. Standardizing the data set by subtracting the means \( \mu \) and dividing by the standard deviations \( \sigma \) of each feature component \( x \) in \( X \) is an appropriate method to do this [20]. Thus, we use the standard relations

\[
\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{1}
\]

\[
\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2} \tag{2}
\]

\[
z_i = \frac{x_i - \mu}{\sigma} \quad (i = 1, 2, ..., n), \tag{3}
\]

where \( z_i \) is the standardized value of the feature \( x_i \) in \( x \). Doing so gives each feature component a mean value of 0 and a standard deviation of 1.0.

The performance of all algorithms are evaluated in terms of classification accuracy. For supervised learning algorithms, accuracy is defined as the number of correctly classified samples divided by the total number of samples in the test data set. For unsupervised learning, accuracy is defined in terms of purity \( \bar{A} \in [0, 1] \), where each cluster \( \omega_k \) is assigned a class \( c_j \) that is most frequent in that cluster, and the accuracy of this assignment is measured by counting the number of correctly assigned class members in each cluster and dividing by the total number of samples [33]. Thus, more formally

\[
\bar{A}(\Omega, C) = \frac{1}{n} \sum_{k=1}^{K} \max_{c_j \in C} |\omega_k \cap c_j|, \tag{4}
\]

where \( \Omega = \{\omega_1, \omega_2, ..., \omega_K\} \) is the superset of clusters and \( C = \{c_1, c_2, ..., c_J\} \) is the superset of classes. Given the binary classification (i.e., \( J = 2 \)) of “rock” and “gravel” materials in this work, the k-means algorithm is used to classify data into two clusters (i.e., \( K = 2 \)). For example, if a total of \( n = 86 \) rock \( (c_1) \) and gravel \( (c_2) \) data are clustered into two clusters \( (\omega_1 \) and \( \omega_2) \), and the clustering results are \( \omega_1: 2 \) rock and 37 gravel, and \( \omega_2: 40 \) rock and 7 gravel. Then the majority class of \( \omega_1 \) is gravel, and the majority class of \( \omega_2 \) is rock. Thus, the clustering quality in terms of purity is \( 1/86 \times (37 + 40) = 0.90 \). This evaluation
measure is simple and transparent. The reader is referred to [33] (Chapter 16) for an excellent overview the types of clustering evaluation measures, including cluster purity.

2.1. Software Tools

Design and evaluation of the material classification methodology is carried out using MATLAB numerical computing software (version 2020a). In particular, built-in functions in MATLAB’s Signal Processing Toolbox (version 8.4) and Statistics and Machine Learning Toolbox (version 11.7) are utilized for processing force signals, extracting features and designing learning algorithms for classification.

3. Data Collection

Data for evaluating the proposed material classification methodology were collected through full-scale autonomous excavation trials with a 14-tonne capacity load-haul-dump (LHD) machine at a mining test facility located near Kvarntorp, Sweden. Data collected in this manner provide a rare opportunity to validate the proposed material classification methodology using force data obtained from a real LHD machine; however, these experiments are very expensive and time-consuming, so only 86 excavation trials were feasible during this field campaign. Furthermore, only a rock and gravel pile were available at the time of the experiments. The experimental set-up is depicted in Figure 3. Details about the machine, excavated materials, autonomous excavation procedure and data set are described in the following subsections.

3.1. LHD Machine

The Epiroc ST14 Scooptram LHD machine, used for autonomous excavation trials in this work, can carry 14-tonnes and has a 335 hp engine. This machine’s functions can be fully automated through a programmable control system that interfaces to various onboard sensors and the machine’s actuators, including throttle, brake, and the electrohydraulic servo valves that actuate its lift and
dump cylinders. The loading mechanism of a typical LHD machine is shown in Figure 3. The systems and hardware relevant to autonomous excavation in this work are described below.

**Pressure Transducers**: Pressure transducers at the head \((h)\) and rod \((r)\) sides of one of the lift cylinders, as indicated in Figure 4, were used to measure the hydraulic pressures. The hardware specifications for pressure sensor are provided in Table 1. The measurements \(P\), along with the cylinder head and rod cross-sectional areas \(A\), were used to estimate the forces \(f = P_h A_h - P_r A_r\) for the dig controller’s feedback signal. The data for material classification is also extracted from this force signal.

