

Article

Nondestructive Inspection of Reinforced Concrete Utility Poles with ISOMAP and Random Forest

Saeed Ullah ¹, Minjoong Jeong ^{2,*} and Woosang Lee ³

¹ University of Science and Technology (UST), 217 Gajeong-ro, Yuseong-gu, Daejeon 34113, Korea; saeedonline12@gmail.com

² Korea Institute of Science and Technology Information (KISTI), 245 Daehak-ro, Yuseong-gu, Daejeon 34141, Korea

³ Smart C&S Co., Ltd., Yongsan-dong, Yuseong-gu, Daejeon 34141, Korea; smartcs@chol.com

* Correspondence: jeong@kisti.re.kr; Tel.: +82-42-869-0632

Received: 1 August 2018; Accepted: 10 October 2018; Published: 15 October 2018



Abstract: Reinforced concrete poles are very popular in transmission lines due to their economic efficiency. However, these poles have structural safety issues in their service terms that are caused by cracks, corrosion, deterioration, and short-circuiting of internal reinforcing steel wires. Therefore, they must be periodically inspected to evaluate their structural safety. There are many methods of performing external inspection after installation at an actual site. However, on-site nondestructive safety inspection of steel reinforcement wires inside poles is very difficult. In this study, we developed an application that classifies the magnetic field signals of multiple channels, as measured from the actual poles. Initially, the signal data were gathered by inserting sensors into the poles, and these data were then used to learn the patterns of safe and damaged features. These features were then processed with the isometric feature mapping (ISOMAP) dimensionality reduction algorithm. Subsequently, the resulting reduced data were processed with a random forest classification algorithm. The proposed method could elucidate whether the internal wires of the poles were broken or not according to actual sensor data. This method can be applied for evaluating the structural integrity of concrete poles in combination with portable devices for signal measurement (under development).

Keywords: nondestructive inspection; machine learning; dimensionality reduction; classification; ISOMAP; random forest

1. Introduction

Reinforced concrete poles are commonly used for telephone and electricity transmission. Their greater mechanical strength, cost effectiveness, longer life span (over 50 years), potential to cover longer distances, and better electrical resistance are the key reasons for their widespread usage [1]. Further, reinforced concrete poles constitute an alternative to steel poles because of their higher variability of architectural shapes and comparatively low maintenance costs. When compared to timber poles, concrete poles offer better resistance to hurricanes and are more robust against decay and fire [2]. However, structural safety defects can occur in reinforced concrete poles because of cracks, voids, corrosion, deterioration, and short-circuiting of internal reinforcing steel wires. These problems in reinforced concrete poles are caused by atmospheric exposure, earthquakes, hurricanes, floods, moisture changes, poor construction practices, and various chemical, mechanical, and physical reactions [3–6]. Structural problems demand more attention than non-structural problems. According to Doukas et al. [7] approximately half of accidents involving reinforced concrete poles are related to structural defects. These defects constitute a major reason for reduced life expectancy, structural strength, and serviceability. Further, structural defects can result in subsequent failure.

Moreover, the failure of poles can cause serious problems, such as falling onto people or animals, causing injury (or death), and service disruption. Early detection of such problems can save precious lives, time, and money. Therefore, it is necessary to identify these defects at an early stage for which accurate inspection and monitoring is mandatory on regular basis. Thus, to ensure the structural integrity of utility poles, those already in service poles must be regularly inspected or constantly monitored [8].

Accurate integrity assessment of existing poles is very important, because maintenance and repair costs are increasing rapidly. Hence, there is an increasing demand to develop more accurate, consistent, and reliable non-destructive methods for assessing the condition of in-service poles [2]. Stewart et al. [5] indicated that structural reliability assessments need to be updated at regular intervals through on-site inspections and testing.

For the inspection of reinforced concrete utility poles, traditional methods, such as visual inspection, are still commonly used. However, direct sampling and observation of the structure is very difficult for utility poles [9]. In such situations, Non-Destructive Testing (NDT) methods are applied. Here, NDT is defined as the course of inspecting, testing, or evaluating materials without destroying the structure or serviceability of any part of the system. Examples of NDT methods for structural health monitoring of concrete (with and without reinforcement) materials are visual inspection, rebound hammer, stress-wave methods, ultrasound monitoring, infrared thermography, nuclear methods, X-ray computed tomography, magnetic and electrical methods, penetrability methods, and radar (Radio Detection and Ranging) [9–17]. Although many of these methods render better results, they require highly qualified, experienced, and licensed inspectors for the testing and interpretation of the results. Additionally, they require expensive and bulky equipment to carry out complex signal processing. Electrical power is needed in some methods, which can lead to short circuits in reinforced concrete. Moreover, safety concerns during testing are also a problem in many of these methods. Concerning all of the methods listed above, many case studies exist in which different techniques have been combined to improve performance, reduce cost, or address any other limitations of the individual methods [9,18].

Recently, Dackermann et al. [19,20] developed NDT methods with advanced signal processing and machine learning techniques for timber poles, self-compacting concrete poles without steel reinforcement, and generic concrete poles without steel reinforcement. They adopted stress-wave techniques for data gathering, achieving accuracy levels up to 93%. A large number of tactile transducers and accelerometers (along with other devices) were used in these methods. Dackermann et al. [19] applied Principal Component Analysis (PCA) for feature reduction and Support Vector Machines (SVM) for classification of signals to predict pole damage conditions, while Dackermann et al. [20] used Frequency Response Functions (FRFs) and PCA for extracting signal features when capturing single-mode stress waves for condition assessment. It is very difficult to find any works in the literature related to the application of machine learning or data analysis techniques for NDT on reinforced concrete utility poles.

