

Advanced Sound Identification Classifiers Using a Grid Search Algorithm for Accurate Audio-based Construction Progress Monitoring

Yong-Cheol Lee¹

Michele Scarpiniti² and Aurelio Uncini²

¹ Department of Construction Management, Louisiana State University, Baton Rouge, LA. Email: yclee@lsu.edu

² Department of Information Engineering, Electronics and Telecommunications (DIET), "Sapienza" University of Rome, Rome, Italy

ABSTRACT

Sounds of work activities and equipment operations at a construction site provide critical information regarding construction progress, task performance, and safety issues. The construction industry, however, has not investigated sound data and their applications, which would offer an advanced approach to the unmanned management and remote monitoring of construction processes and activities. For analyzing sounds emanating from construction work activities and equipment operations, which generally have complex characteristics that entail overlapping construction and environmental noise, a highly accurate sound classifier is imperative for data analysis. To establish the robust foundation for sound recognition, analysis, and monitoring frameworks, this research study examines diverse classifiers and selects those that accurately identify construction sounds. Employing nine types of sounds from about 100 sound data originating from construction work activities, we assess the accuracy of seventeen classifiers and find that one is able to classify sounds with 93.16% accuracy. A comparison with some recent Deep Learning approaches have been also provided, obtaining results similar to the best ones of the traditional machine learning methods. Participants can use the classifier on construction projects to enhance the processes of construction monitoring, performance evaluation, decision making, and safety surveillance.

Keywords: Sound classifier, feature extraction, construction monitoring, machine learning.

INTRODUCTION

The complex structure and the increasing number of requirements of construction projects have pressured project managers to seek advanced ways to automate management activities and monitor construction work activities and equipment operations remotely. The steady and real-time monitoring of construction processes and tasks in the field reduces one of the most salient risks in a construction project's uncertainty. Several dynamic factors on a construction site involving project participants, weather conditions, supply chains, and safety hazards can complicate the control of associated uncertainties. In an effort to reduce such uncertainty, several studies have found evidence that the systematic monitoring of a construction project benefits project managers by providing construction field information in a timely manner and enabling them to identify urgent issues and promptly respond to unexpected problems (Bosché 2010; Golparvar-Fard et al. 2015). In addition, Navon in (Navon 2005) found that project managers can minimize schedule and cost overruns by taking prompt action to overcome obstacles resulting from discrepancies between designed and actual progress, processes, and tasks. Another study (McAfee et al. 2012) suggested that timely and accurately collected construction field data allow domain experts and project participants to monitor and understand construction field activities and issues, minimizing the "not invented here" syndrome, and they form the basis of decision making with respect to project and safety management. The construction industry can derive these benefits by establishing robust and accurate construction field monitoring.

The establishment of a sound and accurate monitoring system requires a considerable amount of information about the behavior of construction workers and equipment operations on a construction site. Such information should reflect construction field conditions, project processes, worker and equipment performance, and any emergency situations pertaining to construction safety and hazards. To improve construction process management of field activities, several previous studies have investigated diverse construction field data (e.g., visual information). Few, however, have investigated the sounds emanating from construction work activities to determine their underlying benefits. Each task and process of construction field activities emits a unique sound that indicates worker behaviors, materials, devices/equipment types, environmental situations, construction processes, and other

relevant factors. Thus, an in-depth understanding and analysis of construction field sounds should provide project managers and other project participants with insightful field information to which they can refer to intuitively determine site conditions and remotely govern onsite processes, labor, and equipment.

Such a purpose can be fulfilled by audio-based construction site monitoring, a promising method and a supportive resource for unmanned field monitoring and safety surveillance that leverages construction project management and decision making. Since sounds generally consist of various features extracted from recorded signals, classifiers that can accurately recognize and understand feature characteristics are pivotal parts of sound identification. To accurately recognize and categorize sounds of construction work activities and equipment operations, with their unique characteristics and complex task information, we can use a highly accurate sound classifier that explicitly informs site conditions and task processes. To the best of our knowledge, however, no study has attempted to ascertain the most suitable classifier for analyzing sounds emanating from construction sites. To fulfill this objective, we intend to use audio-based field monitoring to conduct an in-depth investigation and analysis of the performance of multiple sound classifiers. To identify the accuracy of sound classifiers, we investigate seventeen classifiers and analyze their performance according to nine types of sounds from 97 files data originating from construction work activities and equipment operations. The sound identification classifiers with the highest accuracy will form a sound classification core for the establishment of a robust, audio-based field monitoring framework, that significantly improves the processes of construction monitoring, performance evaluation, decision making, and safety surveillance.

LITERATURE REVIEW

Traditional approaches to the manual collection of on-site work data and remote construction project monitoring are time-consuming, inaccurate, costly, and labor intensive (Davidson and Skibniewski 1995; Navon and Sacks 2007). With the evolution of information technology, the construction industry is seeking state-of-art field data collection and analysis methods that enhance construction project

monitoring and robust field management (Cho et al. 2017). The growing demand for improving real-time field data collection and site monitoring has led to a paradigm shift in new intelligent construction management. Various field data collection methods have been studied and implemented in construction project management. In addition, field data acquisition techniques using GPS, ultra-wide band (UWB), and sensors have been introduced in the construction industry (Cheng and Teizer 2013; Teizer et al. 2007).

Several recent studies (Dimitrov and Golparvar-Fard 2014; Golparvar-Fard et al. 2015; Seo et al. 2015) have used construction field images such as daily construction photography to explore automatic image-based progress monitoring. With flexible vision systems, unmanned aerial vehicles have also been examined as effective tools for construction monitoring, tracking materials, and improving safety (Hubbard et al. 2015; Liu et al. 2014; Siebert and Teizer 2014). Researchers have used imaging processing techniques to interpret images and videos collected from construction sites. However, vision-based field monitoring, compared to audio-based monitoring, generally requires considerable data processing. In addition, the acquisition of image or video data requires a certain level of illumination for daytime vision-based monitoring (Cho et al. 2017). Therefore, as audio-based field monitoring is not as limited with regard to data-processing capabilities and data-collection requirements as vision-based monitoring, it reduces both the time and cost of on-site monitoring.

Sound identification studies involving signal processing and audio classification have focused primarily on four main areas: signal analysis, feature extraction, model training, and model testing (Gaikwad et al. 2010). The most common classifiers – K-nearest neighbor, the Gaussian mixture model, the hidden Markov model, artificial neural networks, deep neural networks (that are neural network architectures with many layers), and support vector machines – have been fully developed and implemented with satisfactory performance and reliability (Gencoglu et al. 2014; Sharan and Moir 2016). Studies of necessary hardware and software requirements have proven the feasibility of monitoring systems using commercially available devices and software (Cheng et al. 2017). Other studies (Cheng et al. 2016; Cheng et al. 2017; Cho et al. 2017) have applied various algorithms such as

the support vector machine (SVM) and the Hidden Markov model (HMM) to test and evaluate the audio-based classification of the activity types related to construction operational equipment.

The automatic monitoring of an environment by means of collected audio signals falls under the broader field of Computational Auditory Scene Analysis (CASA) (Wang and Brown 2006), the aim of which is to successfully analyze a stream of continuous audio to identify and isolate sources of interest that it contains. One can acquire audio by using either (i) microphone arrays (Silverman and Flanagan 1998), or (ii) distributed sensors (Sallai et al. 2011). Subsequent sound is typically analyzed by applying state-of-the-art machine learning techniques (Alpaydin 2014; Witten et al. 2017) that recognize the presence of specific sources. The task of automatically labeling a given audio signal in a set of predefined classes is referred to as Automatic Audio Classification (AAC) (Fu et al. 2011). Specifically, the ACC system works by windowing the audio signal in small and overlapped frames, extracting some meaningful statistical features, and classifying them by a standard machine learning tool (Cheng et al. 2017).

AAC has been studied primarily in the context of single-level applications, where the classes are restricted to a very specific domain. A complex, realistic classification system must be highly modular, flexible, and hierarchical, topics that was only been marginally discussed in the literature until the last decade (Senator 2005). Unfortunately, only a small number of studies have taken this direction of research. For example, one study (Atrey et al. 2006) presented a four-level system for event detection, based only on a single sensor to gather information for a binary classification tasks. The authors of (Abu-El-Quran 2006) and (Zhao et al. 2010) detailed two systems that, starting from a microphone array, perform simultaneous speech and non-speech recognition. Although their work bears some resemblance to the system used in this paper, they were primarily meant for use indoors and performed non-speech classification of domestic sounds for living environment surveillance. One study (Zhou et al. 2009) described an early application of multi-stage classification to an audio stream of data and another (Scardapane et al. 2015) applied it in a real-world scenario.

