Real-Time Human Cognition of Nearby Vibrations Using Augmented Reality

Elijah Wyckoff, Marlan Ball, and Fernando Moreu

Abstract Human-computer interaction is an important part of structural monitoring and the research community is interested in amplifying human cognition of structural response. Augmenting feedback closes the loop between reality and human, where normally the human would need to check a computer or other separate screen for information on the real-time structural response. The research proposes using an Augmented Reality (AR) application to provide inspectors with real-time information about structural responses, to allow the user to perceive real-time changes. The proposed research seeks to validate the AR application as a useful tool in the field. This paper presents the results of an experiment that show how AR can provide a channel for direct sensor feedback. The application displays a live graph of accelerometer data with a low delay close to real time. The results of the reported experiment prove the usefulness of the AR application in the field by comparing feedback delay over a local area network versus a mobile WiFi hotspot.

Keywords Wireless sensor · Structural response · Vibration · Augmented reality · Real time

1 Introduction

It is important to researchers to monitor vibrations to ensure the safety of the nearby environment. Researchers need to track vibration levels to prevent damage to sensitive machines, but current technology does not allow for a researcher to work freely without having to check a computer monitor or use their hands [1]. Development of machine learning and deep learning algorithms in conjunction with smart sensors enables automated predictive maintenance [2]. This is enabled by IoT technology which is used for wireless sensor networks for environmental sensing and healthcare monitoring [3]. New framework has not been developed for the direct augmentation of data into the user's vision. Researchers have also examined the time delay associated with network communication by analyzing Internet communication features. They propose a control structure to overcome the transmission delay [4]. Fu demonstrates the reliability of a proposed time-delay estimation method through Internet time-delay estimation experiments [5]. This research seeks to investigate the delay in local network and a mobile hotspot for use in the field.

Researchers depend on sensors to inform on critical events especially in structural health monitoring (SHM). Smart Infrastructure wireless sensors are useful for their reliability, low-cost, low power, and fast deployment characteristics [6]. Wireless sensor networks are used for monitoring and assessing vibration risk in historical buildings and cultural sites [7]. Forming a network of wireless sensors supports the gathering of data and decision-making before, during, and after a crisis event. A wireless sensor network in Torre Aquila proved the system is an effective tool for assessing the tower stability while delivering data with loss ratios <0.01% with an estimated lifetime over 1 year [8]. Researchers developed an ad hoc wireless sensor deployment for indoor environmental quality monitoring in office buildings consisting of 19 sensor devices continuously measuring vibration and other levels [9]. Researchers reported installations of wireless sensor networks in a suspension bridge, slab bridge, rail tunnel, and water supply pipeline to manage infrastructure in a way that ensures safe and

E. Wyckoff
Department of Mechanical Engineering, University of New Mexico, Albuquerque, NM, USA
e-mail: ewyckoff13@unm.edu

M. Ball · F. Moreu (✉)
Department of Civil, Construction, and Environmental Engineering, University of New Mexico, Albuquerque, NM, USA
e-mail: mball15@unm.edu; fmoreu@unm.edu

K. Grimmelsman (ed.), *Dynamics of Civil Structures, Volume 2*, Conference Proceedings of the Society for Experimental Mechanics Series, https://doi.org/10.1007/978-3-030-77143-0_14

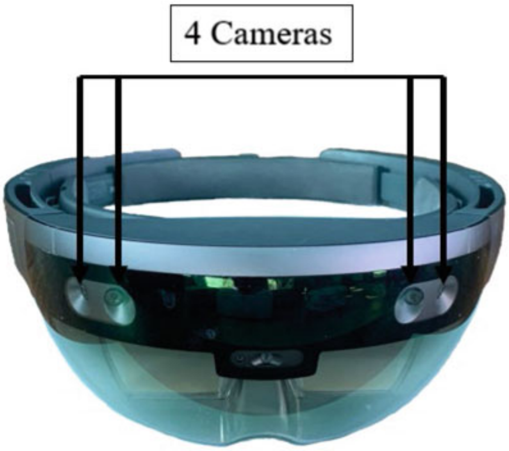

Fig. 1 AR headset—Microsoft HoloLens

efficient operation by informing data about short- and long-term performance [10]. Wireless sensors are optimal for efficient and reliable data feedback.

