MATLABArduino.org: An Open-source Website and YouTube channel for Embedded Systems Education

Pramod Abichandani*, Deepan Lobo, Chris Berry, Vaishali Parikh, William Fligor, William McIntyre

***Abstract*—MATLABArduino.org and its associated YouTube channel were created to provide free-of-cost, high-definition YouTube videos that demonstrate the use of two powerful technologies - MATLAB and Arduino - to perform data-driven, embedded systems tasks for embedded systems education. This initiative aims to broaden the participation of students in engineering through the use of low-cost hardware, real-time visualizations of otherwise abstract concepts, and easy-to-follow instructional videos. This paper describes the video content developed as part of this educational initiative in detail. Viewership trends of the YouTube channel are reported to describe insights about the video engagement and viewer demographics. A qualitative analysis of the YouTube comments left by the viewers is provided, along with highlighted comments that show recurring themes. Finally, challenges encountered in operating this website and YouTube channel in the context of remote online education for embedded systems are described. As physical computing-based online education becomes increasingly pervasive, we anticipate that open educational resources similar to MATLABArduino.org will drive a significant amount of learning for engineering students. The insights presented here may guide current and future open educational initiatives for embedded systems remote learning.**

***Index Terms*—Embedded Systems, Engineering Education, Open Educational Resources, MATLAB, Arduino, Distance Learning**

I. Introduction

The ubiquitous nature of Internet-of-Things (IoT) has generated massive growth for the embedded systems market due to the speedy development of the connected devices [1]. Such growth has created a need to understand the underlying embedded technology that drives sensing and data-driven decision-making [2], [3]. `MATLABArduino.org` is a website dedicated to providing technically rigorous, free of cost, real-time embedded systems, and associated sensor data analytics education [4]. MATLAB, developed by MathWorks, is a scientific software with an extensive toolset used for signal processing, numerical data analysis, modeling, programming, simulation, and computer graphic visualization [5], [6]. Launched in 1985, MATLAB is perhaps the most widely used scientific software for solving critical scientific and engineering problems. The Arduino UNO board and the Arduino IDE software platform are used for rapid prototyping of embedded systems [7]. The Arduino boards feature a microcontroller and associated electronics, making them capable of interfacing with input (sensors) and output (motion, lights, sound) devices. The true power of Arduino is embodied in its low-cost (boards start at approximately $5), open-source hardware design, and the millions of lines of open-source code that can be leveraged to perform fast sensor data acquisition and processing [8], [9].

The authors are with the Robotics and Data Laboratory (RADLab) at the New Jersey Institute of Technology, Newark, NJ 07102, USA. Email: pva23@njit.edu

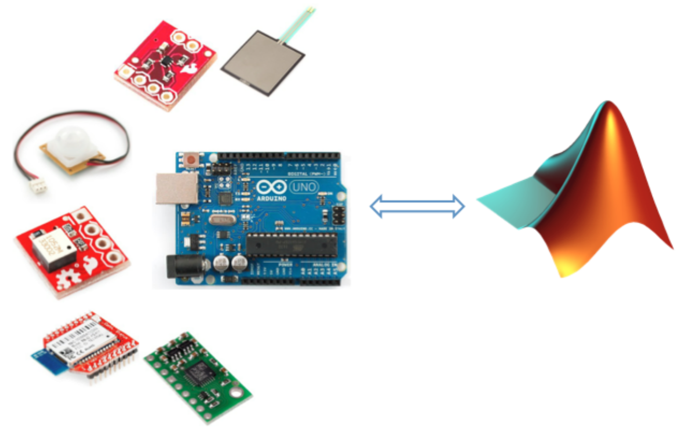

Fig. 1: The website features videos that depict the use of MATLAB and Arduino to perform real-time data-driven tasks

`MATLABArduino.org` combines these two powerful technologies to leverage their strengths: The Arduino board provides a simple and inexpensive method of electronically connecting with a plethora of devices and capture real-world data as and when it is generated (Figure 1). On the other hand, MATLAB's computational abilities and the plethora of sophisticated scientific software libraries allow users to analyze and visualize this real-world data and convert it into information that is rich with actionable insights and qualitative acumen.

The learning content provided by `MATLABArduino.org` is in the form of YouTube videos. YouTube is the most widely available free video streaming service that reaches 185+ countries [10]. These high-definition videos demonstrate the use of MATLAB and Arduino platforms to interface with wired/wireless embedded systems for compelling sensor-based real-time applications. The unifying theme for the tasks featured in the videos is procuring, analyzing, and visualizing real-world data to generate real-time insights that facilitate data-driven decision-making. The `MATLABArduino` YouTube channel has gained viewership from 185 countries with over 5900 subscribers at the time of writing this paper.