In this paper, we analyze the behavior of seventeen different multi-level classifiers and compare their accuracy, false positive counts, and confusion matrices. The classifiers are well known machine-

learning techniques. They include Bayesian and naïve Bayesian networks (John and Langley 1995), the Hoeffding tree (Hulten et al. 2001), the decision table (Kohavi 1995), the decision tree (Quinlan 1993), the random tree (Rokach and Maimon 2014) and the random forest (Breiman 2001), the multilayer perceptron (MLP) (Haykin 2009), sequential minimum optimization (SMO) (Plat 1998), k-nearest neighborhood (Altman 1992; Aha et al. 1991) and PART decision (Frank et al. 2003), locally weighted learning (LWL) (Frank et al. 2003), linear logistic regression (Sumner et al. 2005), random sub space (Ho 1998) and K* learner (Cleary and Trigg 1995), the 1R classifier (Holte 1993) and the Cohen version of the IRAP classifier (Cohen 1995). Additional details on the types of classifiers can be found in Section 3 and Table 2. To provide a fast and easy representation of each approach, we conducted the comparisons in the WEKA software environment (Frank et al. 2016; Witten et al. 2017).

In the literature, it is possible to find several instances of successful applications in the field of environmental sound classification that make use of Deep Learning (DL) techniques (Goodfellow et al. 2016; Aggarwal 2018). For example, in the work of (Piczak 2015), the author exploits a 2-layered CNN working on the spectrogram of the data to perform ESC, reaching an average accuracy of 70% over different datasets. Other approaches, instead of using handcrafted features such as the spectrogram, perform end-to-end environmental sound classification obtaining higher results with respect to the previous ones (Tokozume and Harada 2017). The MelNet architecture described by (Li et al. 2018) has been proven to be remarkably effective in environmental sound classification. This architecture uses a combination of two Deep Convolutional Neural Networks (DCNNs) to classify environmental sound data (like rain, dogs, cats, engines, trains, airplanes, etc.). A similar approach is exploited in the paper of (Maccagno et al. 2019) whose aim is to develop an application able to recognize vehicles and tools used in construction sites, and classify them in terms of type and brand. This task has been tackled with a neural network approach, involving the use of a DCNN, which will be fed with the mel spectrogram of the audio source as input. However, the classification problem presented in this paper is limited to only five classes extracted from audio files collected in several construction sites, containing in situ recordings of multiple vehicles and tools. Finally, (Scarpiniti et al. 2020) propose a Deep Recurrent Neural Network (DRNN) approach, based on LSTM units (Aggarwal 2018) for the classification of real-

world data recorded in construction sites. Both the last methods provide very high accuracy in the classification of audio data.

Motivated by these results of DL approaches, in this paper we also evaluate three different end-to-end methods. Specifically, we will implement: i) a DCNN working on the spectrogram of audio files; ii) a DRNN working on the raw audio data; and, iii) a Deep Belief Network (DBN) (Goodfellow et al. 2016; Aggarwal 2028) that uses the raw data.

The targeting of all the proposed classifiers has been made by using a collection of web data in order to work with clean and clear sounds. However, the validation of the proposed approach has been demonstrated also on the real-world data recorded in three different scenarios related to a bridge construction project.

DATA FOR CONSTRUCTION SOUNDS

To accurately classify construction work sounds and consistently establish a reliable audio-based system, one must collect a broad array of sound data. Such a task, however, is challenging and time-consuming, for it requires manually recording the types of construction work sounds and assembling a huge sound dataset for the construction industry. To address this problem, we have collected 97 sound files of construction work activities and equipment operations from Web resources such as video and audio files that do not involve copyright issues. To obtain sounds of real construction work, we also collected sound data from two construction sites, hospital and industrial construction project sites, in Louisiana and categorized them into nine types that represent “target classes” to which the classifiers assign each item. We analyzed all of the sound data by the seventeen classifiers and plan to share this information with the public through a data archive that the authors have established. After removing the silence and noisy segments from the analyzed files and extracted the 62 features described in Section 4, a total of 64,696 instances have been found. An instance is the set of the 63 features provided as an example to the classifier. Table 1 lists the number of files and details about each sound item. All these files have been split in a train and a test set in such a way that about the 75% of instances are used for training and the remaining 25% for testing the considered classifiers. Table 2 shows the details in terms of duration and number of instances in each class for the train and test sets.

Classifiers types

Among diverse sound classifiers, this paper involves the performance comparison of broadly used seventeen classifiers listed in Table 3. The name of each classifier reported in Table 3 is that adopted in the WEKA software. The table also lists the seminal reference for each classifier.

This section includes a brief description of the employed classifiers.

A *Bayesian Classifier* (BayesNet) is a network that minimizes the probability of misclassification (John and Langley 1995) and maximizes the class probability conditioned on input instances. The problem with Bayesian classifiers is the evaluation of probability densities. Therefore, one study (John and Langley 1995) introduced a family of simple probabilistic classifiers based on the application of Bayes' theorem with strong (naive) independence assumptions between the features. This kind of a classifier is called *Naïve Bayesian Classifier* (NaiveBayes). A simple but good classifier is based on the application of *linear logistic regression* models (SimpleLogistic) (Sumner et al. 2005). Another classification technique is the use of a back-propagation algorithm in a multilayer perceptron with a single hidden layer (Haykin 2009). This classifier generally provides high-performance sound classification and recognition. Another classifier introduced in (Plat 1998), is the *sequential minimal optimization* (SMO), which trains a support vector classifier, normalizes all attributes, and discards all missing values.

The authors of this work also provided comparisons with classification methods based on trees. In particular, *random tree* (RandomTree) is a simple algorithm based on the construction of a tree. The algorithm considers K randomly chosen attributes at each node (Rokach and Maimon 2014) but involves no node pruning. A *Hoeffding tree* (HoeffdingTree) is an incremental and anytime decision tree that exploits the fact that a small sample is often sufficient for choosing an optimal splitting attribute (Hulten et al. 2001). It implicitly assumes that the distribution of examples does not change over time. Another method, *random sub-spaces* (RandomSubSpace), constructs multiple trees by choosing random subspaces (Ho 1998). A *random forest* (RandomForest), which is simply an ensemble of random tree classifiers, constructs a multitude of decision trees (Breiman 2001). A particular algorithm used to construct a decision tree is the *C4.5 algorithm* proposed by Quinlan in (Quinlan 1993) and referred to

in WEKA as J48. One of the simplest but significantly efficient technique, based on the *K-nearest neighbours* (kNN) algorithm and referred to in WEKA as IBk (Altman 1992; Aha et al. 1991), classifies an instance by a majority vote of its neighbors and assigned an object to the class most common among its k nearest neighbors. The K^* classifier (KStar), is an instance-based classifier that uses an entropy-based distance function (Cleary and Trigg 1995). KStar evaluates the class of a test instance based on the class of training instances that perform in a similar way. Another instance-based classifier is *Locally weighted learning* (LWL), which uses an ensemble of instance-based classifiers (Frank et al. 2003).

A very simple but relatively inaccurate classifier is the *1R* algorithm (OneR), which selects only one rule that produces the smallest number of errors (Holte 1993). It typically obtains a rule by constructing a frequency table. Another approach is to use a *decision list* (PART), constructed from a partial decision tree obtained by the C4.5 algorithm (Frank and Witten 1998). An alternative classification method is using a proportional rule learner similar to that in the *Repeated Incremental Pruning to Produce Error Reduction* (RIPPER) algorithm referred to in WEKA as JRip, an optimized version of the IREP algorithm proposed by Cohen in (Cohen 1995). Finally, classification can be performed by building and using a simple decision *table majority classifier* (DecisionTable) (Kohavi 1995).

Deep Learning classifiers

Regarding the Deep Learning approaches, in this paper we consider three different architectures widely used in literature.