**Linear Position Transducer**: A wire potentiometer within the dump cylinder was used to measure its extension length \(d\). The hardware specifications for the linear position transducer are provided in Table 1. The ST14 dump cylinder has a stroke length of 0.549 m.

**Boom Angle Encoder**: The lift cylinders are not actuated during digging; however, the boom angle measurement \(\theta\) was used to ensure the lift cylin-
Pressure sensor (force measurement)

Figure 4: Geometry of an Epiroc ST14 LHD machine’s loading mechanism.

ders were extended slightly above their mechanical stops for effective force measurement. The boom angle encoder was calibrated to measure $\theta = 0$ degrees when the lift cylinders were fully retracted (i.e., hitting their mechanical stops). The boom was lifted to $\theta = 2$ degrees for autonomous digging.

**Control System:** An on-board control system ran in real-time at a frequency of 20 Hz for reading sensors, sending actuator commands, and logging data. Thus, the time-step $T$ is 0.05 s for all recorded signals.

3.2. Excavated Materials

A rock pile located underground and a gravel pile located at the surface of Epiroc’s test facility near Kvarntorp, Sweden were used for the presented autonomous excavation and material classification trials. Images of these piles are shown in Figure 3. The rock pile consisted of a mixture of mud, fine gravel, and large fragments of blasted rock (30 to 70 cm in nominal diameter), which is
representative of a real rock pile found in underground mining. The gravel pile consisted mainly of fine dry gravel with smaller rocks and fines. These were the only two materials available at the test site at the time of the experiments.

3.3. Autonomous Excavation Setup and Procedure

The excavation operation was automated using the admittance-based dig controller developed in [7, 34]. A block diagram of the control structure is shown in Figure 5. The controller is implemented in the loader’s actuator space, which consists of the throttle, dump cylinder and lift cylinders. Only the dump cylinder is controlled during digging. Tractive effort is supplied through a constant input to the throttle actuator $u_t = \bar{u}_t$. The lift cylinders are not actuated $u_l = 0$; thus, hydraulic pressure measurements at the lift cylinders can be used as a force sensor to estimate the bucket-rock interaction forces $f$ that serve as feedback to the dump cylinder admittance controller $H(s)$. For autonomous digging, the admittance controller $H(s)$ computes a velocity change $v_f$, which is combined with a reference velocity $v_d$ to produce a new velocity command to the dump cylinder $v_c = v_d + v_f$. More details about the dig controller can be found in our earlier works [7, 34].

The dig controller was deployed on the ST14 machine to conduct the autonomous excavation trials in this work. The controller is only executed during the
digging phase of the process (i.e., after the pile had been penetrated). However, for consistency between trials, much of the machine initialization was automated. A Finite State Machine (FSM) representation of this process is shown in Figure 6. At the start of each trial, the LHD was manually positioned in front of the pile, the boom and bucket were automatically positioned at their entry positions \((d = 0.192 \text{ m and } \theta = 2 \text{ degrees})\); similar to the configuration shown in Figure 4, followed by driving forward at throttle \(u_t = 0.5\) and executing the dig controller when the pile was penetrated \((f > 2.5 \text{ MN})\). The values for throttle and force threshold for pile penetration in this setup were obtained through experimentation. The automated process ended when the dump cylinder was fully extended \((d = 0.549 \text{ m})\). Note that both the rock and gravel piles had finite amounts of material, so the material was dumped back onto the pile after each autonomous excavation trial.

3.4. Data Set

A total of \(n = 86\) force signals were available for the purposes of this research: 42 signals from rock pile excavation trials and 44 signals from gravel pile excavation trials. Figure 7 shows a plot of the force signal data set. All force signals are cropped to the Dig phase in the autonomous excavation pro-
cess. Each force signal is represented in discrete time as $f[k] = f(kT)$, where $k = 0, 1, 2, \ldots L$ is the time-step and $T$ is the sampling time (0.05 s). The length $L$ of each signal varies because different parameter values for $\bar{v}_d$ and $\bar{u}_t$ were used in the autonomous excavation trials.