In this paper, we propose a novel NDT method for structural health monitoring of reinforced concrete utility poles that is based on machine learning techniques. To train the machine-learning algorithm, we used data gathered through magnetic sensors at actual site by SMART C&S [21]. Moreover, ISOMAP [22] is applied on the data for feature reduction, and the resulting refined data constitute the input to a random forest classifier [23,24]. In particular, the refined signals are trained with a random forest method and are categorized into safe and crack signals that are based on actual field experiments. Safe signals are those corresponding to non-damaged wires, whereas crack signals correspond to damaged wires. The proposed system is able to predict 97% correct signals among safe and crack signals on unclassified sensor data. We also applied two other well-known machine-learning algorithms on our data: Support Vector Machine (SVM) [25] and decision trees [26]. The detailed comparison of these two algorithms with random forest on our data is described in Section 4.2.

All of the algorithms used in this work were written and executed with the Python language machine learning scikit-learn library [27].

The rest of our paper is organized as follows: Section 2 describes the complete experimental setup for data gathering. The proposed system resulting from our research work is explained in Section 3. The experimental results are presented and discussed in Section 4. Finally, the conclusions are presented in Section 5.

2. Experimental Setup

In this section, we present the applied data gathering techniques, the used devices, and their specifications. The whole setup for this project is as follows: a magnetic sensing device, a Data Acquisition (DAQ) device, a cable, and a portable computer. To inspect a utility pole, the magnetic sensing device can be inserted into the pole through holes, and the signals can be gathered with the help of the DAQ device. This device can be attached to a portable computer to check the signals with the patterns of our application. Regarding the training of our algorithm, the signals were gathered with the same device and the dataset was manually configured by the field engineers of SMART C&S. They collected signals from 30 reinforced concrete utility poles containing signals from damaged and non-damaged wires and labeled them accordingly. They uninstalled the poles and broke down the concrete cover over steel wires to pull the wires out the concrete. In particular, the field engineers pulled all of the wires out those 30 poles and gathered the signals from each wire. They inserted the sensors into the poles through the first bolt hole nearest to the ground and detected the signals from bottom portion up to the first bolt hole of the pole, because the tendency of defects in wires inside poles is mostly in the bottom portion. The measuring section for signal gathering inside poles was set to a maximum height of 4 m, and the measurement rate was fixed to 0.3 m/s. The diameter of steel wires in all of the poles was from 9 to 12 mm, and there were 16 steel wires (eight tension and eight reinforcing) in each utility pole. The thickness of the concrete cover over the steel wires was from 12 to 24 mm in each pole. The field engineers gathered 101 Hall Effect values for every signal. These values are referred to as “features” in our dataset. The dataset could contain many ambiguous, repeated, and unimportant features. Therefore, we filtered the dataset into meaningful features to be used with a classification algorithm, which is further explained in Section 3.

The eight-channel magnetic sensing device along with all the parts is depicted in Figure 1. The specifications of all the parts of this device are displayed in Table 1.