The first approach is based on a Deep Convolution Neural Network (DCNN). CNNs are a particular type of neural networks, which use the convolution operation in one or more layers for the learning process. These networks are inspired by the primal visual system, and are therefore extensively used with image and video inputs (Goodfellow et al. 2016). A DCNN is composed by the sequential cascade of three main layers: i) a convolution layer; ii) a detector layer; and, iii) a pooling layer. A final fully connected layer and a soft-max one terminate the architecture.

The second approach adopts a Deep Recurrent Neural Network (DRNN) (Goodfellow et al. 2016) that exploits the intrinsic temporal structure of audio data. This allows to exhibit a temporal dynamic

behavior that can be helpful for audio data. Since the training of DRNNs suffers from the problem of vanishing gradient, the long short-term memory (LSTM) units have been introduced (Aggarwal 2018). LSTMs allow to obtain very high accuracy in classification of audio data.

Finally, the third approach is based on the Deep Belief Network (DBN) (Goodfellow et al. 2016; Aggarwal 2018), that is a generative model with several layers of latent variable. The latent variables are binary while the visible ones, in the case of audio, are real-valued. Every unit in each layer is connected to every unit in adjacent layers, but there are no interlayer connections. Although not commonly used for audio classification, DBNs can provide good results.

SOUND CLASSIFICATION AND CLASSIFIERS TYPES

A multiclass sound classifier executes the process by which it automatically assigns an individual sound item to one of a number of trailed categories or classes according to its analyzed characteristics. The classification is performed on the base of some features extracted from the recorded audio signals after the removing of silent segments. The scheme of the overall algorithm is shown in Fig. 1.

Feature extraction and pre-processing

Each signal is segmented in a certain number of superframes of 200 ms with an overlap of 50%. In order to detect the presence of silence, each superframe is on turn segmented in smaller frames of 50 ms with no overlap. If the Root Mean Square (RMS) (Fu et al. 2011) of a frame is under the threshold of -30 dB, the frame is discarded as silence. The superframe is then reconstructed from the remaining constituent frames. Subsequently, suitable features are extracted.

Many audio features have been proposed in the literature for audio signal classification. A total of 62 features, grouped into 15 distinct sets of features and exploiting both time and frequency characteristics, have been extracted from the audio signals (Patsis and Verhelst 2008; Lu et al. 2002). Each feature is obtained by segmenting again the superframe in smaller frames as shown in Fig. 2 and evaluating the needed values. Each feature uses a frame of different length and overlap.

The symbols, number and names of these sets, along with each window duration and its overlap, are shown in Table 4.

The descriptors have been chosen so as to efficiently represent different domains of the audio signal: characteristics such as the ZCC or the RMS are extracted in the time domain, whilst VSFLUX or SBC pertain to the frequency domain. The reason is that the utility of each descriptor changes depending on the nature of the task. A brief description of the employed features is as follows.

The *Zero Crossing Count* (ZCC) is defined as the number of times the signal reverses its sign. The values of the ZCC over the subframes are subdivided into 10 bins, and the normalized frequencies of these bins are used as features for the superframe, together with the variance of the ZCC values (denoted as VZCR). Another feature extracted here is the *High Order ZCC Ratio* (HZCRR), defined as the ratio of the number of frames whose ZCC is above 1.5 times the average ZCC of the superframe.

The RMS of each superframe is also computed. As before, the RMS values are subdivided into 10 bins, the bins are normalized, and their relative frequencies are used as features. Additionally, we compute the ratio of *Low Energy Frames* (LEF), i.e. the ratio of frames whose RMS is lower than 50 % the average RMS for the superframe. The mean value of a similar feature, the *Short Time Energy* (STE) (Fu et al. 2011), is also computed. Then, we extract the *Low Short Time Energy Ratio* (LSTER), defined as the ratio of the number of frames whose STE is less than half the average STE of the superframe.

For the following features the superframe is convolved with a low-pass filter to retain only the frequencies between 1.5 kHz and 1.6 kHz. Both the low-band superframe and the original superframe are then segmented. The *Low-Band Energy Ratio* (VLER) is defined as the variance of the ratio of the low-band STE to the whole band STE. Next, the variance of the *Spectral Flux* (VSFLUX) is computed, where the spectral flux is defined as the 2-norm between the frequency spectra of the current frame and of the previous frame. An additional feature computed here is the *Low Frequency RMS* (LFRMS), where the LFRMS is defined as the median of the RMS of the low-band version of the frame. The *Low Frequency ZCC* (LFZCC) is computed as the 9-th moment of the ZCC values of the frames.

Additionally, we compute the *Sub-Band Correlation* (SBC) (McAuley et al. 2005) for 4 different filters of increasing frequencies, and the *High-Order Crossing* (HOC) (Petrantonakis and Hadjileontiadis 2010)

up to the 4-th order. Moreover, 7 *Linear Predictive Coefficients* (LPC) have also been evaluated (Makhoul 1976): these coefficients can be thought as the parameter of an AR system that generates the signal when it is fed with a white Gaussian noise.

Finally, we extract also 13 *Mel-Frequency Cepstrum Coefficients* (MFCC) (Zheng et al. 2001), that are a representation of the short-term power spectrum of the signal, based on the linear cosine transform of the log power spectrum on a nonlinear mel scale of frequency.

Deep Learning methods, instead, use an end-to-end (E2E) approach (Aggarwal 2018) in the sense that the final classification is obtained by using directly raw audio data (samples) as input to the architecture. Only the approach based on the DCNN use the time-frequency representation (spectrogram) as input.

METHODOLOGY OF THE ACCURACY ANALYSIS

To provide a useful and accurate interpretation of the seventeen classifiers, we have used several measurement techniques. One is counting the number of items correctly and locating incorrectly classified items (Witten et al. 2017). While the word “positive” (P) is generally used to describe an event that has been identified, the term “negative” (N) describes an event that has been rejected. With this terminology, let us define the following: i) “true positive”, when an instance is correctly classified in its own class; ii) “true negative”, when an instance that does not belong in a class is correctly classified as not belonging in that class; iii) “false positive”, when an instance that does not belong in a class is incorrectly classified as belonging from that class; and, iv) “false negative”, when an instance that belongs from a class is incorrectly classified as not belonging from that class.

By using these simple indicators, it is possible to construct a number of useful indexes in order to establish the effectiveness of a classifier. In this paper, we use accuracy, precision, recall, F-measure, MCC (Matthews correlation coefficient) and AUC (area under the curve). These evaluation criteria are well known in literature and a formal definition can be found in (Witten et al. 2017). The AUC is the area

under the Receiver Operating Characteristic (ROC) (Witten et al. 2017), which is a measure of the accuracy of a classifier. An AUC close to one indicates a good accuracy in classification.

All these criteria are evaluated as a binary classification in the sense that for each j -th class (among the N available), we have considered the j -th class as the “true positive” and the remaining $(N - 1)$ classes as the “false positive”. Hence, for each j -class the specific criteria has been computed and finally a weighted mean of the obtained criteria values with respect the number of instances in each class has been evaluated.

In addition, another efficient instrument for evaluating the accuracy of a multiclass classifier is the confusion matrix, which shows how single instances have been classified.

Deep Learning approaches have been evaluated only in terms of single class accuracy and overall accuracy.

RESULTS OF ACCURACY PERFORMANCE ANALYSES

To test the accuracy of the compared multiclass classifiers, the feature extraction procedure has been implemented in a MATLAB environment, while WEKA software has been used to perform the classification stage. To train the classifiers, we used the previous collection of sounds. In particular, all the sounds were resampled at 16 kHz, and a total of about 5 hours of sound have been considered for training, while about 2 hours for testing. Hence, features were extracted according to the algorithm previously described, obtaining the 49,361 instances of Table 2. After the training of the considered classifiers, these have been evaluated on the 15,335 instances of the test set.

All the hyper-parameters used in the tested classifiers have been fine tuned by using a grid search algorithm in suitable intervals, in order to improve the obtained accuracy. The main hyper-parameters of all the classifiers are summarized in the following Table 5. The name of the used hyper-parameters is that used in the WEKA software and their meaning can be found in (Witten et al. 2017). All the other parameters have been set to their default values.

Results in terms of accuracy, precision, recall, F-measure, MCC and AUC (evaluated as the weighted average over all the classes), are shown in Table 6.