Augmented Reality is extremely useful to researchers in informing of real-time data. The concept has been explored in several different areas with the goal of projecting data between the image-based environment and the AR environment. Researchers at the University of New Mexico augmented displacement data, but these values were first recorded and stored in a database before they were graphed in AR [11]. Researchers at Princeton developed a human-machine interface which organizes metadata and provides actionable information by visualizing data about the built environment both on and off-site using AR [12]. Ballor et al. investigated using AR in infrastructure inspections. The framework uses the headset's sensors to capture a high-resolution 3D measurement of the infrastructure which can be used to analyze the state of the structure over time, and track damage progression [13]. Researchers at Los Alamos National Laboratory used AR for structural health monitoring including detecting heat emitted from electronic equipment [14]. AR has a wide range of uses making it a valuable tool for SHM.

The new interface has been explored in the domain of structural design since it is now possible to interface the structural responses with holograms and other models permitting the engineer or technician to quantify structural dynamics in the augmented interface. The interface includes a LEWIS5 (Low-cost Efficient Wireless Intelligent Sensor) which is an Arduino Metro M4 microcontroller equipped with an accelerometer to measure vibrations wirelessly. The sensor requires a power source, which is connected via micro-USB. This data is sent over WiFi using TCP connection to the Microsoft HoloLens AR headset, where acceleration values are plotted real time in the user's field of view. A comprehensive menu is included in the user-interface. This work is innovative in the area of human-structures interfaces, and it enables a new close to real-time control of structures and sensing of dynamics. This research validates the application as a useful tool in the field by comparing the time delay of the graph on a local area network versus a mobile WiFi hotspot.

2 Background

Augmented reality blends interactive digital elements with a real-world environment. An environment is generated by a computer and super-imposed onto the real-world environment, which allows the user to interact with the merged environment. The platform used in this research is optical see-through display, the Microsoft HoloLens. The head-mounted display (HMD) allows for contact-free operation by hand gestures and voice commands. The headset collects data via its image sensors. It also contains an inertial measurement unit, four sensors, an ambient light sensor, a depth camera, a photographic video camera of 2.4-megapixels, and four microphones [15]. The HoloLens has four light-sensitive grayscale cameras labeled in Fig. 1 which track visual features.

The sensing platform is developed to read acceleration data in a triaxial coordinate system as a wireless Structural Health Monitoring (SHM) system. This is done with a Low-cost Efficient Wireless Intelligent Sensor, abbreviated as LEWIS5. The sensor connects via WiFi but requires a power source hooked up via micro-USB. The physical components of the fully assembled sensor are labeled in Fig. 2.

The Metro M4 Express is a 32-bit microcontroller with the ATSAMD51 microchip [16]. The Cortex M4 core runs at 120 MHz with floating point support. The board is powered via micro-USB or barrel jack connection. The Airlift WiFi

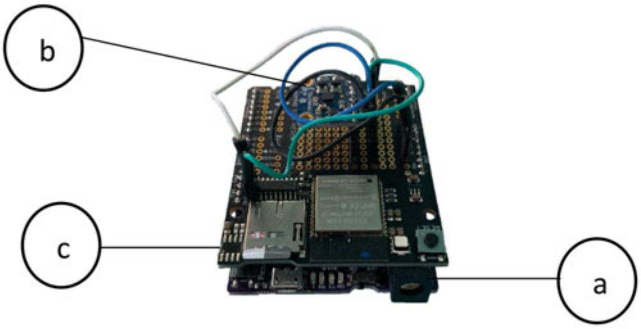

Fig. 2 LEWIS5 sensor full assembly; (a) Metro M4 express; (b) MMA8451 accelerometer; (c) Airlift WiFi shield

Shield allows the use of the ESP32 chip as a WiFi co-processor [17]. The Metro M4 microcontroller does not have WiFi built in, so the addition of the shield permits WiFi network connection and data transfer from websites as well as the sending of socket-based commands. The triple-axis accelerometer used for this project is the high-precision MMA8451 with a 14-bit Analog-to-digital converter [18]. Digital accelerometers are used to detect motion, tilt, and basic orientation. For the purpose of this project the accelerometer is used to detect motion as well as vibrations. Its usage range varies from ±2G up to ±8G, which is ideal for its application to this project.