4. Feature Selection

The feature space used as input to the machine learning algorithms for material classification was selected by processing and analyzing the force signals obtained from the autonomous excavation trials. Specific knowledge about the autonomous excavation process gained through observations in these field trials provided insight for signal processing and feature extraction. These are described in the following subsections.
4.1. Signal Processing

The force signals were processed by segmenting each signal into three phases, and by computing each signal’s time derivatives $\dot{f}$ and frequency power spectrum $|P(\nu)|$. The three phases are evident by qualitatively inspecting the data set shown in Figure 7. The first phase $f[0, t_1]$ is taken as the initial bucket curl, which causes the forces to sharply increase at the start of digging. The second phase $f[t_1, t_2]$ consists of a majority of the digging where the forces evolve due to bucket-material interaction. The third phase $f[t_2, L]$ is taken as the last bit of bucket curl where the forces begin to decrease, which is likely due to material falling to the back of the bucket. By visually inspecting the force signals and the dump cylinder extension signals $d$ for a subset of trials, it was determined that $t_1$ can be approximated by the time-step $k$ where $d = 0.220$ m, and $t_2$ can be approximated by the time-step $k$ where $d = 0.430$ m. The first subplots of Figures 8, 9, and 10 show the segmentation for a subset of the trials. Although not perfect, this process segments the signals to the three phases described above relatively well.

Figures 8, 9, and 10 show examples of signal processing results for three different sets of autonomous excavation trials. Each set groups together signals from excavation trials that used the same dig controller parameters $\bar{v}_d$ and $\bar{u}_t$. The first subplot in each figure shows the three segmentation phases of the force signal. The second and third subplots in each figure show the time derivatives $\dot{f}$ and the frequency spectra $|P(\nu)|$ of the respective force signals. The time derivatives $\dot{f}$ were computed using MATLAB’s numerical \texttt{gradient} function, which computes the derivative of a signal using central difference for interior data points and single sided differences for the edges of the signal. The frequency power spectra $|P(\nu)|$ were computed using MATLAB’s \texttt{fft} function, which computes the Discrete Fourier Transform (DFT) of a signal. Note that the frequency spectrum is only computed for the second phase of each force signal.

Figure 8: Example of force signal processing for feature extraction. These signals are from autonomous excavation trials with $\bar{v}_d = 0.03$ m/s and $\bar{u}_t = 0.4$.

$f[t_1, t_2]$. The lengths of the signals in the first and third phases are generally too short to obtain useful features with the DFT.

Several distinguishing features can be observed in the processed force signals shown in Figure 9. For example, the magnitude and variation of $\dot{f}[t_1, t_2]$ is much greater in rock compared to gravel. As well, the dominant peak frequency $\nu_1$ occurs at a lower frequency for rock compared to gravel. These observations were used to define useful features from the force signals for material classification.

4.2. Feature Extraction

Basic features such as means and standard deviations, were extracted from the processed and segmented force signals. After some inspection and analysis,
Figure 9: Example of force signal processing for feature extraction. These signals are from autonomous excavation trials with $\bar{v}_d = 0.04$ m/s and $\bar{u}_t = 0.3$. 

---

**Force Signal**

**Time Derivative of Force Signal**

**Power Spectrum of Force Signal in $[t_1, t_2]$**
Figure 10: Example of force signal processing for feature extraction. These signals are from autonomous excavation trials with $\bar{v}_d = 0.02 \text{ m/s}$ and $\bar{u}_t = 0.5$. 
seven features were selected for classification. These features are summarized in Table 2. MATLAB’s \texttt{mean}\footnote{https://www.mathworks.com/help/matlab/ref/mean.html} and \texttt{std}\footnote{https://www.mathworks.com/help/matlab/ref/std.html} functions were used to compute the means and standard deviations of the segmented signals. For reference, the \texttt{mean} features in Table 2 were calculated as

\begin{equation}
\text{mean}(g[p, q]) = \mu = \frac{1}{q - p} \sum_{i=p}^{q} g[i],
\end{equation}

and the \texttt{std} features in Table 2 were calculated as

\begin{equation}
\text{std}(g[p, q]) = \sqrt{\frac{1}{q - p} \sum_{i=p}^{q} (g[i] - \mu)^2}.
\end{equation}

Features $x_1$, $x_2$, $x_3$, $x_4$ were extracted from the first phase of the force signals. These features capture the average magnitude and variation of the forces $f$ and time derivatives of the forces $\dot{f}$ in the initial bucket curl. It was expected that these features would have higher values for rock than for gravel. Features $x_5$ and $x_6$ capture the average magnitude and variation of $\dot{f}$ in the second phase. As observed in Figure 9, these feature values are expected to higher for rock than for gravel. Feature $x_7$ is the frequency with the highest peak in the force signal’s power spectrum $P(\nu)$. The \texttt{findpeaks}\footnote{https://www.mathworks.com/help/signal/ref/findpeaks.html} function in MATLAB’s Signal Processing Toolbox was used to find the highest peak of each power spectrum. This function finds the local maxima (peaks) of the input signal vector. A local peak is a data sample that is larger than its two neighbouring samples. No distinguishable features were extracted from the third phase of the force signal $f[t_2, L]$.