Results summarized in Table 6 show that the analyzed classifiers behave in different way on the used data set. Some classifiers, like the LWL and the OneR, provide very poor results with an accuracy lower than 40%. Just the opposite, other classifiers behave very well. Specifically, the best six classifiers, listed in order of best accuracy, are: RandomForest (93.16%), MLP (91.06%), IBk (85.28%), PART (83.66%), RandomSubspace (81.81%) and KStar (80.37%). The rest of the classifiers behave in an intermediate way, providing results in the range 56 – 80%. The same considerations done for the accuracy metric can be done for the other implemented indexes. In fact, precision, recall, F-Measure, MCC and the AUC confirm the same ranking of the analyzed classifiers. Fig. 3 explicitly illustrates the comparison of the analyzed accuracy performances of all classifiers.

From the above discussion, we can argue that some of the six best-performing classifiers can be used to solve the task of sound identification in construction site process monitoring. However, additional studies and experimental results, maybe using real-world data recorded in huge and noisy construction sites, are need to identify one or two target classifiers to be used in the proposed task.

In order to better highlight the classification capabilities of the proposed classifiers, we show the confusion matrix of some the best ones. Specifically, Table 7 shows the confusing matrix of the Random Forest, while Table 8 shows the confusing matrix of the MLP classifier. The confusion matrices of kNN (IBk) and PART are shown in Tables 9 and 10, respectively.

A careful observation of these tables shows that all classes are usually well balanced, with a similar number of examples for each class. The only slightly unbalanced classes are the bulldozer (class 03), truck (class 05) and the concrete mix (class 07). However, this data unbalancement is not a problem, since the reached accuracy is high also for the unbalanced classes.

Table 7 also underlines the high accuracy obtained by the Random Forest (93.16%), since the confusion matrix shows high numbers into the diagonal entries while the rest of values are very small, often close to zero. On the contrary, Tables 8 – 10 show some slightly greater values on the off-diagonal

entries. Interestingly enough, these Tables also show that the class 01 (concrete breaking) is always the class that presents the best accuracy.

Moreover, an examination of Tables 7 – 10 highlights that the most misclassified items are certainly those related to the unbalanced classes 05 and 07 (truck and concrete mix). However, there exist other classes with a certain degree of misclassification, but the particular class showing this problem depends on the used classifier. For example, the Random Forest produces a 12% of misclassified items in class 04 (piling) and 08 (concrete grinding), while PART obtains a 20% of misclassification in class 09 (drilling) and a 30% in class 08 (concrete grinding). Similarly, IBk produces a 18% of misclassified items in the class 07 (concrete mix) and in the same 08 (concrete grinding). Obviously, the misclassification errors increase when the accuracy of a classifier decreases, and the number of classes with high misclassification rates increases. As an example, Table 11 shows the confusion matrix of the J48 which performs with an accuracy of 70.24%. Although the misclassification error is about 30%, some classes provide high classification errors, such as a 52% for class 08 (concrete grinding), a 32% for class 09 (drilling) and a 50% of class 04 (piling). In addition, an observation of Table 11 highlights that the off-diagonal terms assume higher values in the bottom left region of the matrix with respect to the other regions.

A final observation is the following: many of the classifiers with an intermediate accuracy tend to classify item from class 02 (grounding excavator) as items belonging from classes 06 (grading), and vice versa; items from class 09 (drilling) as items belonging again from class 06 (grading) and items from class 03 (bulldozer) as items belonging from class 08 (concrete grinding).

Deep Learning Approaches

In order to test the accuracy of the considered Deep Learning classifiers, we have implemented the three architectures in Python by using the facilities of Keras open-source API library. The dataset is the same used for evaluating the traditional machine learning approaches.

The main hyper-parameters have been set by a grid search and are summarized as follows. A complete set of hyper-parameters can be found in (Maccagno et al. 2019; Scarpiniti et al. 2020). For the DCNN we have used 5 convolutional layers, followed by a dense layer and a soft-max layer as

output. All layers use the ReLU activation function. A batch size of 64 has been used, while in the dense layer a dropout with rate 0.3 is used. The loss function is the cross-entropy and the Adam optimizer with a learning rate of 0.0005 has been chosen. A total of 100 training epoch has been run. For the DRNN we have used 3 layers, the first two with LSTMs. the first LSTM layer have an output size of 128 and return only the output of the last hidden state, the second LSTM layer have an output size of 32 and return the full sequence of hidden states output. A frame length of 50 ms has been used. The loss function is the cross-entropy and the Adam optimizer with a learning rate of 0.0005 has been chosen. A total of 100 training epoch has been run. For the DBN we have used 4 hidden layers plus an output soft-max layer. The hidden layer used a Gaussian activation function with a batch size of 32. The training is based on the contrastive divergence algorithm (Goodfellow et al. 2016) and the Adam optimizer has been used once again. A total of 240 training epochs has been run.

The per-class accuracy and the overall accuracy, obtained as the weighted mean across the different classes, of the three considered DL approaches are shown in Table 12. This table show that all the considered deep learning methods are able to obtain quite high accuracies. Specifically, DCNNs and DRNNs behave very similar, while DBNs show the lowest accuracy. Form a comparison of the single accuracies on each class, we can see that the approach based on the DCNN provide the best results. However, we have to underline that the class 5 (Truck) always presents an accuracy considerably lower than the other classes and tends to degrade the overall accuracy. This behavior can be explained by considering that, as shown in Table 2, this class show a limited number of audio data. As it is known, deep learning techniques overcomes the traditional ML ones only if the training is made over a huge set of data. In this regards, numerical results in (Maccagno et al. 2019; Scarpiniti et al. 2020) performed on a larger dataset confirm the superior performance of DL approaches over all classes.

CASE STUDIES

To evaluate the performance of the proposed sound classifiers, this study involves the case study, which employed sound data of a real construction project. This research team visited the off-system bridge construction project, Tucker Road Bridge over Drainage Bayou located in Louisiana and recorded

the sounds of excavating the ground, erecting and driving the piles for the bridge for several hours. The Fig. 4 shows the sound recording of erecting and driving the bridge pile in the construction site. The type of this construction project is the removal of existing timber and concrete bridges, installation of concrete slab span bridges, grading, base course, and asphalt concrete surfacing. Transportation construction such as roadway and bridge construction is one most challenging job among all type of construction activities around the globe. This kind of construction is basically involved with ample number of actions where mostly different equipment and machines used to make it accurate, the activity faster and less time consuming. Although these are quite significant to materialize the construction conspicuously, but they also can create massive sound at construction site which is detrimental for human body who are working over there.

In order to provide a fair comparison of sound classifiers on real world recorded data, the set of 62 features have been evaluated on the whole data and a total of 5,000 instances have been randomly selected to be used as the new test set. Table 13 shows the accuracy obtained for the considered case studies by four of the best performing classifiers, namely: Random Forest, Random Subspace, Multilayer Perceptron and IBk.

From Table 13, we can argue that the selected four classifiers (Random Forest, Random Subspace, Multilayer Perceptron and IBk) are able to provide an acceptable accuracy even in the case of sounds recorded in real world scenarios. Specifically, results are close or greater to 70% (excepting for the IBk that performs poorly in the Backhoe scenario), with a peak close to 86%. Only in the case of 'Driving the pile' the obtained accuracy is slightly smaller, around 68% for the Random Forest and Random Subspace and 62-63% for the MLP and IBk. Note that, differently from results provided in Section 6 for an hypothetical noise-free scenario, results shown in Table 13 refer to sounds recorded along the unavoidable background noise, due to the microphones themselves, the recording system and the other related activities in the construction site. From this consideration and the obtained results, we are confident that the proposed approach can represent a good technique to detect events in construction sites.

Moreover, the sound identification approach can capture each work activities as well as a task cycle such as pile lifting, erecting, driving, and releasing. Thus, a project manager or an owner who is in or out of site can not only directly recognize the type of current work activity or idle time, but also accurately expect a next step. As shown in Fig. 5, once this system identifies one cycle of Bent 5 Piles including pile lifting, erecting, driving, and releasing, project participants or project managers can expect and prepare the start of the Bent 6 Piles' task. Since we can easily calculate a duration of each task, an analysis of work performance and a prediction of a project schedule can be executed according to a sound-based analysis of previous work activities. A construction project such as this bridge site located in the suburban area of Louisiana needs an automated monitoring system to remotely oversee the work activities and equipment operations. Three supervisors had to stay in the bridge construction site to supervise the site and report its progress. The authors expect that this sound-based monitoring method can reduce significant time and effort to remotely govern a construction project. In particular, each State has to manage a large number of construction projects. Thus, this approach is expected to allow state or federal officers to remotely monitor construction projects and make an accurate log of work activities of each project.