Programming and development of the AR application is done in Unity 2018.4.19f1 taking advantage of the Mixed Reality Toolkit (MRTK) from Microsoft. The application is developed for Universal Windows Platform (UWP) which allows deployment to the HoloLens. The programming platform is Visual Studio 2019, and the code is written in C#. The graph of the live data is developed as a scatter plot, which was chosen as the most effective and efficient solution. The graph is developed based on a tutorial from Catlike Coding [19]. Points at each appropriate coordinate are generated by Unity's default cube game object based on the incoming parsed accelerometer data. Each data point is graphed as a small 3D cube for visual feedback, and the cubes are connected with a Unity LineRenderer so that the graph appears as a line chart. The graph updates with each frame, meaning the cubes are adjusted as time progresses, defined by the function $f(x, t)$. The points are positioned using an Update method with a for loop. A custom matte shader was made for each cube using Unity's material assets. Red designates the x acceleration, yellow designates the y acceleration, and cyan designates the z acceleration.

The application interface consists of six different buttons with specific functionality. From top to bottom: Client Start connects the client to the server via Transmission Control Protocol (TCP). In the context of the application, the computer running the Arduino program acts as the server and the device running the AR application is the client. The Unity code requires the IP Address of the Arduino board, and the Unity code and Arduino code are set up on the same port. Client Stop closes the client connection to the server. The View Start button initiates the Unity function ContinueInput. Incoming data from the server is parsed into x y and z vectors converted to terms of the gravitational constant G, which is graphed on the left side of the interface. The x and y data are also offset so that the x line does overlap and hide the y line. Therefore, the graph axis is labeled as "Z Acc" for the purpose of the experiment as well as simplicity. Included below the graph is a horizontal line indicating the time in seconds of the data. View Stop zeros out the three data lines but does not disconnect the client. The view may be resumed by selecting View Start again. The purpose of the slider at the bottom is a visual aid. When the data breaches the desired threshold value, the line flashes pink as a warning to the user (Fig. 3).

The objective of the experiment was to prove the validity of the real-time acceleration graph and to investigate and verify the minimal delay in the data sending, receiving, and plotting over WiFi on a network and mobile hotspot compared to real time. In the experiment, the researchers used the Microsoft HoloLens 1 AR headset and a Model K2007E01 "SmartShaker," which is an electrodynamic exciter from The Modal Shop. The AR headset was used to augment live acceleration data sent directly from the LEWIS5 sensor. This data was controlled by a set frequency input to the shaker. Therefore, the AR graph could be validated by the known input. Additionally, a video capture allowed for the investigation of the graph delay. This was done with simple frame-by-frame video analysis, which returned an approximate value for the delay in the graph for all two different WiFi connections. The experiment was conducted in a researcher's home office due to restrictions imposed by the global pandemic. The LEWIS5 sensor is positioned horizontal to the ground with the accelerometer z axis matching gravitational acceleration. The sensor is attached to the top of the electrodynamic exciter for controlled input and is connected with a micro-USB cable to a laptop computer as a power source (Fig. 4).