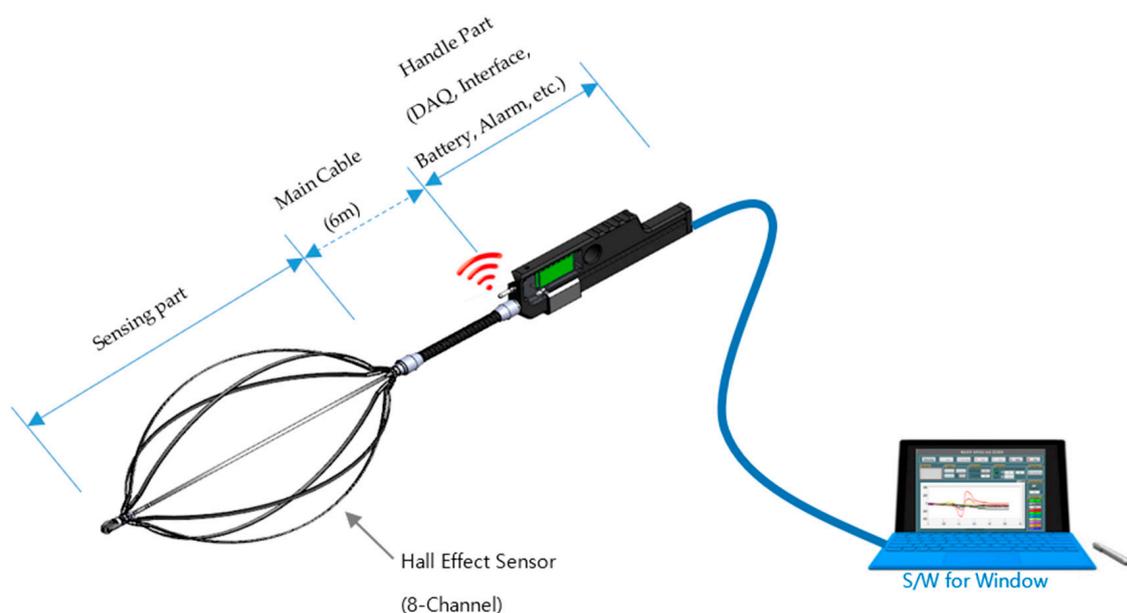
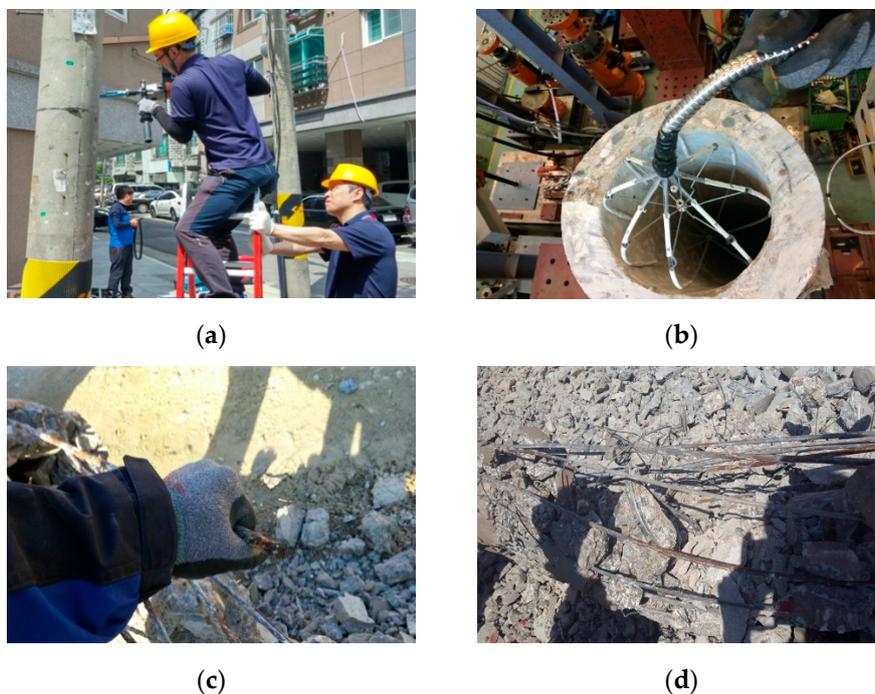


Figure 1. Magnetic sensing device with eight channels.

Table 1. Specifications of all the parts of the magnetic sensing device.

Sensor and DAQ	
ADC Resolution	16 bit
ADC Input Channel	8 Differential Input Channels
ADC Sampling rate	50 S/s
Main Cable	
Length	6 m
Diameter	22 mm

Figure 2a illustrates the process of removing a maintenance bolt and securing the bolt hole to insert the magnetic sensing device into a working utility pole. Figure 2b shows an example of laboratory verification experiments of the data-gathering device. Figure 2c shows an example of a broken steel wire and Figure 2d shows multiple steel wires inside the poles.

**Figure 2.** (a) Field testing; (b) laboratory testing; (c) a broken steel wire; (d) multiple steel wires.

3. Proposed System

The dataset for this research project is composed of n samples with m features, $R = \{(x_i^d, y_i), i = 1, 2, \dots, n\}$, where in our case $n = 240$, $m = 101$ and $d = 1, 2, \dots, m$, x_i is the input data and every row in x_i has a label $y_i \in \{0, 1\}$. Note that 0 indicates safe signals, while 1 denotes crack signals. Each row of x_i represents a signal, while d indicates the features of every signal, which are the Hall Effect values for each signal.

Figure 3 depicts plots of safe and crack signals from our dataset in two-dimensional graphs. Figure 3a shows a single sample of a safe signal in our dataset. Figure 3b depicts a single sample of a crack signal from our dataset. Figure 3c represents all of the safe signals in our dataset. Figure 3d shows all the crack signals in our dataset.

Magnetic sensors were used for data gathering and the dataset contains many replicated features. To reduce these, it is essential to apply a dimensionality reduction technique. Dimensionality reduction is also important for finding meaningful low-dimensional hidden structures in high dimensional data and to improve the performance of classification algorithms.