\subsection*{4.3. Feature Sets}

The extracted features were standardized, using (1), (2) and (3), and analyzed using visualization tools and feature selection techniques in MATLAB to reduce the feature space and improve classification performance. Histograms of
the seven features are shown in Figure 11. It is evident that there is no single feature that can uniquely discriminate between the rock and the gravel materials. The features were further inspected using a Parallel Coordinates Plot as shown in Figure 12. This plot shows the median, the 25 percent quantile and the 75 percent quantile for the standardized feature values. There are only three features with no overlap between the two material classes are $x_2$, $x_5$ and $x_6$.

Two feature selection techniques in MATLAB’s Statistics and Machine Learning Toolbox were also utilized to check the independence of the seven standardized features and rank them in order of relevance. The first technique is a Filter Type Feature Selection method `fscchi2` which ranks features using the chi-square test. The ranking score is determined by the negative log of chi-square tests’s $p$-value. Numerical feature data are categorized into bins to perform the chi-square test. The default number of 10 bins was used for the feature data in this work. The second technique is a Sequential Feature Selection algorithm known as the minimum redundancy maximum relevance (MRMR) algorithm (`fscmrmr`), which finds an optimal set of features that is mutually and maximally dissimilar. The algorithm quantifies the redundancy and relevance using the mutual information of variables—pairwise mutual information of features and mutual information of a feature and the class.[35] The ranking scores obtained from these two feature selection techniques for the seven features are summarized in Table 3. The chi-squared test returned $x_7$ and $x_6$ as the top-ranked features for classification with $x_2$ and $x_5$ tied for third place, and the MRMR algorithm returned $x_3$ and $x_6$ as top ranked features with the rest of the features receiving a zero score.

Based on the above analysis, five feature sets were formed to evaluate the classification performance. The first feature set $S_1 = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ consist of the entire feature set; the remaining four feature sets are subsets of $S_1$. The second feature set $S_2 = \{x_2, x_5, x_6\}$ consist of the top three features based

---

8https://www.mathworks.com/help/stats/fscchi2.html
Figure 11: Histogram analysis of the standardized feature set.
on the Parallel Coordinates Plot feature analysis. The third feature set consists of four features $S_3 = \{x_1, x_2, x_3, x_4\}$, which are the features extracted from the first phase $f[0,a]$ of the force signal. The idea behind testing this feature set is to evaluate the possibility of early classification of the material, which can then be used to adapt the control system more quickly. The fourth feature set $S_5 = \{x_3, x_6\}$ consist of the top two features from the MRMR feature selection algorithm. Finally, the fourth feature set $S_4 = \{x_2, x_5, x_6, x_7\}$ consists of the top four features from the chi-square feature selection method.

5. Learning Algorithms

This section describes the two supervised learning algorithms, $k$-nearest neighbours (KNN) and artificial neural network (ANN), and the unsupervised learning algorithm $k$-means used for the proposed material classification methodology.

Figure 12: Parallel coordinates plot.
5.1. k-Nearest Neighbours

The k-nearest neighbour algorithm is a simple and efficient classifier. It categorizes a new input \( x' \) by assigning it the label of the majority samples \( x \) among its \( k \) nearest samples (with \( k \) odd to avoid ties) [13]. The algorithm is initialized using a set of training data. The nearest neighbours of a new input \( x' \) are determined using a distance metric. In this work, the Euclidean distance \( ||x' - x_i||^2 \) is used.

A \( k \)-fold cross-validation approach, with \( k = 5 \), is used to train and test the KNN classification algorithm. In this approach, the entire input feature set is randomly ordered and split into five groups (i.e., folds). Then each unique group of features is tested after training the KNN algorithm with the remaining four groups. This is repeated for each of the five groups of data. The five testing evaluation results are averaged and reported as the classification performance of the algorithm.