II. Related Works

There is a growing community of educators that are leveraging the benefits of MATLAB and Arduino and developing curricula to cover engineering concepts such as control systems, robotics, and computer science [11]–[14]. The popularity and accessibility of the Arduino and MATLAB platforms also make them ideal tools in developing an online curriculum [15]. Blended courses that use open-source video content hosted online are seeing widespread adoption across universities [16]. Additionally, the COVID-19 pandemic caused a dramatic change in education and made distance learning one of the few options for educational institutions to continue their operations [17].

In [15], the authors presented a redesigned embedded systems course to help students gain skills to use microcontrollers with specialized tools. The lecture-laboratory course was conducted as a 3-credit course for electronics and mechatronics engineering students at *Universidad Tecnológica de Bolívar*, Cartagena, Colombia. The authors noted that open-source toolchains such as Arduino worked well since the students have ample support from online resources. In [11], the authors developed and validated a novel real-time educational platform based on the MATLAB/Simulink package and the LEGO EV3 brick for educational use in robotics and computer science. The platform was used to evaluate different robot path planning algorithms. The motivation of the work was to develop a platform with features applicable across multiple engineering domains. In [18], the authors documented an e-kit for a virtual lab that used Tinkercad's web-based simulator for Arduino circuits. The virtual lab was evaluated with a group of computer science students. Student evaluations noted that the e-kit could be introduced to other related subjects to enhance the learning experience. In [17], the authors presented a distance learning course for embedded systems using an active learning methodology. The instruction medium for active, lecture-free learning was established using a learning management system, YouTube, various web resources, an Arduino hardware kit, and a software development environment. Results from student assessment and feedback surveys indicate strong student engagement and positive perceptions.

This paper presents a unique and novel extension to the above body of work by reporting on the following contributions:

1) A comprehensive catalog of an expansive set of technically rigorous, free-of-cost, video content that covers in detail embedded sensors, wireless technologies, and unique 2D/3D real-time visualizations using the Arduino and MATLAB platform. This content is proven to serve as a powerful resource or roadmap into the world of embedded systems and IoT for undergraduate or graduate STEM students.
2) Qualitative and quantitative analysis of viewership trends and viewer engagement to guide anyone looking to create and maintain a globally accessible free-of-cost educational website.

III. MATLABArduino.org Video Content

Videos form the primary mode of content delivery for `MATLABArduino.org`. As shown in Figure 2, embedded systems infer the world around them using sensors and provide

Fig. 2: The core tasks involved in Embedded Systems driven applications.

the ability for computing systems to analyze and visualize data. The data analysis is used to generate insights and conclusions and report them effectively to the end-user. As such, the videos emphasize sensing through a thematic categorization based on application type. Additionally, modern embedded systems leverage wireless technologies such as Bluetooth, ZigBee, and GSM. These wireless technologies are covered in the videos using popular chipsets. Intuitive and coherent 2D and 3D visualizations of static and real-time data are presented to aid students in understanding sensor measurements and computational concepts. The videos demonstrate select code snippets and explain how the code works.

A. Video Topics

The content covered by the videos is thematically organized in the following categories:

1) Serial Communication: The first video covers the steps involved to establish a serial connection between the Arduino UNO board and MATLAB using the USB port for Windows and Mac computers [19]. There are multiple ways of achieving serial communication between an Arduino and MATLAB, and this video shows the most fundamental method of transmitting handshake characters between the UNO and MATLAB. Alternatively, users can leverage MATLAB's inbuilt Arduino support packages to establish the serial communication [20].

2) Visualizations: A total of 3 videos covered the fundamentals of creating real-time visualizations based on sensor data using MATLAB's 2D and 3D visualization capabilities. As shown in Figure 3, the 2D video created animations for 2D objects using the `fill()` function based on fundamental transformations of translations and scaling. For 3D visualization, a gyroscope is used to affect the motion of a MATLAB 3D object in real-time. These videos provides an all-encompassing guide to perform data visualization in MAT-