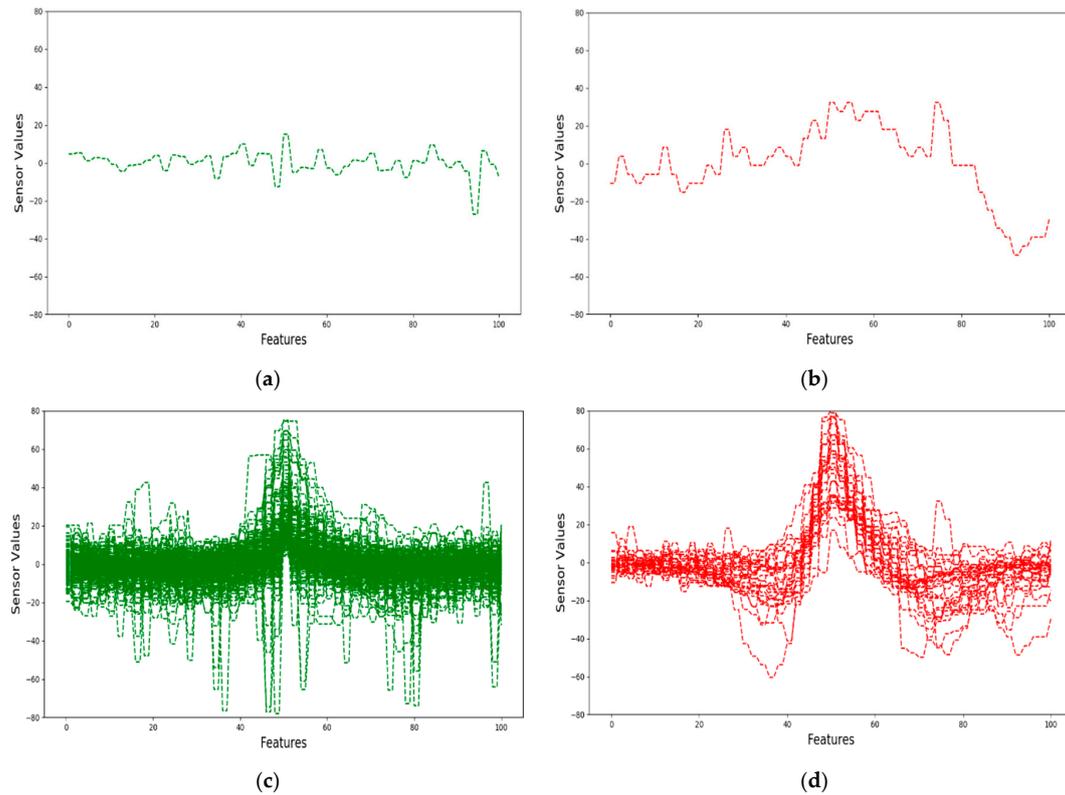


Figure 3. (a) A safe signal; (b) a crack signal; (c) safe signals; and, (d) crack signals.

3.1. The Flowchart of Our Proposed System

The overall flow of our proposed system is depicted in Figure 4.

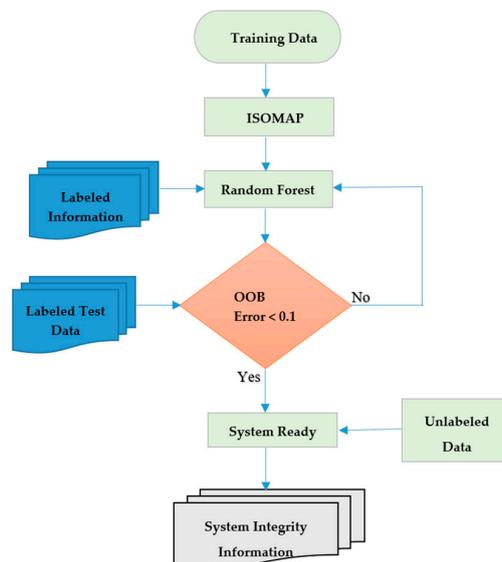


Figure 4. Flowchart of the proposed system.

The flowchart in Figure 4 illustrates that for the dimensionality reduction of our data, we applied ISOMAP, which is a manifold-based global geometric framework mainly used for non-linear dimensionality reduction. Non-linear dimensionality reduction techniques became very popular in the last decade due to their superior performance on high dimensional data when compared to linear techniques [28]. Jeong [29] showed that ISOMAP outperforms PCA (a linear dimensionality reduction technique) on high dimensional data.

3.2. ISOMAP

ISOMAP is an extension of Multidimensional Scaling (MDS), a classical method for embedding dissimilarity information into a Euclidean space. The main concept of ISOMAP is to replace Euclidean distances with an approximation of the geodesic distances on the manifold. ISOMAP is a three-step process: (1) Construct neighborhood graph; k -nearest neighbors of every data point are defined and represented by a graph G ; in G , every point is connected to its nearest neighbors by edges. (2) Compute the shortest paths; the geodesic distances between all the pairs of points are estimated using the Dijkstra algorithm [30]; the squares of these distances are stored in a matrix of graph distances $D^{(G)}$. (3) Construct d -dimensional embedding; the classical MDS algorithm is applied on $D^{(G)}$ in order to find a new embedding of the data in a d -dimensional Euclidean space Y .

Deciding the Number of Dimensions for the ISOMAP Algorithm

Our dataset is composed of 240 data items with 101 features. Firstly, we have to determine the number of dimensions of the ISOMAP space. For this purpose, we calculated the residual variance [22,31], which is typically used to evaluate the error of dimensionality reduction. Residual variance is defined in Equation (1), as follows:

$$R_d = 1 - r^2(G, D_d) \quad (1)$$

where R_d is the residual variance, G is the geodesic distance matrix, D_d is the Euclidean distance matrix in the d -dimensional space, and $r(G, D_d)$ denotes the correlation coefficient of G and D_d . The value of d is determined by a trial-and-error approach to reduce the residual variance.

Figure 5 shows that in all cases the residual variance decreases as the number of dimensions d is increased. It is recommended in [22,31] to select the number of dimensions at which this curve ceases to decrease significantly with added dimensions. To evaluate the intrinsic dimensionality of our data, we searched for the “elbow”, at which this curve is not decreasing significantly with increasing dimensions. The arrow mark highlights the approximate dimensions in our data. If the dimensions are increased and exhibit some residual variance of the increased dimensions, then the data can be better explained by ISOMAP and the classification algorithm will perform better based on the residual variance. For example, in our dataset for a number of dimensions equal to 50 having a residual variance equal to 0.03, the performance is slightly better than for a number of dimensions equal to 8. We set eight dimensions for our classification algorithm because of the computational cost of both ISOMAP and the random forest. We had already achieved better performance with eight dimensions. However, the performance is not improved with increasing dimensions once the residual variance becomes zero.

