MotionWall: A Cost-Effective System For Structural Health Monitoring

Eamon Magdoubi
E.Magdoubi@liverpool.ac.uk
University of Liverpool

Terry R. Payne
T.R.Payne@liverpool.ac.uk
University of Liverpool

Chris Xiaoxuan Lu
xiaoxuan.lu@ed.ac.uk
University of Edinburgh

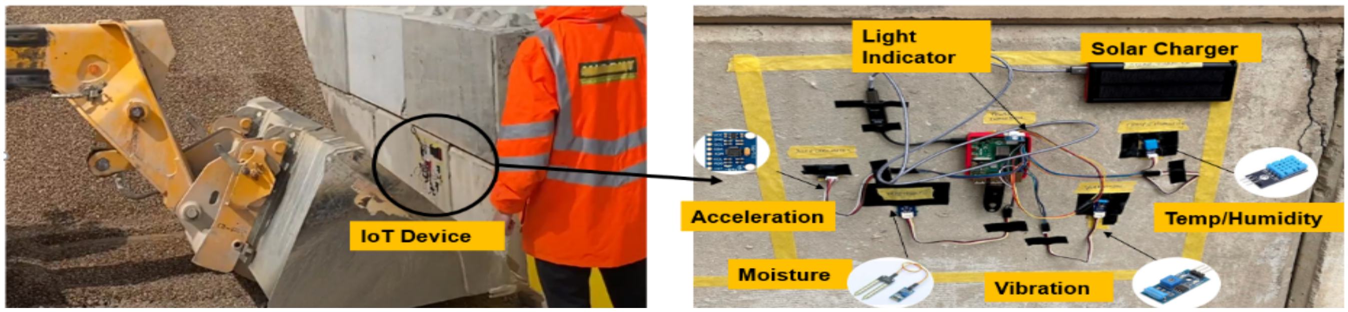

Figure 1. Testing MotionWall device against a concrete wall structure

Abstract

One enterprise that lacks a significant volume of innovation is the Structural Health industry. These systems often require an expensive per-unit cost with an average bridge device costing upwards of **$8,806** [2]. This demo attempts to demonstrate low-cost hardware and software solutions for structure monitoring, and deal with the widespread missing-value issue in the real world. The form of missing data substitution is done via machine learning and using random forest regression to predict missing values associated with sensory readings. The learning model is displayed through a front-end web application created specifically for the research paper as it allows for graph customization and specifies choices in cloud technologies.

ACM Reference Format:
Eamon Magdoubi, Terry R. Payne, and Chris Xiaoxuan Lu. 2020. MotionWall: A Cost-Effective System For Structural Health Monitoring. In *BuildSys '20: BuildSys '20: ACM Build System, November 13–14, 2020, Yokohama, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/1122445.1122456

BuildSys '20, November 13–14, 2020, Yokohama, Japan

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

1 Introduction

Structural Health Monitoring (SHM) is the process in construction development that identifies potential risks, faults and damages before they transpire [1]. Due to the scale of most SHM projects relying on bridges such as Coastal Bridge which labour and hardware would cost $29,000 for hub installation, smaller-scaled projects cannot exercise these architectures [2]. Therefore, it is an urgent necessity of using a combination of low-cost sensors and automatic softwares to attempt to accumulate supportive evidence at a budget.

Table 1. Sensors used and Purpose

Sensor Name	Purpose	Price
MPU-6050 Sensor	Acceleration in X, Y and Z Axis	£2.50
DHT11	Temperature and Humidity	£3.80
SW-420	Monitor vibration levels	£4.70
Moisture Sensor	Measure the moisture levels	£2.05

This work utilised sensors include an MPU-6050 accelerometer, and SW-420 vibration unit to emphasise displacement and mass movement through motion and stress detection without requiring extensive and complex set-ups [4]. Extra-dimensional data readings from Temperature and Humidity clarify any stiffness and mass shift in a nonlinear manner through environmental conditions [5]. The total cost base for our cloud infrastructure and IoT device is **£84.05** which is **100 times cheaper** then other case studies [2].

2 Design

System Design. Communication mechanisms and front-end user experience was a crucial factor as many current systems utilise third-party dashboards which don't allow for user customisation or unique features [3]. Data is transmitted through server-less functions as node interaction can be scaled automatically through the cloud to allow for potential nodes. The entire projects back-end and hosting were done entirely on the AWS Cloud for effective and reliable use. Content Delivery Network (CDN) managed the hosting for large global networks of node servers to achieve instantaneous load times within the five global regions that Amazon operates in. Incorporating a cloud hosting provider and the use of a Dynamic NO-SQL solution, DynamoDB acknowledges the project to be scaled for future expansion of increased nodes for a lower price than traditional methods. Fig. 2 illustrates our system workflow.