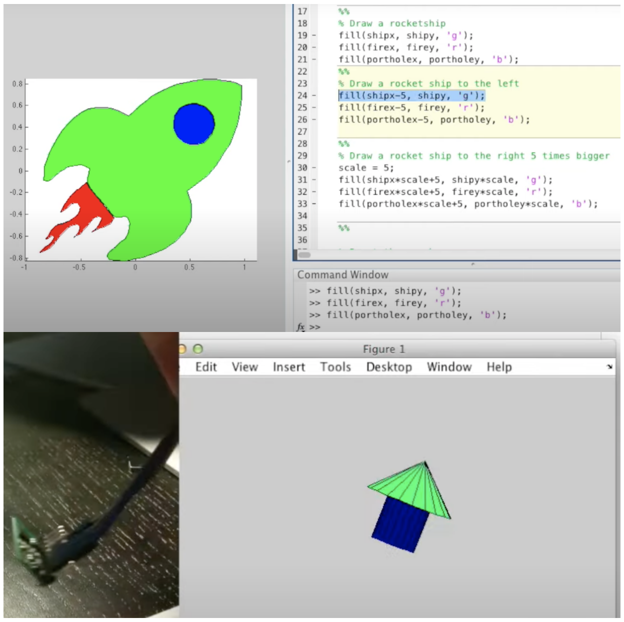

Fig. 3: (Top) 2D visualizations and (Bottom) 3D visualizations focussed on scaling, translation, and rotation operations.

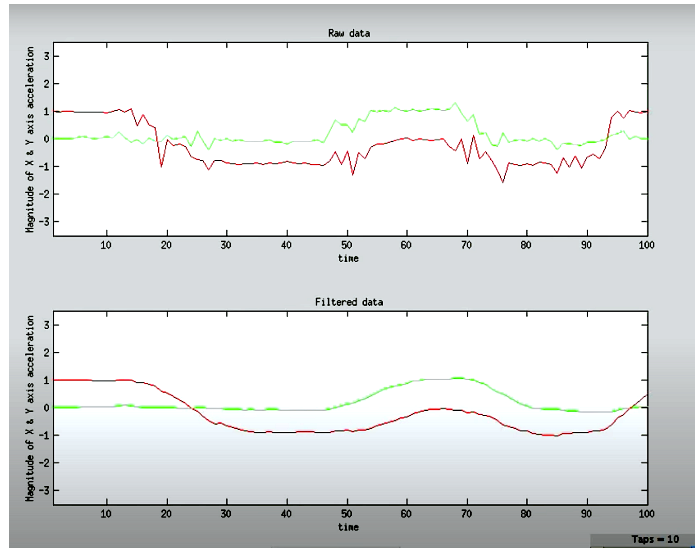

Fig. 5: The simple moving average (SMA) filtering can be easily observed by the smoothness of the data shown in the bottom subplot compared to the raw and unfiltered Accelerometer magnitude data shown in the top sub-plot.

LAB using `hgtransform()`, and `hfill()` commands to perform 2D and 3D rotation, translation, and scaling.

3) Inertial Sensors (Accelerometers, Gyroscopes, and Magnetometers): These videos were perhaps the most significant contribution of `MATLABArduino.org` to the embedded systems community. Before these videos, there was a lack of video material covering real-time inertial sensor data acquisition, calibration, noise filtering, and visualization. These videos covered in-depth the use of 3-axes accelerometers, 3-axes gyroscopes, and 3-axes magnetometers.

The videos provide a step-by-step guide to calibrate the sensors and visualize the data. Calibration of inertial sensors is a critical step to using them correctly. Figure 4 depicts the different methods used to calibrate these sensors. Of note is the use of servo motors to calibrate the gyroscope sensor.

Once the sensor calibration is complete, data is procured in real-time and visualized in 2D and 3D settings. For the accelerometer, the data is visualized as acceleration vectors and magnitudes plots. For the gyroscope, angular speeds measured in radians/sec along the three axes are used to move a 3D object. Finally, for the magnetometer, Earth's magnetic field as measured in the units of Gauss is displayed.

4) Signal Processing: Signal processing operations such as filtering noise out of sensor data and detecting threshold crossing are necessary steps when working with a real-time