5.2. Artificial Neural Network

Feedforward multi-layer perceptron (MLP) networks, trained using supervised gradient descent methods such as backpropagation, are the most commonly used artificial neural networks (ANN) for pattern recognition and classification [13]. The simplest configuration is a three-layer network, which consist of an input layer, hidden layer and an output layer as shown in Figure 13. The units in each layer are interconnected by modifiable weights \( w \) in a feedforward configuration. Thus, signals are processed in one direction, from input to output. Furthermore, there is a single bias unit \( b \), which serves as a threshold for activating each unit, connected to every unit other than the input units.

The number of input units is determined by the number of input features \( m \), and the number of output units are determined by the number of classes \( c \). The number of hidden units \( n_H \) is a design parameter. The input layer simply passes through an \( m \)-dimensional input feature vector to each unit in the hidden layer. Each hidden unit computes the weighted sum of its inputs to form a scalar net
activation

\[ a_j = \sum_{i=1}^{m} x_i w_{ji} + b, \quad (7) \]

where \( w_{ji} \) denotes the connection weight from input unit \( i \) to hidden unit \( j \).

Each hidden unit emits an output that is a non-linear function of its activation
\[ y_j = f(a_j). \quad (8) \]

In this work, a sigmoid activation function is used in the hidden units
\[ y_j = \frac{1}{1 + e^{-a_j}}. \quad (8) \]

Similarly, each output unit \( k \) computes its net activation based on the hidden signals \( y_j \) and the connection weights between the hidden units and the output units \( w_{kj} \), so that
\[ a_k = \sum_{j=1}^{n_H} y_j w_{kj} + b. \quad (9) \]

In this work, a softmax activation function is used for the output layer. Formally, the softmax function is calculated as
\[ z_k = \frac{e^{a_k}}{\sum_{i=1}^{c} e^{a_i}}, \quad (10) \]

which transforms the maximum output to 1.0 and reduces all other outputs to 0.0. For rock and gravel material classification, two output nodes are used (i.e., \( c = 2 \)). Each node is trained to output a value of 1.0 for the class, and 0.0 if not.

Once the network configuration is selected, the connection weights must be set such that the network provides the desired output for a given input. The
backpropagation algorithm, which is used in this work, is one of the simplest and most general methods to do this. The basic approach is to first randomly initialize the network weights \( w \). Next, the network is presented with input patterns \( x \) from a training data set, which are also labelled with the desired target outputs \( t \). For each input pattern, the error between network output \( z \) and the target output \( t \) is calculated as the mean squared error

\[
J(w) = \frac{1}{c} \sum_{k=1}^{n} (t_k - z_k)^2.
\]

(11)

Using a method based on gradient descent, the weights are updated in a direction that will reduce the error

\[
\Delta w_{pq} = -\eta \frac{\partial J}{\partial w_{pq}},
\]

(12)

where \( w_{pq} \) is the connection weight from unit \( p \) to unit \( q \), and \( \eta \) is merely a learning rate that indicates the relative size of the change in weights.

5.3. \( k \)-Means

The unsupervised learning algorithm, \( k \)-means, partitions \( n \) observations \( x_1, x_2, \ldots, x_n \) into \( k \) clusters. To begin, \( k \) centroids (or means) \( \mu_1, \mu_2, \ldots, \mu_k \) are initialized in the input space. For example, this can be done by randomly choosing \( k \) observations from the data set. The value of \( k \) may be assigned based on the final application (i.e., the number of desired classes). In this work, \( k = 2 \) is used for the binary classification of rock and gravel materials.

The clustering algorithm is implemented as follows. First, each observation \( x \) is classified to the nearest centroid \( \mu_j \) using a distance metric such as Euclidean distance \( ||x_i - \mu_j||^2 \). After all \( n \) observations are classified, the centroids are recomputed as the means of their clusters

\[
\mu_j = \frac{1}{N_j} \sum_{i=1}^{N_j} x_i
\]

(13)

where \( N_j \) is the number of observations belonging to the cluster \( j \). The algorithm is iterated until no observation reassignments are made and the centroids stop moving.
6. Results and Discussion

The binary classification of rock and gravel material was evaluated using the five feature sets \( S_1, S_2, S_3, S_4 \) and \( S_5 \) described in Section 4 and the three machine learning algorithms presented in Section 5. To ensure better reliability in classification results, different structures and multiple initializations were tested for each learning algorithm as described below.