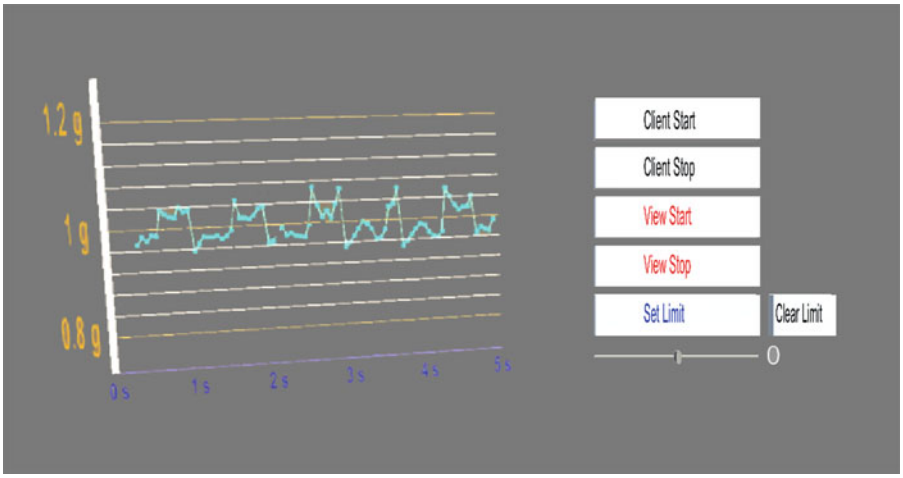

Fig. 3 Application interface as seen in the user's view

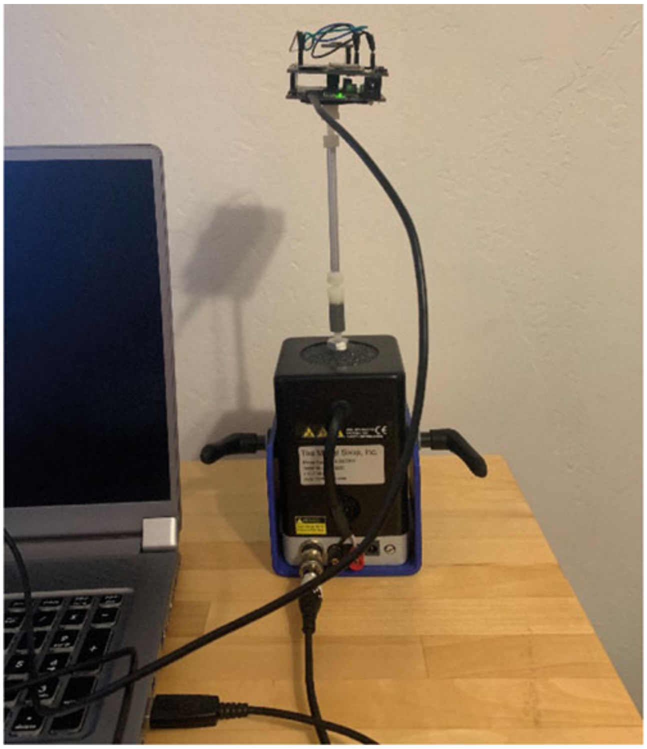

Fig. 4 Experimental setup—sensor-shaker configuration

The experiment was conducted following this procedure:

1. Open Arduino IDE and connect LEWIS5 to power source.
2. Attach the LEWIS5 sensor and stinger pairing to the mounting insert on the SmartShaker.
3. Press reset on the sensor once to load the program.
4. Open Serial Monitor and wait for the server to connect to WiFi.
5. Initiate the SmartShaker by selecting a 10 Hz input in a smart phone frequency generator application, which is connected to the shaker via 3.5 mm jack.
6. Once message "Listening for clients on port" is initiated, the Client Start button is selected in the AR application.
7. When the client connects, the Serial Monitor will begin auto scrolling and displaying the three columns of acceleration data, at which time View Start is selected.