DISCUSS AND PLANS FOR THE APPLICATION OF AN ACCURATE SOUND CLASSIFIER

An accurate sound recognition system capable of classifying the unique characteristics of sounds generated by workers and equipment on a construction site will improve the processes of monitoring work progress, evaluating task performance, and surveilling safety. With such a system, project managers will be able to monitor the status of workers remotely, investigate the effective distribution of hours, and detect issues of safety in a timely manner. With a better understanding of on-site situations, domain experts should be able to make better data-driven decisions, optimize labor arrangements, and provide a higher standard of safety management. To achieve these goals, we tested, compared, and evaluated the implementation of sound classifiers on both web-based and real-world data. To adapt the results of our evaluation, which identifies the most appropriate classifier for

analyzing construction work sounds, we will perform an audio-based performance evaluation of construction activities and equipment operations.

Working hours of construction laborers can be divided into idle hours and effective hours. The number of effective working hours can be captured to evaluate work/equipment operational performance and plan an optimized construction procedure. However, as calculating and logging the hours of each worker and equipment are manual processes and thus labor intensive and infeasible, the next study will determine features of sound detection that will help project managers identify actual work hours and monitor work activities on a construction site. We will collect data from a real construction project on site and wirelessly transmit the data to a server in real-time for an analysis of construction site conditions and the identification of the process status. We will carry out classification tasks via trained models with desired accuracy. The results of classification, combined with location information integrated by Building Information Modeling (BIM), will be available to project managers.

At this time, the main limitations of the proposed approach are two-fold. On one hand, the performance of a sound classifier rapidly degrades when the signal to noise ratio (SNR) becomes failing (i.e., lower than -10 dB). On the other hand, these traditional sound classifiers struggle to take a decision when more than one sound event is overlapped in the same time interval. Both these limitations can be overcome by using microphone arrays with high directivity, which is a set of microphones that focus their “attention” towards a specific spatial direction. Possible stages of wideband noise reduction and acoustic blind source separation pre-processing techniques can also be applied in order to limit the accuracy degradation due to these drawbacks. Another power solution consists in applying other novel techniques provided by the Deep Learning approach (Salamon and Bello 2017), towards which our future research will be addressed.

CONCLUSIONS

Audio-based field monitoring has significant potential to overcome current challenges, such as monitoring angle, time, and data processing efficiency, inherent in manual and vision-based monitoring

frameworks. The success of a monitoring technique depends on the selection of a suitable analytics algorithm. Thus, to facilitate the task of construction site activity monitoring, we analyzed about 64,700 instances of construction site sound signals and found that the Random Forest classifier achieved 93.16% accuracy, sufficient for constructing a sound-based construction site monitoring system and performance checking framework that project participants can use to enhance their monitoring of work progress, labor efficiency, and safety. Even when tested on real-world data recorded in a construction site this classifier has shown an accuracy up to 85%. Moreover, a comparison with three Deep Learning approaches has provided results with accuracies in the range of 90%-94%. By providing project managers with a more comprehensive understanding of on-site situations, the sound-based monitoring system should improve the data-driven decisions of domain experts that will enhance work activity forecasts and productivity and enable them to respond quickly to problems that arise on construction sites. However, sounds emanating from a construction site, collected by a sensor, typically overlap, which creates a challenging task that must be investigated by future studies.

REFERENCES

- Abu-El-Quran, A. (2006). "Security monitoring using microphone arrays and audio classification." *IEEE Transactions on Instrumentation and Measurement*, 55(4), 1025–1032.
- Aggarwal, C. C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer.
- Aha, D. W., Kibler, D., and Albert, M. K. (1991). "Instance-based learning algorithms." *Machine Learning*, 6(1), 37–66.
- Alpaydin, E. (2014). *Introduction to machine learning*. MIT press, 3 edition.
- Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression." *The American Statistician*, 46(3), 175–185.
- Atrey, P. K., Maddage, N. C., and Kankanhalli, M. S. (2006). "Audio based event detection for multimedia surveillance." *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2006)*, Vol. 5, 1–5.

- Bosché, F. (2010). "Automated recognition of 3D CAD model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction." *Advanced Engineering Informatics*, 24(1), 107–118.
- Breiman, L. (2001). "Random forests." *Machine Learning*, 45(1), 5–32.
- Cheng, C.-F., Abbas, R., Davenport, M. A., and Anderson, D. (2016). "Audio signal processing for activity recognition for construction heavy equipment." *Proceedings of the 33rd International Symposium on Automation and Robotics in Construction (ISARC 2016)*, 642–650.
- Cheng, C.-F., Rashidi, A., Davenport, M. A., and Anderson, D. V. (2017). "Activity analysis of construction equipment using audio signals and support vector machines." *Automation in Construction*, 81, 240–253.
- Cheng, T. and Teizer, J. (2013). "Real-time resource location data collection and visualization technology for construction safety and activity monitoring applications." *Automation in Construction*, 34, 3–15.
- Cho, C., Lee, Y.-C., and Zhang, T. (2017). "Sound recognition techniques for multi-layered construction activities and events." *Computing in Civil Engineering 2017*, Vol. 2017, ASCE, 326–334.
- Cleary, J. G. and Trigg, L. E. (1995). "K*: An instance-based learner using an entropic distance measure." *12th International Conference on Machine Learning*, 108–114.
- Cohen, W. W. (1995). "Fast effective rule induction." *Twelfth International Conference on Machine Learning*, 115–123.
- Davidson, I. N. and Skibniewski, M. J. (1995). "Simulation of automated data collection in buildings." *Journal of Computing in Civil Engineering*, 9(1), 9–20.
- Dimitrov, A. and Golparvar-Fard, M. (2014). "Vision-based material recognition for automated monitoring of construction progress and generating building information modeling from unordered site image collections." *Advanced Engineering Informatics*, 28(1), 37–49.
- Frank, E., Hall, M., and Pfahringer, B. (2003). "Locally weighted naïve bayes." *19th Conference in Uncertainty in Artificial Intelligence*, 249–256.
- Frank, E., Hall, M., and Witten, I. H. (2016). "The WEKA workbench." *Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*, Morgan Kaufmann, 4 edition, 553–572.

- Frank, E. and Witten, I. H. (1998). "Generating accurate rule sets without global optimization." *Fifteenth International Conference on Machine Learning*, 144–151.
- Fu, Z., Lu, G., Ting, K. M., and Zhang, D. (2011). "A survey of audio-based music classification and annotation." *IEEE Transactions on Multimedia*, 13(2), 303–319.
- Gaikwad, S. K., Gawali, B. W., and Yannawar, P. (2010). "A review on speech recognition technique." *International Journal of Computer Applications*, 10(3), 16–24.
- Gencoglu, O., Virtanen, T., and Huttunen, H. (2014). "Recognition of acoustic events using deep neural networks." *Proceedings of the 22nd European Signal Processing Conference (EUSIPCO 2014)*, 506–510 (September 1–5).
- Golparvar-Fard, M., Peña-Mora, F., and Savarese, S. (2015). "Automated progress monitoring using unordered daily construction photographs and IFC-based building information models." *Journal of Computing in Civil Engineering*, 29(1).
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. The MIT Press.
- Haykin, S. (2009). *Neural Networks and Learning Machines*. Pearson Publishing, 2nd edition.
- Ho, T. K. (1998). "The random subspace method for constructing decision forests." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8), 832–844.
- Holte, R. C. (1993). "Very simple classification rules perform well on most commonly used datasets." *Machine Learning*, 11(1), 63–91.
- Hubbard, B., Wang, H., Leasure, M., Ropp, T., Lofton, T., and Hubbard, S. (2015). "Feasibility study of UAV use for RFID material tracking on construction sites." *51st ASC Annual International Conference Proceedings*, 669–676.
- Hulten, G., Spencer, L., and Domingos, P. (2001). "Mining time-changing data streams." *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 97–106.
- John, G. H. and Langley, P. (1995). "Estimating continuous distributions in Bayesian classifiers." *Eleventh Conference on Uncertainty in Artificial Intelligence*, San Mateo, 338–345.
- Kohavi, R. (1995). "The power of decision tables." *8th European Conference on Machine Learning*, 174–189.