The \( k \)-Nearest Neighbour (KNN) algorithm was tested with \( k=1 \) and \( k=5 \) nearest neighbours (labelled 1-NN and 5-NN, respectively). The Artificial Neural Network (ANN) was tested with \( n_H = 2 \), \( n_H = 4 \) and \( n_H = 7 \) hidden layers (labelled ANN-2, ANN-4 and ANN-7, respectively). The \( k \)-means algorithm was tested using two centroids (i.e., 2-means) as only a binary classification is considered in this paper.

For the unsupervised learning algorithms, the standardized data set is divided into training and testing data. The KNN algorithms were trained and tested using a 5-fold cross validation. The ANN algorithms were tested using a 70/15/15 split for training/validation/testing. For the \( k \)-means algorithm, the standardized feature sets were inputted into the algorithm directly for clustering.

6.1. Classification Results

Performance is evaluated in terms of classification accuracy for the two supervised learning algorithms (KNN and ANN) and in terms of cluster purity for the unsupervised algorithm (\( k \)-Means). Classification accuracy is calculated by dividing the number of correctly classified test data by the total number of test data, and cluster purity is calculated by using (4). To compensate for sensitivity of these algorithms to initialization, each algorithm was evaluated using 50 independent runs with random initialization of the data, weights, etc. Figure 14 presents the classification results as averages of the 50 independent trials for each algorithm; the error bars indicate the standard deviation of the classification accuracy results.
Figure 14: The classification accuracies of the six different algorithm structures for the five different feature sets. The bars indicate the average classification accuracy of 50 independent runs for each algorithm, and the error bars indicate the standard deviations.
Overall, high classification accuracies are obtained using the various feature sets and classification algorithms. The highest classification accuracy is 95% using the 1-NN algorithm with both \(S_1\) and \(S_2\) feature sets. The lowest classification accuracy is 75% using the ANN-2 algorithm and the \(S_3\) data set. Comparisons of the classification performances between the feature sets and the classification algorithms are discussed next.

6.2. Comparison of Feature Sets

The average classification accuracies of the five feature sets across the six algorithms are summarized in Table 4. \(S_1\), \(S_2\), and \(S_5\) achieved the highest classification accuracies (90% to 91%). Given that \(S_2\) and \(S_5\) are subsets of \(S_1\), it is clear that some of the features in \(S_1\) are redundant. The lower performance of \(S_3\) indicates that the features in the first phase of the signal \(f[0, t_1]\) is not sufficient to distinguish between rock and material. Different features may be required for improved classification early in the digging phase. Another method that could be investigated for early classification during digging is to extract features from a sliding window of the force signal. \(S_4\) also had lower performance, but this may be due to the feature set’s low dimensionality.

6.3. Comparison of Classifiers

Using \(S_2\) classification results as a benchmark, the six algorithms are compared. Out of the five supervised classifiers, the best performance was achieved with the 1-NN (95% accuracy). However, the 1-NN algorithm tends to over-fit the decision boundaries, as shown in Figure 15(a) which is not desirable in a classification algorithm. The 5-NN algorithm creates a more generalized decision boundary, shown in Figure 15(b) which provides more realistic classification results. In this case, the 5NN algorithm provided 93% classification accuracy, which is still very good.

The three ANN algorithms have similar classification performances for \(S_2\) (average of 89%); however, their classification performance has more variation.
in the 50 independent runs compared to other algorithms. This indicates that
the ANNs are more sensitive to initialization than other algorithms.

The $k$-means classifier also achieves good performance (85%) with the $S_2$
data set. The resulting clusters are shown in Figure [16]. Similar amounts of
gravel and rock feature sets are misclassified in this case.

6.4. Limitations of This Work

The high classification rates from the binary classification of rock and gravel
have proved the sufficiency of using proprioceptive force sensing and machine
learning for material classification; however, given that only two types of ma-
terial are available, these rates may not be so surprising. In practical civil and
mining applications, such as autonomous excavation control adaptation or op-
timizing blasting and crushing operations, there are many other categories of
excavation materials to identify. These categories may be based on parameters
such as soil type, mechanics, or rock size. In these scenarios, the assumptions
with respect to force sensing, basic features and simple classification algorithms
may have limited classification performance. For example, we have assumed
force to be the variable that governs the classification in the proposed method-
ology. Furthermore, in this work, the force measurement is acquired from the
dig controller’s feedback signal. Given more categories of excavation materials
that in practice require different control strategies, or even a different location
for the force sensor on the vehicle, the measurement signals could vary signifi-
cantly. This may require more sophisticated features to be extracted (e.g., via
Deep Feature Synthesis) or advanced classification algorithms. Further experi-
mentation and development is required to address these limitations.