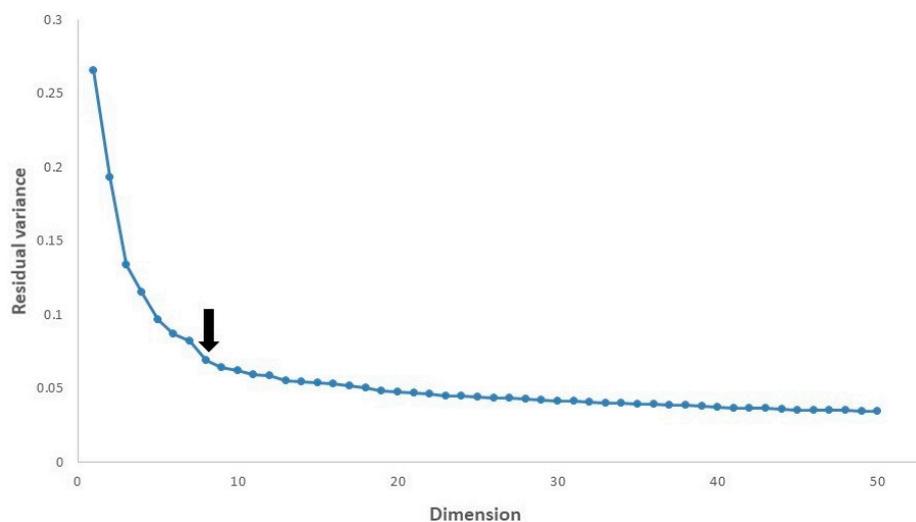


Figure 5. The residual variance of ISOMAP in our data.

3.3. Random Forest

ISOMAP reduces our dataset into eight dimensions on which a random forest classification algorithm is applied for the purpose of training features in the data. The random forest is an ensemble method, and generally, ensemble methods perform better than single classifiers. They are built from a set of classifiers and then use the weightage of their predictions to render the final output [32]. These methods employ more than one classification technique and then combine their results. They are notably less prone to overfitting. The random forest algorithm consists of three steps: (1) Construct n_{trees} bootstrap samples from the input data while using the Classification And Regression Trees (CART) [33] methodology. (2) Grow an unpruned tree for each of the n_{trees} , randomly sample m_{try} of the predictors, and choose the best split among those variables. (3) Aggregate the predictions of the n_{tree} trees for the prediction of new data. We applied the scikit-learn [27] implementation that combines classifiers by averaging their probabilistic prediction.

Choosing the number of trees in a random forest is an open question. Breiman [23] mentioned that, the greater the number of trees, the better the performance of the random forest. However, it is very difficult to find the optimal number of trees for the algorithm. Oshiro et al. [34] applied random forest on 29 different types of datasets with varying number of trees L in exponential rates using base two, which is $L = 2^j, j = 1, 2, \dots, 12$. They concluded that sometimes a larger number of trees in the forest only increases its computational cost, having no significant impact on its performance. We applied the same method as that of Oshiro et al. [34] to decide the number of trees in our random forest classifier. The number of trees and their performance on our dataset is reported in Table 2.

Table 2. Performance evaluation with different number of trees.

No of Trees	2	4	8	16	32	64	128	256	512
Accuracy	0.93	0.94	0.97	0.97	0.97	0.97	0.97	0.98	0.98

Table 2 shows that using 8, 16, 32, 64, or 128 as the number of trees in a random forest on our dataset resulted in the same performance. Therefore, we set the number of trees to eight. We did not set a large number of trees, as this would increase the computational cost.

To decide on the number of random samples, the common term applied is \sqrt{p} , where p is the number of predictors. Our dataset for random forest has eight features, therefore, we choose $m_{\text{try}} = 3$.

The OOB (out-of-bag) error estimate is used to check the performance of a random forest model. This error is calculated by averaging only those trees corresponding to bootstrap samples that render wrong predictions [23,35]. We stopped training our model when the OOB error reduced below 10%, which is a quite good fit and shows that more than 90% of the decision trees are rendering correct predications. As we are applying random training to a random forest, we executed the model 100 times and then calculated the mean OOB error from all the OOB error values.