- Li, S., Yao, Y., Hu, J., Liu, G., Yao, X., and Hu, J. (2018). "An ensemble stacked convolutional neural network model for environmental event sound recognition," *Applied Sciences*, 8(7).
- Liu, P., Chen, A. Y., Huang, Y. N., Han, J. Y., Lai, J. S., Kang, S. C., Wu, T.-H., Wen, M.-C., and Tsai, M. H. (2014). "A review of rotorcraft unmanned aerial vehicle (UAV) developments and applications in civil engineering." *Smart Structures and Systems*, 13(6), 1065–1094.
- Lu, L., Zhang, H.-J., and Jiang, H. (2002). "Content analysis for audio classification and segmentation." *IEEE Transactions on Speech and Audio Processing*, 10(7), 504–516.
- Maccagno, A., Mastropietro, A., Mazziotta, U., Scarpiniti, M., Lee, Y.-C., Uncini, A. (2019). "A CNN Approach for Audio Classification in Construction Sites," *29th Italian Workshop on Neural Networks (WIRN 2019)*, Vietri sul Mare, Salerno, Italy.
- Makhoul, J. (1976). "Linear prediction: A tutorial review." *Proceedings of IEEE*, 63(4), 561–580.
- McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D., and Barton, D. (2012). "Big data: the management revolution." *Harvard Business Review*, 90(10), 61–67.
- McAuley, J., Ming, J., Stewart, D., and Hanna, P. (2005). "Subband correlation and robust speech recognition." *IEEE Transactions on Speech and Audio Processing*, 13(5), 956–964.
- Navon, R. (2005). "Automated project performance control of construction projects." *Automation in Construction*, 14(4), 467–476.
- Navon, R. and Sacks, R. (2007). "Assessing research issues in automated project performance control (appc)." *Automation in Construction*, 16(4), 474–484.
- Patsis, Y. and Verhelst, W. (2008). "A speech/music/silence/garbage classifier for searching and indexing broadcast news material." *9th International Workshop on Database and Expert Systems Application (DEXA2008)*, 585–589 (September).
- Petrantonakis, P. C. and Hadjileontiadis, L. J. (2010). "Emotion recognition from eeg using higher order crossings." *IEEE Transactions on Information Technology in Biomedicine*, 14(2), 186–197.
- Piczak, K. J. (2015). "Environmental sound classification with convolutional neural networks," *IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP 2015)*, 1–6.

- Plat, J. (1998). "Sequential minimal optimization: A fast algorithm for training support vector machines." *Report No. MSR-TR-98-14*, Microsoft Research.
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.
- Rokach, L. and Maimon, O. (2014). *Data mining with decision trees: theory and applications*. World Scientific Publishing Company, 2 edition.
- Salamon, J. and Bello, J. P. (2017). "Deep convolutional neural networks and data augmentation for environmental sound classification." *IEEE Signal Processing Letters*, 24(3), 279–283.
- Sallai, J., Hedgecock, W., Volgyesi, P., Nadas, A., Balogh, G., and Ledeczki, A. (2011). "Weapon classification and shooter localization using distributed multichannel acoustic sensors." *Journal of Systems Architecture*, 57(10), 869–885.
- Scardapane, S., Scarpiniti, M., Bucciarelli, M., Colone, F., Mansueto, M. V., and Parisi, R. (2015). "Microphone array based classification for security monitoring in unstructured environments." *AEÜ – International Journal of Electronics and Communications*, 69(11), 1715–1723.
- Scarpiniti, M., Comminiello, D., Uncini, A. and Lee, Y.-C. (2020). "Deep Recurrent Neural Networks for Audio Classification in Construction Sites", submitted to the *28th European Signal Processing Conference (EUSIPCO 2020)*, Amsterdam, The Netherlands.
- Senator, T. E. (2005). "Multi-stage classification." *Proc. of the 2005 IEEE International Conference on Data Mining*, Vol. 5, 386–393.
- Seo, J., Han, S., Lee, S., and Kim, H. (2015). "Computer vision techniques for construction safety and health monitoring." *Advanced Engineering Informatics*, 29(2), 239–251.
- Sharan, R. V. and Moir, T. J. (2016). "An overview of applications and advancements in automatic sound recognition." *Neurocomputing*, 200, 22–34.
- Siebert, S. and Teizer, J. (2014). "Mobile 3D mapping for surveying earthwork projects using an unmanned aerial vehicle (UAV) system." *Automation in Construction*, 41, 1–14.
- Silverman, H. F. and Patterson, W. R. and Flanagan, J. L. (1998). "The huge microphone array." *IEEE Concurrency*, 6(4), 36–46.
- Sumner, M., Frank, E., and Hall, M. (2005). "Speeding up logistic model tree induction." *9th*

- European Conference on Principles and Practice of Knowledge Discovery in Databases*, 675–683.
- Teizer, J., Lao, D., and Sofer, M. (2007). “Rapid automated monitoring of construction site activities using ultra-wideband.” *24th International Symposium on Automation and Robotics in Construction (ISARC 2007)*, Vol. 2, 23–28.
- Tokozume, Y., and Harada, T. (2017), “Learning environmental sounds with end-to-end convolutional neural network.” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, 2721-2725.
- Wang, D. and Brown, G. J. (2006). *Computational auditory scene analysis: Principles, algorithms, and applications*. Wiley-IEEE Press.
- Witten, I. H., Frank, E., Hall, M. A., and Pal, C. J. (2017). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 4 edition.
- Zhao, D., Ma, H., and Liu, L. (2010). “Event classification for living environment surveillance using audio sensor networks.” *Proc. of IEEE International Conference on Multimedia and Expo (ICME2010)*, 528–533.
- Zheng, F., Zhang, G., and Song, Z. (2001). “Comparison of different implementations of MFCC.” *Journal of Computer Science and Technology*, 16(6), 582–589.
- Zhou, N., Ser, W., Yu, Z., Yu, J., and Chen, H. (2009). “Enhanced class-dependent classification of audio signals.” *Proc. of 2009 WRI World Congress on Computer Science and Information Engineering*, Vol. 7, 100–104.

List of Tables

1	Details of different sound classes used in the paper.	27
2	Details of train and test sets.	28
3	The seventeen different classifiers compared in this paper.	29
4	Set of the 62 used features.	30
5	The main hyper-parameters used in the tested classifiers.	31
6	Accuracy and other performance metrics of the evaluated classifiers.	32
7	Confusing matrix of the Random Forest.	33
8	Confusing matrix of the MLP.	34
9	Confusing matrix of the kNN (IBk).	35
10	Confusing matrix of the PART.	36
11	Confusing matrix of the J48.	37
12	Per-class and overall accuracy of Deep Learning approaches considered for comparison.	38
13	Accuracy [%] obtained by four of the best performing classifiers in the three considered scenarios.	39

N.	Name	Description	N. of files	Duration	N. of instances
1	concrete_breaking	Breaker	15	49:21	6059
2	ground_excavating	Excavator	8	1:43:06	15598
3	bulldozer	Bulldozer	9	38:24	2450
4	piling	Pile, drifter drill	9	34:51	5310
5	truck	Dumper	10	13:49	2737
6	grading	Grader	9	54:39	15984
7	concrete_mix	Concrete mixer	15	21:55	2334
8	concrete_grinding	Grinder	15	29:38	4277
9	drilling	Hand held power drill	7	1:20:20	9947
Total			97	7:06:03	64696

TABLE 1. Details of different sound classes used in the paper.

N.	Name	Train Set		Test Set	
		Duration	N. of instances	Duration	N. of instances
1	concrete_breaking	35:33	4661	13:48	1398
2	ground_excavating	1:11:38	11997	31:28	3601
3	bulldozer	26:35	1887	11:49	563
4	piling	25:35	4111	09:16	1199
5	truck	11:49	2138	02:00	599
6	grading	40:02	11828	14:37	4156
7	concrete_mix	15:45	1795	06:10	539
8	concrete_grinding	20:57	3292	08:41	985
9	drilling	1:00:00	7652	20:20	2295
Total		5:07:54	49361	1:58:09	15335

TABLE 2. Details of train and test sets. (NEW!!)