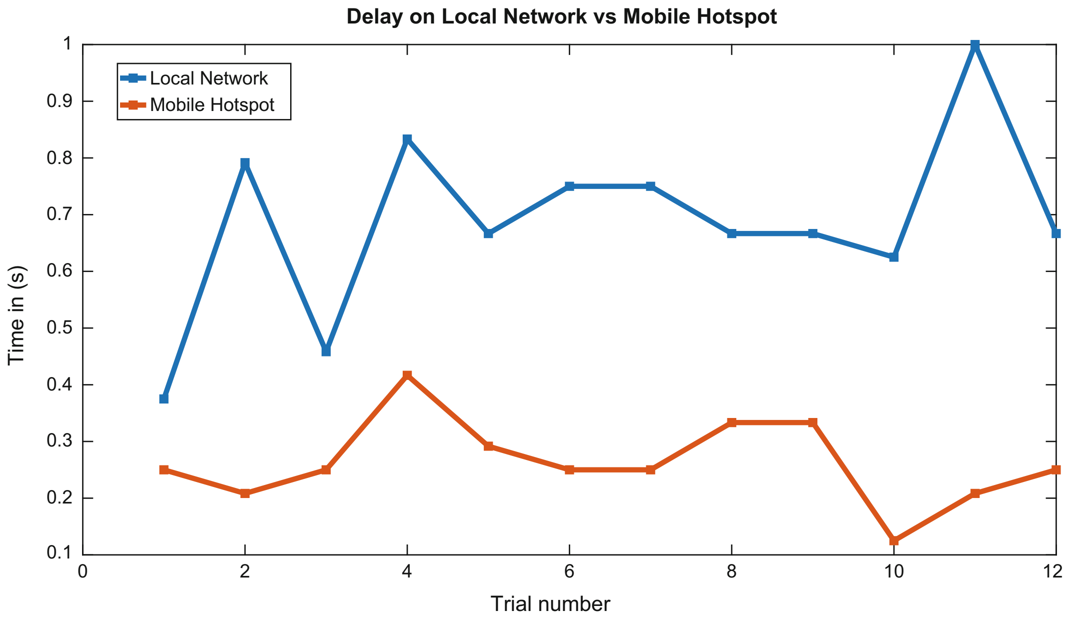

Fig. 5 Delay on local network vs. mobile hotspot

8. The three colored lines are plotted on the graph in the user's view.
9. View Stop and Client Stop are selected to close the graph and shut down the client's connection to the server.
10. The visual feedback from the graph is recorded in the HoloLens as a video to validate the results and analyze the delay.
11. The time-delay data and Arduino data are plotted in MATLAB.

The time delay in the program was investigated using video analysis. By counting the frames between the initial sensor acceleration and the recorded response, the time delay can be approximated. With the known value of the video framerate in FPS, the time delay of the application can be calculated using the following formula.

$$(\text{Frame}_1 - \text{Frame}_0) * \frac{1}{\text{Video FPS}} = \text{Time Delay} \tag{1}$$

3 Test Results and Analyses

The test was run with a total of three trials with four excitations for each WiFi connection. Between each trial the sever-client connection was shut down and the AR application was restarted. Therefore, the graph displayed a total of 12 positive peaks which could be analyzed to determine the time delay. Frame_0 was designated as the time of initial acceleration upwards by the sensor. Thus, Frame_1 designates the response plotted on the graph in the user's view. The difference in frames was recorded and the time delay was calculated with Eq. (1). The results were plotted in MATLAB for comparison in Fig. 5 and the individual values are reported in Table 1.

The results of the experiment validate the AR application for use in the field. As proven clearly in the graph, the time delay on the local network was inconsistent and slow. The time delay in the graph response on the mobile WiFi network is consistently under 0.5 s, proving the reliability of the application for work in the field. The average time delay in the graph on the mobile network is much closer to real time at 0.263 s versus 0.6875 s on the local network. Additionally, the delay on the local network peaks at one full second while the delay peaks at less than 0.5 s on the mobile network.

Table 1 Experimental results

Trial	LAN delay in (s)	Mobile delay in (s)
1	0.3750	0.2500
2	0.7917	0.2083
3	0.4583	0.2500
4	0.8333	0.4167
5	0.6667	0.2917
6	0.7500	0.2500
7	0.7500	0.2500
8	0.6667	0.3333
9	0.6667	0.3333
10	0.6250	0.2500
11	1	0.2083
12	0.6667	0.2500
Average	0.6875	0.2630

4 Conclusions

This paper developed and tested an augmented reality application for live sensor feedback. An analysis of the delay in the acceleration graph shown in AR was conducted and the graph was verified by a known vibration frequency input to the sensor. Researchers demonstrate the reliability of the AR application in graphing acceleration data close to real time. This emphasizes the importance of accurate measurements and transmission consistency. The variance in time delay on the local network makes for a less reliable mode of data transmission in the AR application, whereas the connection on the mobile hotspot relays and graphs vibration data close to real time. The research helps to understand the reliability of several WiFi network options in close to real-time acceleration data transmission while developing an AR tool for data feedback and perception for use in the field.