4. Results and Discussion

4.1. Performance Evaluation of Our Classifier

We randomly chose 210 data items for training and 30 data items to test our classification algorithm. The structural safety labels were separated and were not shown to the algorithm for evaluation purposes. The predicted results from the algorithm were then compared with the labels. The accuracy was calculated as the ratio between accurate predicted instances and the total number of examined instances, as defined in Equation (2). As the random forest algorithm renders different results on different executions, it was executed 100 times. The accuracy, defined as the mean value of all the 100 calculated accuracies, was 97% in our case. Our dataset consisted of 211 safe signals and 29 crack signals. Therefore, it is an imbalanced dataset. It is likely that the algorithm will only predict safe signals and calculate better accuracy on the basis of the result. In such cases, accuracy alone is not

enough for determining the performance of an algorithm. The accuracy mostly biases to the majority class data, and presents several other weaknesses, such as less distinctiveness, discriminability, and informativeness. Further, it is possible for a classifier to perform well on one metric while being suboptimal on other metrics. Moreover, different performance metrics measure different tradeoffs in the predictions made by a classifier. Therefore, it is essential to assess algorithms on a set of performance metrics [36]. We thus calculated the confusion matrix, precision, recall [37], and F-measure [37,38], to discriminate accurately and to select an optimal solution for our model. The confusion matrix is widely used for describing the performance of a classifier, because it displays the ways in which a classification model is confused during predication. The confusion matrix is composed of rows and columns corresponding to the classification labels. The rows of the table represent the actual class, while the columns represent the predicted class. The main diagonal elements in the confusion matrix are TP and TN, which denote correctly classified instances, while other elements (FP and FN) denote incorrectly classified instances. Here, TP means true positive, which is when the algorithm correctly predicts the positive class, while TN means true negative, which is when the algorithm correctly predicts the negative class. Further, FP means false positive, which is when the algorithm fails to predict the negative class correctly. Finally, FN means false negative, which is when the algorithm fails to predict the positive class correctly. In our data, the safe signals are represented as a positive class, while crack signals are represented as a negative class. The confusion matrix of our random forest algorithm on the test data is presented in Table 3.

Table 3. Confusion matrix of our proposed system.

	Predicated Safe Signals	Predicated Crack Signals
Actual Safe Signals	TP = 25	FN = 0
Actual Crack Signals	FP = 1	TN = 4

Note that total test points are equal to 30, and the total number of correctly classified instances is calculated as $25 + 4 = 29$, while the total number of incorrectly classified instances is given by $0 + 1 = 1$.

The confusion matrix clearly shows that there is only one mistake made by our algorithm, namely a crack signal predicted as safe signal. On the basis of the confusion matrix, the result is quite good. However, an analysis exclusively based on the confusion matrix is not sufficient when evaluating the performance of an algorithm. In the case of imbalanced classes, precision, and recall constitute a useful measure of success of prediction. Precision and recall are commonly used in information retrieval for evaluation of retrieval performance [39]. Precision is applied to measure the correctly predicted positive patterns from the total predicted patterns in a positive class, whereas recall is used to find the effectiveness of a classifier in identifying positive patterns. For full evaluation of the effectiveness of a model, examining both precision and recall is necessary. High precision represents a low false positive rate, while high recall represents a low false negative rate. Accuracy A , precision P , and recall R are defined in Equation (2).

$$A = \frac{TP + TN}{TP + FP + FN + TN}, P = \frac{TP}{TP + FP} \text{ and } R = \frac{TP}{TP + FN} \quad (2)$$

The precision of our classifier on the test data is 96%, while the recall is 100%. These are high values. However, precision and recall are still not sufficient to select an optimal solution or algorithm. To find a balance between precision and recall, F-measure is used. For computing score, the F-measure considers both precision and recall. Moreover, F-measure indicates how many instances the classifier classifies correctly without missing a significant number of instances. The greater the F-measure,

the better the performance of the model. F-measure is defined as the harmonic mean of precision and recall, as expressed in Equation (3).

$$F - \text{measure} = \frac{2}{1/P + 1/R} \quad (3)$$

The F-measure of our algorithm is 98%. The performance can now be measured in terms of classification accuracy, confusion matrix, precision, recall, and F-measure. We calculated all of them for the evaluation of our classifier, and the results demonstrate that our classifier is very reliable.

4.2. Comparison of Random forest with SVM and Decision Trees on Our Data

To evaluate and compare the results of a random forest with other machine learning algorithms on our ISOMAP-based reduced data, we applied SVM and decision trees. SVM attempts to find the optimal separating hyperplane between objects of different classes. Further, SVM is the most suitable for binary classification but can also be configured for multi-classification tasks. Decision trees are applied for both classification and regression tasks. In decision trees, each node represents a feature, each link represents a decision, and each leaf represents an outcome. The root node is defined as the attribute that best classifies the training data. This process is repeated for each branch of the tree. An SVM classifier with sigmoid kernel and a gini index-based unpruned decision tree with best split method were implemented. Using SVM and decision trees, we achieved 90% and 93% accuracy, respectively. A complete comparison of all these algorithms on our test data is depicted in Figure 6.