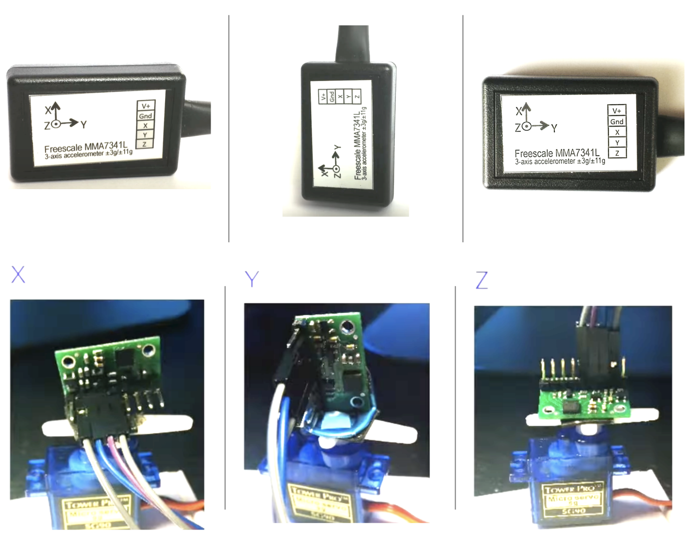

Fig. 4: Calibration of the accelerometer and gyroscopes as demonstrated in the videos.

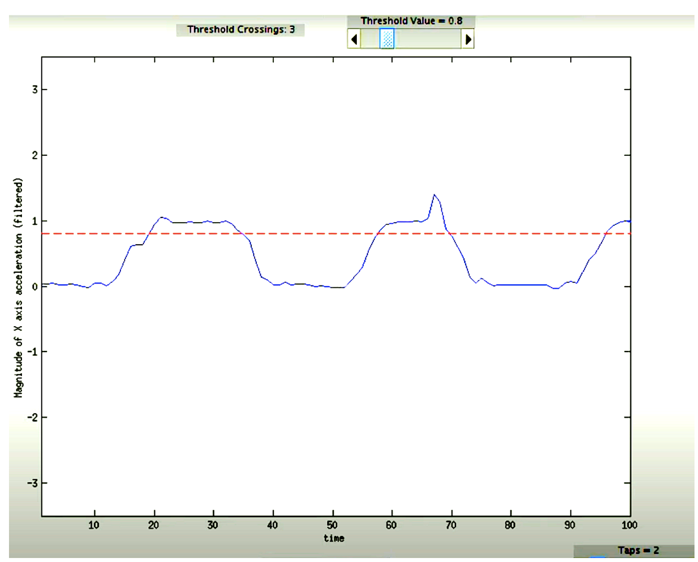

Fig. 6: Filtered data (in this case Accelerometer magnitude measurements) can be further processed to calculate the number of times it crosses a threshold.

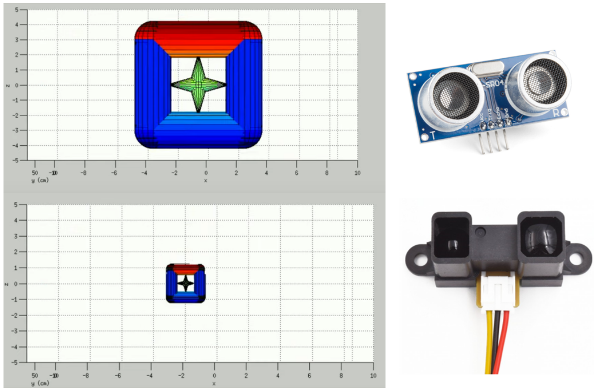

Fig. 7: (left) 3D animation used to elucidate the working of distance/proximity sensors. The 3D animated object translates and scales proportional to the distance measurements obtained from the respective distance sensors. (Top-right) Ultrasonic sensor, and (Bottom-right) Infrared sensor.

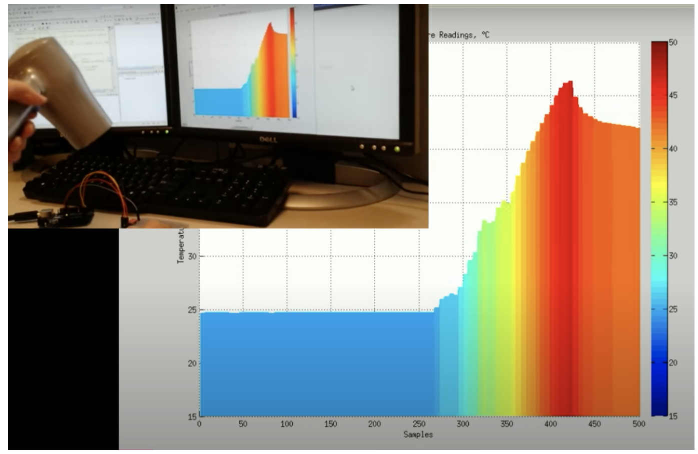

Fig. 9: Temperature sensor data visualization using a dynamic heat-map. Heat was applied to the sensor using a blow dryer as seen in the top left insert.

sensor-based system. In these videos, MATLAB is used to filter noise from the measurements received from a 3-axis accelerometer in real-time. The filtering techniques covered were Exponential Moving Average (EMA, low pass, Infinite Impulse Response - IIR) and Simple Moving Average (SMA, Finite Impulse Response - FIR) filters. Once the data is filtered, the number of times the data crosses a threshold value is counted. Different behaviors of the filters based on the relevant parameter values were studied and visualized as depicted in Figure 5. In addition, threshold crossing examples were also visualized using the filtered data as depicted in Figure 6.