Classifier	Description	References
BayesNet	Bayesian Classifier	(John and Langley 1995)
DecisionTable	Decision table majority classifier	(Kohavi 1995)
HoeffdingTree	Hoeffding Tree	(Hulten et al. 2001)
IBk	k Nearest Neighborhood classifier (kNN)	(Altman 1992; Aha et al. 1991)
J48	C4.5 decision tree	(Quinlan 1993)
JRip	Cohen version of IREP	(Cohen 1995)
KStar	K* instant based learner	(Cleary and Trigg 1995)
LWL	Locally weighted learning	(Frank et al. 2003)
MLP	MultiLayer Perceptron	(Haykin 2009)
NaiveBayes	Naïve Bayesian Classifier	(John and Langley 1995)
OneR	1R classifier	(Holte 1993)
PART	Decision list	(Frank and Witten 1998)
RandomForest	Forest of random trees	(Breiman 2001)
RandomSubSpace	Multiple Trees	(Ho 1998)
RandomTree	Random tree (no pruning)	(Rokach and Maimon 2014)
SimpleLogistic	Linear logistic regression	(Sumner et al. 2005)
SMO	Sequential Minimum Optimization	(Plat 1998)

TABLE 3. The seventeen different classifiers compared in this paper.

Symbol	Description	# of features	Window [ms]	Overlap [ms]
ZCC	Zero Crossing Count	10	5	2.5
CZCR	Variance of Zero Crossing Rate	1	5	2.5
HZCRR	High Order Zero Crossing Rate Ratio	1	5	2.5
HOC	High Order Crossing	8	5	2.5
RMS	Root Mean Square	10	5	2.5
LEF	Low Energy Frame	1	5	2.5
STE	Short Time Energy	1	5	2.5
LSTER	Low Short Time energy Ratio	1	20	10
VLER	Variance of Low-Band Energy Ratio	1	20	10
VSFLUX	Variance of Spectrum Flux	1	20	10
LFRMS	Low-Frequency Root Mean Square	1	20	10
LFZCC9	9th-order Moment of LFZCC	1	10	5
SBC	Sub-Band Correlation	5	15	0
LPC	Linear Predictive Coefficients	7	200	0
MFCC	Mel-Frequency Cepstral Coefficients	13	200	0

TABLE 4. Set of the 62 used features.

Classifier	Parameter	Value
BayesNet	Batch size	100
	Estimator	Simple Estimator
	Search Algorithm	K2 (Hill climbing)
DecisionTable	Batch size	100
	Search Algorithm	Best First
HoeffdingTree	Batch size	100
	Grace Period	200
	Hoeffding Tie Threshold	0.05
	Minimum fraction of weight info gain	0.01
	Split confidence	10 ⁻⁷
IBk (kNN)	Batch size	100
	Number of neighbors	1
	Window size	0
J48	Batch size	100
	Confidence factor	0.25
	Number of Folds	3
JRip	Batch size	100
	Number of folds	3
	Minimum total weight	2
	Optimization runs	2
	Seed	1
KStar	Batch size	100
	Global blend	20
LWL	Batch size	100
	Number of neighbors	14
	Weighting kernel	0
MLP	Batch size	100
	Number of hidden units	40
	Activation function	Hyperbolic tangent
	Learning rate	0.01
	Momentum term	0.05
	Number of epochs	500
NaiveBayes	Batch size	100
OneR	Batch size	100
	Bucket size	6
PART	Batch size	100
	Confidence factor	0.25
	Number of folds	3
	Seed	1
RandomForest	Batch size	100
	Execution slots	1
	Number of iterations	500
RandomSubSpace	Batch size	100
	Execution slots	1
	Number of iterations	500
	Seed	5
	Sub-space size	0.5
RandomTree	Batch size	100
	<i>K</i> value	0
	Max depth	Unlimited
	Minimum total weight	1
	Min variance	0.001
	Seed	1
SimpleLogistic	Batch size	100
	Number of iterations	500
	Heuristic stop	50
	Weight trim beta	0.0
SMO	Batch size	100
	<i>c</i>	1.0
	ϵ	10 ⁻¹²
	Kernel	Polynomial
	Tolerance parameter	0.001

TABLE 5. The main hyper-parameters used in the tested classifiers.

Classifier	Accuracy [%]	Precision	Recall	F-Measure	MCC	AUC
BayesNet	79.74	0.808	0.797	0.801	0.769	0.964
DecisionTable	56.61	0.608	0.552	0.547	0.576	0.894
HoeffdingTree	41.22	0.589	0.544	0.531	0.497	0.891
IBk (kNN)	85.28	0.860	0.853	0.855	0.831	0.925
J48	70.24	0.716	0.702	0.707	0.661	0.829
JRip	78.84	0.799	0.788	0.792	0.759	0.914
KStar	80.37	0.814	0.804	0.807	0.799	0.901
LWL	33.62	0.358	0.331	0.344	0.208	0.843
MLP	91.06	0.913	0.932	0.932	0.919	0.957
NaiveBayes	79.81	0.808	0.798	0.802	0.770	0.884
OneR	39.48	0.401	0.398	0.399	0.247	0.612
PART	83.66	0.846	0.837	0.840	0.814	0.916
RandomForest	93.16	0.934	0.932	0.932	0.919	0.998
RandomSubSpace	81.81	0.829	0.818	0.822	0.793	0.897
RandomTree	77.36	0.784	0.774	0.777	0.741	0.869
SimpleLogistic	75.51	0.766	0.755	0.759	0.720	0.936
SMO	77.17	0.782	0.772	0.775	0.739	0.951

TABLE 6. Accuracy and other performance metrics of the evaluated classifiers. The values represent weighted averages among the classes.

	01	02	03	04	05	06	07	08	09
01	1398	0	0	0	0	0	0	0	0
02	0	3428	6	7	15	138	5	2	0
03	0	2	520	5	6	16	11	3	0
04	11	5	14	1072	25	29	12	13	18
05	0	41	8	2	529	13	0	5	1
09	14	77	54	49	27	3916	7	9	3
07	0	13	3	0	0	4	517	0	2
08	0	10	26	9	12	27	4	882	15
09	0	35	12	22	48	79	17	58	2024

TABLE 7. Confusing matrix of the Random Forest.

	01	02	03	04	05	06	07	08	09
01	1397	0	0	0	0	1	0	0	0
02	2	3365	14	21	26	152	11	7	3
03	0	5	489	8	14	17	9	20	1
04	17	14	26	1017	33	24	19	23	26
05	1	38	12	6	508	17	5	8	4
09	15	84	57	61	44	3851	21	12	11
07	0	19	12	0	0	7	497	0	4
08	2	21	35	16	12	35	8	837	19
09	0	42	18	27	45	83	16	61	2003

TABLE 8. Confusing matrix of the MLP.

	01	02	03	04	05	06	07	08	09
01	1396	0	0	1	0	1	0	0	0
02	4	3248	23	42	74	162	13	33	2
03	0	5	412	7	15	13	10	94	7
04	21	32	52	893	37	48	61	19	36
05	0	31	48	29	426	35	9	15	6
09	24	99	73	61	57	3708	69	47	18
07	1	22	34	3	8	29	379	53	10
08	2	43	67	55	13	37	25	707	36
09	5	61	34	24	72	103	19	69	1908

TABLE 9. Confusing matrix of the kNN (IBk).

	01	02	03	04	05	06	07	08	09
01	1394	0	0	1	0	3	0	0	0
02	5	3221	26	50	27	158	64	41	9
03	0	7	385	12	24	33	12	84	6
04	19	35	63	859	41	55	72	21	34
05	1	26	54	37	401	39	12	21	8
09	28	104	69	75	64	3681	68	42	25
07	3	31	42	9	21	29	341	47	16
08	2	43	65	57	18	46	28	691	35
09	8	57	48	41	95	99	31	59	1857

TABLE 10. Confusing matrix of the PART.