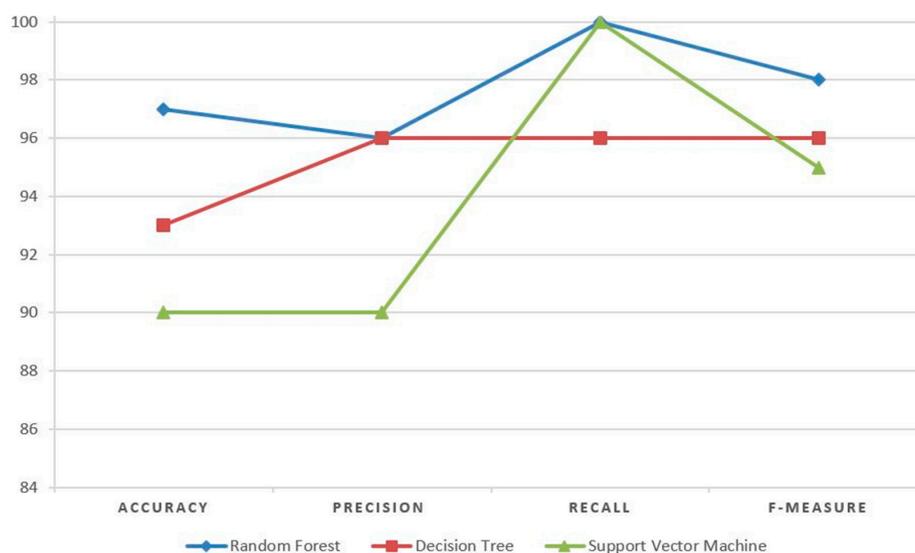


Figure 6. Performance measure graph of different algorithms.

Figure 6 clearly shows that random forests outperform SVM and decision trees. The precision of random forests and decision trees are the same, as well as the recall of random forests and SVMs, but the accuracy and F-measure of random forests are superior.

5. Conclusions

In this paper, we proposed a structural safety assessment method for reinforced concrete utility poles with the use of ISOMAP and random forests. The proposed system is able to identify the condition of wires inside reinforced concrete poles. It is also an easy and inexpensive system to adopt, because only a few devices are needed. We had a limited number of trained data items from field experiments. Therefore, we opted for machine learning techniques rather than deep learning methods. We used ISOMAP for data reduction and then applied a random forest classifier for classification purposes. The random forest algorithm outperformed other machine learning algorithms (such as SVM

and decision trees) on our data set. In our first attempt, we achieved better performance. In future, we can receive more experimental data from field engineers and apply deep learning methods to accomplish better performance with higher reliability.

Author Contributions: M.J. initiated the idea and provided supervision and guidance in experiments and analysis; S.U. performed machine learning modeling and was the primary writer of the paper; W.L. designed and conducted the experiments for data gathering.

Funding: This work was supported by the research program of Korea Institute of Science and Technology Information (KISTI).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kliukas, R.; Daniunas, A.; Gribniak, V.; Lukoseviciene, O.; Vanagas, E.; Patapavicius, A. Half a century of reinforced concrete electric poles maintenance: Inspection, field-testing, and performance assessment. *Struct. Infrastruct. Eng.* **2018**, *14*, 1221–1232. [[CrossRef](#)]
2. Baraneedaran, S.; Gad, E.F.; Flatley, I.; Kamiran, A.; Wilson, J.L. Review of In-Service Assessment of Timber Poles. In Proceedings of the Australian Earthquake Engineering Society (AEES), Newcastle, New South Wales, Australia, 11–13 December 2009.
3. Miszczyk, A.; Szocinski, M.; Darowicki, K. Restoration and preservation of the reinforced concrete poles of fence at the former Auschwitz concentration and extermination camp. *Case Stud. Constr. Mater.* **2016**, *4*, 42–48. [[CrossRef](#)]
4. Cairns, J.; Plizzari, G.A.; Du, Y.; Law, D.W.; Franzoni, C. Mechanical properties of corrosion-damaged reinforcement. *ACI Mater. J.* **2005**, *102*, 256.
5. Stewart, M.G. Spatial and time-dependent reliability modelling of corrosion damage, safety and maintenance for reinforced concrete structures. *Struct. Infrastruct. Eng.* **2012**, *8*, 607–619. [[CrossRef](#)]
6. Ying, L.; Vrouwenvelder, T. Service life prediction and repair of concrete structures with spatial variability. *Heron* **2007**, *52*, 251–267.
7. Doukas, H.; Karakosta, C.; Flamos, A.; Psarras, J. Electric power transmission: An overview of associated burdens. *Int. Energy Res.* **2011**, *35*, 979–988. [[CrossRef](#)]
8. Val, D.; Stewart, M.G. Reliability Assessment of Ageing Reinforced Concrete Structures—Current Situation and Future Challenges. *Struct. Eng. Int.* **2009**, *19*, 211–219. [[CrossRef](#)]
9. Breyse, D. *Non-Destructive Assessment of Concrete Structures: Reliability and Limits of Single and Combined Techniques*; Springer: Talence, France, 2012.
10. No, T.C.S. Guidebook on Non-Destructive Testing of Concrete Structures. Available online: https://www-pub.iaea.org/MTCD/Publications/PDF/TCS-17_web.pdf (accessed on 12 October 2018).
11. Helal, J.; Sofi, M.; Mendis, P. Non-Destructive Testing of Concrete: A Review of Methods. *Electron. J. Struct. Eng.* **2015**, *14*, 97–105.
12. Davis, A.G. *Nondestructive Test Methods for Evaluation of Concrete in Structures*. Available online: <https://pdfs.semanticscholar.org/de1b/9bf772ab6e63b9a7f90254ef8ae94765f3c5.pdf> (accessed on 12 October 2018).
13. Szymanik, B.; Frankowski, P.K.; Chady, T.; John Chelliah, C.R.A. Detection and Inspection of Steel Bars in Reinforced Concrete Structures Using Active Infrared Thermography with Microwave Excitation and Eddy Current Sensors. *Sensors* **2016**, *16*, 234. [[CrossRef](#)] [[PubMed](#)]
14. Hermanek, P.; Carmignato, S. Reference object for evaluating the accuracy of porosity measurements by X-ray computed tomography. *Case Stud. Nondestruct. Test. Eval.* **2016**, *6*, 122–127. [[CrossRef](#)]
15. Schickert, M.; Tümmeler, U. Initial Measurement Results of an Automated Ultrasonic Scanning and Imaging System. In Proceedings of the 7th International Symposium on Non Destructive Testing in Civil Engineering, Nantes, France, 30 June–3 July 2009.
16. Rucka, M.; Wilde, K. Ultrasound monitoring for evaluation of damage in reinforced concrete. *Bull. Pol. Acad. Sci. Tech. Sci.* **2015**, *63*, 65–75. [[CrossRef](#)]
17. Makar, J.; Desnoyers, R. Magnetic field techniques for the inspection of steel under concrete cover. *NDT E Int.* **2001**, *34*, 445–456. [[CrossRef](#)]