5) Proximity Sensors: These videos cover ultrasound and infrared proximity sensors. The 3-part video collection demonstrates the steps involved in connecting the respective sensors to the Arduino UNO, collecting the sensor measurements, and visualizing it using a real-time 3D animation as depicted in Figure 7.

6) Force and Environmental Sensors: Force Sensitive Resistors (FSR) are made of polymer materials whose resistance changes when a force is applied. The FSR video demonstrates how to calibrate it using the Arduino UNO board and MATLAB. Additionally, the video demonstrates how to obtain data from the FSR and visualize it using 3D animations. The example uses deadweights to elucidate the working of the FSR as depicted in Figure 8.

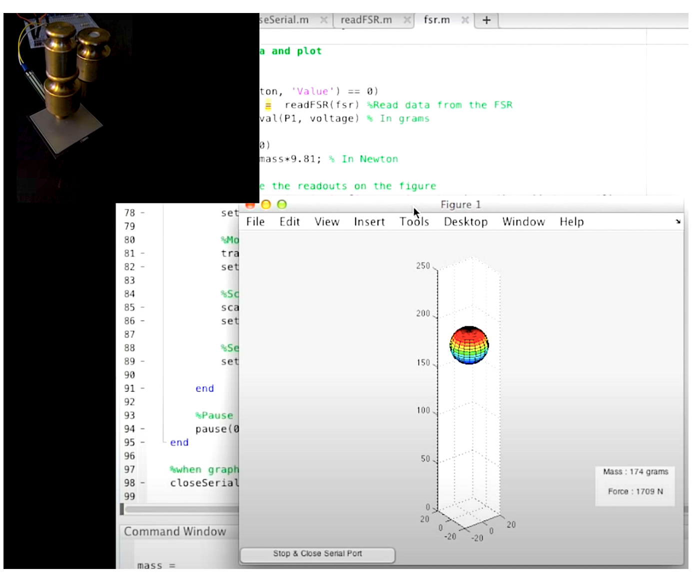

Fig. 8: 3D animation used for the FSR translates and scales a 3D object based on the amount of deadweight placed on the FSR. Deadweights are shown in the top left insert.

Another video demonstrates data acquisition and visualization of a commonly used temperature sensor as depicted in Figure 9. The dynamic heat-map visualization illustrates an effective method to use 2D plots in MATLAB. The video also covers the concept of quantization through the dynamic heat map, given that the temperature sensor is analog.

7) GPS: The Global Positioning System (GPS) forms the backbone of logistics, transportations, and military operations. This video collection demonstrates how to log GPS data in real-time using the Arduino UNO board and MATLAB. The Arduino logs real-time latitude and longitude data using the USGlobalSat EM-406A GPS Module along with the time-stamps. A second video demonstrates how to display geo-tagged temperature sensor data on a Google Maps interface as depicted in Figure 10.

8) Wireless Networking Technologies: This module demonstrates the use of chipsets based on the popular Zigbee and GSM wireless communication standards [21]. The first part of the video describes how to use MATLAB to transmit and receive data from a 3-axis accelerometer between a pair of Xbee radios. The second part of the video demonstrates how to use a GSM/GPRS modem that uses a sim card to send text messages (SMS) containing temperature sensor data in real-time.