References

1. Karlsson, M., Hörnqvist, F.: Robot condition monitoring and production simulation. Ph.D. dissertation, Dept. Civilingenjör, Maskinteknik, Luleå Tekniska Univ., Institutionen för teknikvetenskap och matematik, Luleå, Sweden (2018)
2. Namuduri, S., Narayanan, B.N., Davuluru, V.S., Burton, L., Bhansali, S.: Review—Deep learning methods for sensor based predictive maintenance and future perspectives for electrochemical sensors. J. Electrochem. Soc. **167**(3), 037552 (2020). https://doi.org/10.1149/1945-7111/ab67a8
3. Raj, A., Steingart, D.: Review—Power sources for the Internet of Things. J. Electrochem. Soc. **165**(8) (2018). https://doi.org/10.1149/2.0181808jes
4. Yang, S.H., Chen, X., Tan, L.S., Yang, L.: Time delay and data loss compensation for Internet-based process control systems. Trans. Inst. Meas. Control. **27**(2), 103–118 (2005). https://doi.org/10.1191/0142331205tm133oa
5. Fu, X., Fu, X.: A predictive algorithm for time delay Internet network. In: 2011 International Conference on Electronics, Communications and Control (ICECC), Ningbo, pp. 666–669 (2011). https://doi.org/10.1109/ICECC.2011.6066402
6. Morimoto, R.: A socio-economic analysis of Smart Infrastructure sensor technology. Transp. Res. Part C Emerg. Technol. **31**, 18–29 (2013). https://doi.org/10.1016/j.trc.2013.02.015
7. Abruzzese, D., Angelaccio, M., Giuliano, R., Miccoli, L., Vari, A.: Monitoring and vibration risk assessment in cultural heritage via Wireless Sensors Network. In: 2009 2nd Conference on Human System Interactions (2009). https://doi.org/10.1109/hsi.2009.5091040
8. Ceriotti, M., et al.: Monitoring heritage buildings with wireless sensor networks: the Torre Aquila deployment. In: 2009 International Conference on Information Processing in Sensor Networks, San Francisco, CA, pp. 277–288 (2009)
9. Brunelli, D., Minakov, I., Passerone, R., Rossi, M.: POVOMON: an Ad-hoc Wireless Sensor Network for indoor environmental monitoring. In: 2014 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems Proceedings, Naples, pp. 1–6 (2014). https://doi.org/10.1109/EESMS.2014.6923287
10. Bennett, P., Stoianov, I., Fidler, P., Maksimovic, C., Middleton, C., Graham, N., Soga, K., Hoult, N.: Wireless sensor networks: creating 'smart infrastructure'. Inst. Civil Eng. **162**, 136–143 (2009)
11. Aguero, M., Maharjan, D., Rodriguez, M.D., Mascarenas, D.D., Moreu, F.: Design and implementation of a connection between augmented reality and sensors. Robotics. **9**(1), 3 (2020). https://doi.org/10.3390/robotics9010003
12. Napolitano, R., Liu, Z., Sun, C., Glisic, B.: Combination of image-based documentation and augmented reality for structural health monitoring and building pathology. Front. Built Environ. **5** (2019). https://doi.org/10.3389/fbuil.2019.00050
13. Ballor, J.P., et al.: Augmented reality for next generation infrastructure inspections. In: Barthorpe, R. (ed.) Model Validation and Uncertainty Quantification Conference Proceedings of the Society for Experimental Mechanics Series, vol. 3. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-74793-4_23

14. Morales Garcia, J.E., Gertsen, H.J., Liao, A.S.N., Mascarenas, D.D.L.: Augmented reality for smart infrastructure inspection. Technical report, Los Alamos National Lab, (LANL), Los Alamos, NM, USA (2017)
15. Holmdahl, T.B.: A closer look at the Microsoft HoloLens hardware. Microsoft Devices Blog (2015)
16. Industries, A.: Adafruit Metro M4 feat. Microchip ATSAMD51. https://www.adafruit.com/product/3382 (n.d.)
17. Industries, A.: Adafruit AirLift Shield - ESP32 WiFi Co-Processor. https://www.adafruit.com/product/4285 (n.d.)
18. Industries, A.: Adafruit triple-axis accelerometer - ±2/4/8g @ 14-bit - MMA8451. https://www.adafruit.com/product/2019 (n.d.)
19. Flick, J.: Building a graph. https://catlikecoding.com/unity/tutorials/basics/building-a-graph/ (2020)

Elijah Wyckoff is a graduate research assistant in the Department of Mechanical Engineering at the University of New Mexico, pursuing a MS. His BS is from Arizona State University in the Department of Mechanical Engineering (2020).

Marlan Ball is a research assistant in the Department of Civil, Construction and Environmental Engineering at the University of New Mexico. His BS is from LeTourneau University in Electrical Engineering Technology.

Fernando Moreu is an assistant professor in the Department of Civil, Construction and Environmental Engineering of the University of New Mexico, with courtesy appointments in the Department of Mechanical Engineering and the Department of Electrical and Computing Engineering. His PhD is from the University of Illinois at Urbana-Champaign in Structural Engineering (2015).