18. Benyahia, A.K.; Ghrici, M.; Kenai, S.; Breyse, D.; Sbartai, Z.M. Analysis of the relationship between nondestructive and destructive testing of low concrete strength in new structures. *Asian J. Civ. Eng.* **2017**, *18*, 191–205.
19. Dackermann, U.; Yu, Y.; Niederleithinger, E.; Li, J.; Wiggenger, H. Condition Assessment of Foundation Piles and Utility Poles Based on Guided Wave Propagation Using a Network of Tactile Transducers and Support Vector Machines. *Sensors* **2017**, *17*, 2938. [[CrossRef](#)] [[PubMed](#)]
20. Dackermann, U.; Yu, Y.; Li, J.; Niederleithinger, E.; Wiggenger, H. A new non-destructive testing system based on narrow-band frequency excitation for the condition assessment of pole structures using frequency response functions and principle component analysis. In Proceedings of the International Symposium Non-Destructive Testing in Civil Engineering, Berlin, Germany, 15–17 September 2015.
21. SMART C&S. Available online: <http://smartcs.co.kr/> (accessed on 12 October 2018).
22. Tenenbaum, J.B.; de Silva, V.; Langford, J.C. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* **2000**, *290*, 2319–2323. [[CrossRef](#)] [[PubMed](#)]
23. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
24. Liaw, A.; Wiener, M. Classification and Regression by random Forest. *R News* **2002**, *2*, 18–22.
25. Osuna, E.; Freund, R.; Girosi, F. *Support Vector Machines: Training and Applications*; Massachusetts Institute of Technology: Cambridge, MA, USA, 1997; Available online: <https://dspace.mit.edu/handle/1721.1/7290> (accessed on 12 October 2018).
26. Quinlan, J.R. Induction of decision trees. *Mach. Learn.* **1986**, *1*, 81–106. [[CrossRef](#)]
27. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
28. Van Der Maaten, L.; Postma, E.; Jaap Van Den, H. Dimensionality Reduction: A Comparative Review. Available online: <http://www.math.chalmers.se/Stat/Grundutb/GU/MSA220/S18/DimRed2.pdf> (accessed on 12 October 2018).
29. Jeong, M.; Choi, J.H.; Koh, B.H. Isomap-based damage classification of cantilevered beam using modal frequency changes. *Struct. Control Health Monit.* **2014**, *21*, 590–602. [[CrossRef](#)]
30. Rivest, R.L.; Cormen, T.H.; Leiserson, C.; Stein, C. *Introduction to Algorithms*; MIT Press: Cambridge, MA, USA, 2001.
31. Liang, Y.M.; Shih, S.W.; Shih, A.C.C.; Liao, H.Y.M.; Lin, C.C. Unsupervised Analysis of Human Behavior Based on Manifold Learning. Available online: https://www.researchgate.net/profile/Sheng-Wen_Shih/publication/221373574_Unsupervised_Analysis_of_Human_Behavior_based_on_Manifold_Learning/links/02bfe50cf22736cb7a000000.pdf (accessed on 12 October 2018).
32. Dietterich, T.G. Ensemble Methods in Machine learning. Available online: https://link.springer.com/chapter/10.1007/3-540-45014-9_1 (accessed on 12 October 2018).
33. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; Wadsworth: Belmont, CA, USA, 1984.
34. Oshiro, T.M.; Perez, P.S.; Baranauskas, J.A. How Many Trees in a Random Forest? Available online: https://link.springer.com/chapter/10.1007/978-3-642-31537-4_13 (accessed on 12 October 2018).
35. Friedman, J.; Hastie, T.; Tibshirani, R. *The Elements of Statistical Learning*; Springer: New York, NY, USA, 2001.
36. Caruana, R.; Niculescu-Mizil, A. An Empirical Comparison of Supervised Learning Algorithms. In Proceedings of the 23rd International Conference on Machine learning, Pittsburgh, PA, USA, 25–29 June 2006.
37. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [[CrossRef](#)]
38. Sasaki, Y. The Truth of the F-Measure. Available online: <https://www.cs.odu.edu/~mukka/cs795sum09dm/Lecturenotes/Day3/F-measure-YS-26Oct07.pdf> (accessed on 12 October 2018).
39. Lewis, D.D. Evaluating Text Categorization. Available online: <http://www.aclweb.org/anthology/H91-1061> (accessed on 12 October 2018).