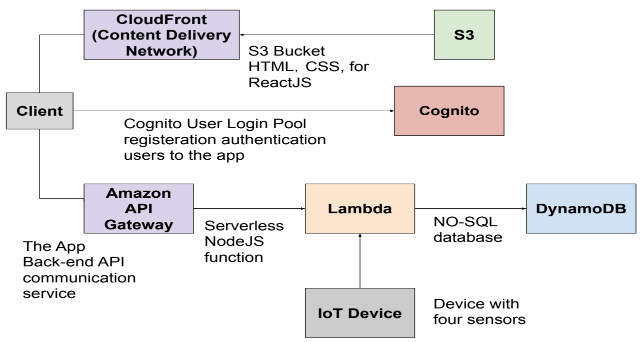

Figure 2. Entire system architecture for IoT to cloud front-end connectivity

Data Imputation Approach Design. In order to handle the missing value caused by temporary sensor malfunction or failures, machine learning algorithms are leveraged to perform data imputation on device. Due to the superior trade-off between prediction power and computation efficiency, Random Forest Regression provided a high enough precision and recall against IoT data, creating multiple pathways with high correlation for data imputation. This approach requires a training phase to generate route path estimations and does so with a **75%** training phase and **25%** testing phase comparison.

3 Implementation and Testing

The purpose of the experimentation and implementation was to identify the responsive, straightforward setup and deployment against rough exteriors. A **two-hour** time-frame was designated for testing; thus, a **JCB 550-80 digger** was brought to simulate comprehensive impacts such as hitting a surface with force (see Fig. 1). Force variation was conducted as general labour alongside the IoT device was captured through driving alongside and transporting a collection of rubble alongside the wall.

4 Initial Findings against Imputation

Each 10-second interval data was saved via the serverless cloud API created for the project for further evaluation. Using **Jupyter labs** on-board on the Raspberry Pi 4, the device can receive all the data at the end of the working day and interpolate the information against its predicted values. Vibration and Acceleration were employed in conjunction via normalised ratios to obtain four times the data in which **100,000** test scenario branches were generated.

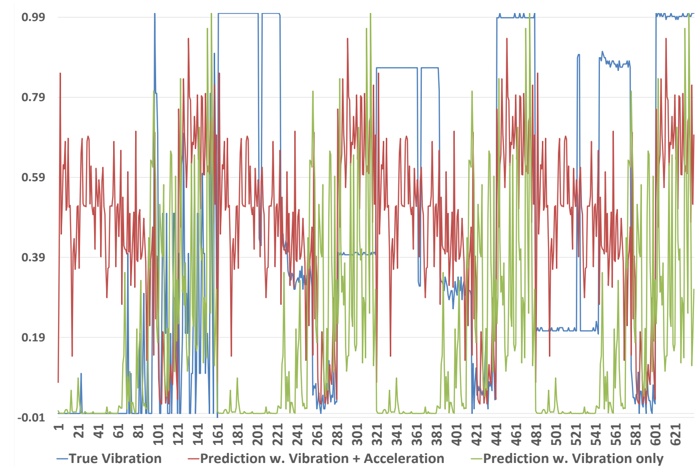

Figure 3. Comparing data through Random Forest Regression over two hour expected time period

As shown in Fig. 3, employing the value predictions with both historical vibration and accelerometer sensors, the imputation values were on average, **37%** different from the ground truth. This imputation is more effective than the **51%** accuracy by using only historical vibration values.

5 Future works

Future work will establish multiple IoT nodes working in conjunction on a single concrete testing sight to contribute together to achiever better accuracy in interpolation analysis. Resourced-constrained deep learning models for data imputation will be investigated as well.

References

[1] J.M.W Brownjohn. 2007. Structural Health Monitoring Current State and Future Trends. *Phil. Trans. R. Soc.* 365, 1851 (Feb. 2007), 589––622. https://doi.org/10.1098/rsta.2006.1925

[2] Justin R. Martinez; Duzgun Agdas, Jennifer A. Rice and Ivan R. Lasa. 2016. Comparison of Visual Inspection and Structural-Health Monitoring As Bridge Condition Assessment Methods. *Journal of Performance of Constructed Facilities* 30, 3 (June 2016), 1–10. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000802

[3] Tousiq Wiqar Zawar H. Khan Haroon Malik, Khurram S. Khattak and Ahmed B. Altamimi. 2019. Low Cost Internet of Things Platform for Structural Health Monitoring. *2019 22nd International Multitopic Conference (INMIC)* 1, 1 (Nov. 2019), 1404–1423. https://doi.org/10.1109/INMIC48123.2019.9022801

[4] Alessandro Pegoretti. 2020. Structural Health Monitoring Current State and Future Trends. *An SAE Technical Paper Compilation* 2020, 1 (June 2020), 1––144. https://doi.org/10.4271/PT-194

[5] Xu Wang Lili Dong Zengshun Chen, Xiao Zhou and Yuanhao Qian. 2017. Deployment of a Smart Structural Health Monitoring System for Long-Span Arch Bridges: A Review and a Case Study. *Sensors (Basel) Switzerland* 1, 1 (Sept. 2017), 1–21. https://doi.org/10.3390/s17092151