	01	02	03	04	05	06	07	08	09
01	1363	5	3	5	3	9	3	4	3
02	18	2939	68	84	71	234	97	56	34
03	26	41	195	62	42	41	32	83	41
04	44	55	94	610	69	86	106	57	78
05	23	67	78	65	183	93	41	26	23
09	51	156	97	128	96	3317	112	108	91
07	32	63	74	68	59	71	123	25	24
08	46	59	97	73	58	70	44	472	66
09	57	83	89	122	109	141	57	68	1569

TABLE 11. Confusing matrix of the J48.

N.	Class	DCNN	DRNN	DBN
1	concrete_breaking	92.13	90.72	87.75
2	ground_excavating	91.02	89.60	86.77
3	bulldozer	87.69	87.02	83.39
4	piling	97.68	98.89	94.79
5	truck	83.25	82.53	78.42
6	grading	98.79	97.16	95.02
7	concrete_mix	99.34	98.85	95.77
8	concrete_grinding	98.79	97.76	95.97
9	drilling	98.21	97.48	92.31
	Overall	94.10	93.33	90.02

TABLE 12. Per-class and overall accuracy of Deep Learning approaches considered for comparison.

Classifier	Backhoe	Cycles lifting	Driving the pile
RandomForest	71.89	85.88	68.88
RandomSubSpace	70.69	84.95	68.03
MLP	69.66	82.13	62.23
IBk	58.84	67.48	63.74

TABLE 13. Accuracy [%] obtained by four of the best performing classifiers in the three considered scenarios.

List of Figures

1	Architecture of the classifier.	41
2	Segmentation of superframes in subframes.	42
3	Comparison of accuracy performances of the classifiers.	43
4	Sound recording of the erecting and driving the bridge pile	44
5	Schedule of the bridge construction	45

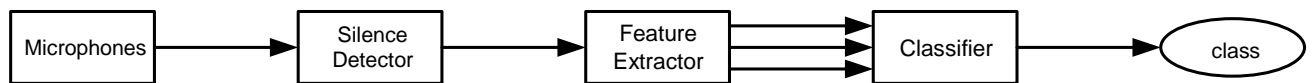


FIG. 1. Architecture of the classifier.

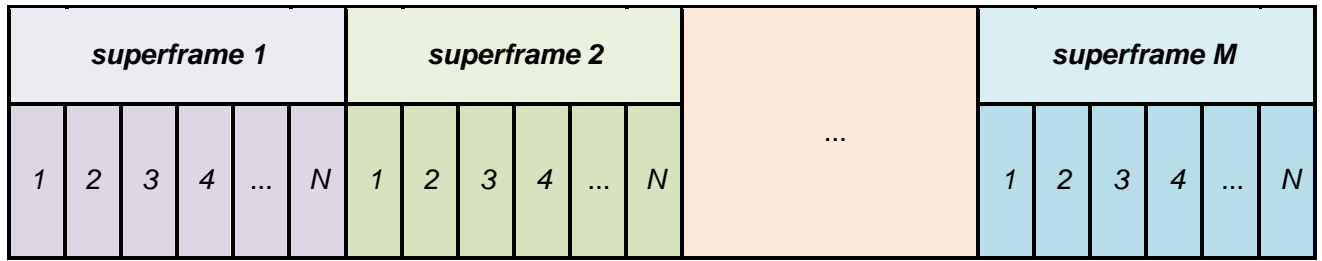


FIG. 2. Segmentation of superframes in subframes.

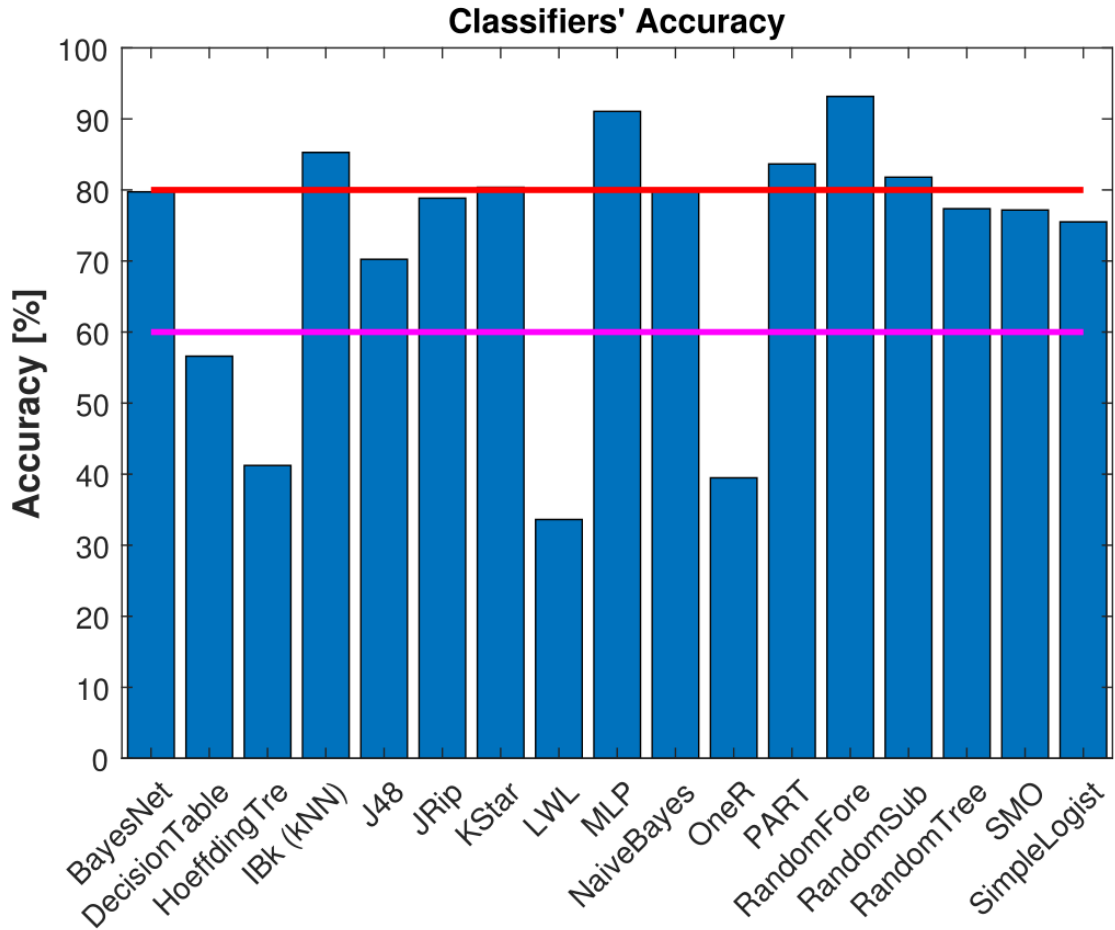


FIG. 3. Comparison of accuracy performances of the classifiers.



FIG. 4. Sound recording of the erecting and driving the bridge pile














East Baton Rouge Bridge Project				
Activity Name	WBS Name	Original Duration	Start	Finish
 Bent 4 Piles	Tucker	5	23-Feb-18	02-Mar-18
 Bent 3 Piles	Tucker	5	06-Mar-18	14-Mar-18
 Bent 2 Piles	Tucker	5	15-Mar-18	21-Mar-18
 Bent 1 Piles	Tucker	5	23-Mar-18	30-Mar-18
 Bent 5 Piles	Tucker	5	03-Apr-18	09-Apr-18
 Bent 6 Piles	Tucker	5	10-Apr-18	17-Apr-18
 Bent 7 Piles	Tucker	5	18-Apr-18	27-Apr-18
 South Side Drainage E	Tucker	3	25-Apr-18	30-Apr-18
 North Side Drainage E	Tucker	3	26-Apr-18	28-Apr-18
 North Side Rip Rap	Tucker	2	27-Apr-18	30-Apr-18
 South Side Rip Rap	Tucker	2	27-Apr-18	28-Apr-18
 Bent 4 Cap	Tucker	3	01-May-18	03-May-18
 North Side Embankme	Tucker	1	01-May-18	01-May-18
 South Side Embankme	Tucker	1	01-May-18	01-May-18
 North Side Storm Drain	Tucker	1	02-May-18	02-May-18
 South Side Storm Drain	Tucker	1	02-May-18	02-May-18

FIG. 5. Schedule of the bridge construction