7. Conclusions and Future Work

This paper presented a novel methodology for identifying excavation mate-
rials using only force data acquired from an autonomous digging system and
machine learning algorithms. Because the method only depends on propriocep-
tive sensing, it is applicable in dark, dusty and harsh environments that limit
Figure 15: Decision boundaries for $k$-nearest neighbour classification.
Figure 16: $k$-means unsupervised clustering results.
the use of exteroceptive sensor data. Furthermore, utilizing the consistent data acquired from an autonomous digging system enables the use of simple features and basic classification algorithms for material identification, which provides low operational complexity and computation overhead. The work presented in this paper, which evaluated the binary classification of rock and gravel materials as a proof of concept, demonstrated high classification accuracy (90% average) across the three machine learning algorithms that were tested (KNN, ANN, k-means). These results are promising because the data for classification were acquired from full-scale autonomous excavation experiments.

Future work should investigate scaling the method to identify more classes of materials, which could potentially include material characteristics such as rock size distributions. Integrating other proprioceptive sensing modalities, such as vibrations and sound, could also be investigated to improve classification accuracy. This material identification methodology has practical significance as it could be used to improve autonomous operation of robotic excavators, as well as provide useful knowledge about excavation materials to civil/mining engineers to help improve downstream operations.

Acknowledgements

The authors would like to thank Håkan Almqvist and Johann Larsson for their assistance with the field experiments; as well as Ola Petersson, Jörgen Appelgren, and Robert Lundh for many useful discussions; and Qing Huang for assistance with programming.
References


37


Table 1: Specifications for pressure and linear position transducers

<table>
<thead>
<tr>
<th>Specification</th>
<th>Pressure transducer</th>
<th>Linear position transducer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer</td>
<td>Danfoss</td>
<td>MTS Temposonics</td>
</tr>
<tr>
<td>Measurement range</td>
<td>0 to 350 Bar</td>
<td>0 to 550 mm</td>
</tr>
<tr>
<td>Analog voltage range</td>
<td>0.5 to 4.5 VDC</td>
<td>0.25 to 4.75 VDC</td>
</tr>
<tr>
<td>Accuracy</td>
<td>&lt; ±0.05% F.S.</td>
<td>&lt; ±0.04% F.S.</td>
</tr>
<tr>
<td>Response time</td>
<td>&lt; 4 ms</td>
<td>unknown</td>
</tr>
<tr>
<td>A2D conversion</td>
<td>10-bit</td>
<td>10-bit</td>
</tr>
</tbody>
</table>

Table 2: Descriptions of extracted features.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>$\text{mean}(f[0, t_1])$</td>
</tr>
<tr>
<td>$x_2$</td>
<td>$\text{std}(f[0, t_1])$</td>
</tr>
<tr>
<td>$x_3$</td>
<td>$\text{mean}(f\dot{[0, t_1]})$</td>
</tr>
<tr>
<td>$x_4$</td>
<td>$\text{std}(f\dot{[0, t_1]})$</td>
</tr>
<tr>
<td>$x_5$</td>
<td>$\text{mean}(f\dot{[t_1, t_2]})$</td>
</tr>
<tr>
<td>$x_6$</td>
<td>$\text{std}(f\dot{[t_1, t_2]})$</td>
</tr>
<tr>
<td>$x_7$</td>
<td>$\nu_1$</td>
</tr>
</tbody>
</table>

Table 3: Scores from feature selection methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>fsccmrrr</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.2669</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.3584</td>
<td>0.0000</td>
</tr>
</tbody>
</table>
Table 4: Average classification accuracy for the five feature sets across the six tested algorithms

<table>
<thead>
<tr>
<th></th>
<th>S_1</th>
<th>S_2</th>
<th>S_3</th>
<th>S_4</th>
<th>S_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>90 %</td>
<td>90 %</td>
<td>80 %</td>
<td>83 %</td>
<td>91 %</td>
</tr>
</tbody>
</table>