9) Internet of Things: Internet-of-Things (IoT) integrates various devices into a network to share data and generate information. IoT devices are usually managed via web interfaces. The final video collection of `MATLABArduino.org` consists of 3 videos providing an in-depth view into the

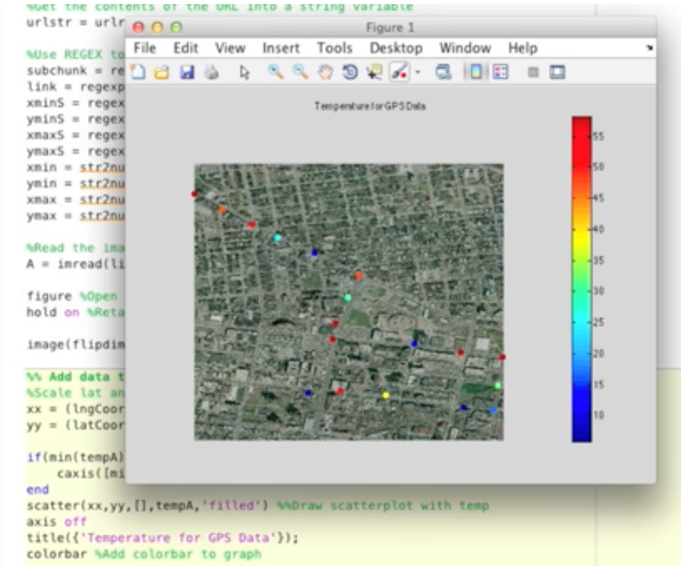

Fig. 10: Geo-tagged temperature sensor data as visualized in a Google Maps interface using MATLAB.

technology behind IoT. The videos guide the students in setting up a local web server on a Raspberry Pi to manage data wirelessly received using an Xbee radio. The data is transmitted from an Arduino board that also has an Xbee radio [22]. Finally, websockets programming is demonstrated to efficiently manage data on the server-side for client requests [23].

IV. In-class Implementation

The website formed an essential resource for engineering classes at several universities across the world. This section provides information about two classes conducted at a research university in the United States that used this website as part of the curriculum.

A. Introduction to 2D/3D Visualizations for ENGR 101/102

The website was used in an introductory computational problem solving and programming course sequence (ENGR 101/102) attended by 1200+ first-year engineering students. In this class, students used MATLAB to perform mathematical modeling and simulations with 2D and 3D animations. The students created animations that included translation, rotation, and scaling of geometric objects following specific trajectories.

Looking at the seasonality in the viewership of one of the videos as depicted in Fig. 11, a clear and sharp viewership growth is observed when the first class deliverables are due, followed by a sharp decrease in viewership. The viewership picks up again during the second class deliverable, indicating that the students used the video as a resource for their classwork.

B. Evaluation and Presentation of Experimental Data - Engineering Data Analytics ENGR 201/202

The website formed the backbone of an engineering data analytics class for 800+ second-year engineering students. The

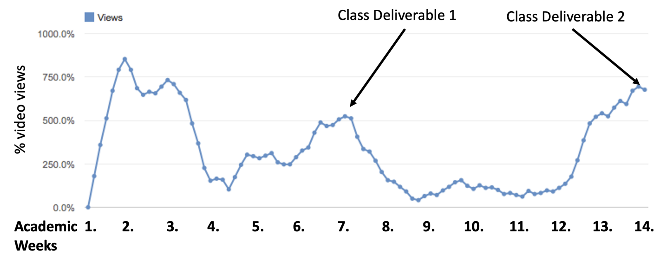

Fig. 11: Seasonality in viewership for the video discussing 2D animations indicates a growth just before the class deliverables are due followed by a dip.

class featured an active-learning-based instructional framework wherein the students worked on a final project that focused on creating real-world apps using MATLAB and a 3-axis accelerometer. The course was broken down into smaller hands-on exercises involving a 3-axis accelerometer interfaced via the Arduino UNO board. These bi-weekly exercises immersed students in the technical challenges of procuring 3-axis accelerometer data in real-time, filtering noise out the data, and applying appropriate signal processing techniques to perform data-driven decision-making. The class used five videos from the website that depicted all the necessary steps to perform the tasks in this class.

Students in this class provided feedback that the idea of breaking the videos down into smaller and focused parts helped them finish their work more effectively. The students also appreciated how smaller portions of their work culminated towards a more extensive and fully functional project.

V. Trends and Discussion

Since the first video was published on YouTube on April 28, 2013, the website and the YouTube channel have witnessed significant activity.

A. Viewership Trends

The following discussions report the analytics obtained from the YouTube channel for the viewership geographies, device accessibility, gender demographics, and top user traffic sources.

1) Country-wise percentage viewership and watch time: From the beginning of the channel until writing this, the channel has garnered $600,000$+ views. Table I, depicts the top 10 countries by percentage of views from over 155 countries to date. The United States is at the top with almost 15.7% of the total viewership. Viewers from India form the second-highest percentage with 9.2%. Other countries feature a healthy mix from Europe, Asia, and South America. Table I also ranks the top 10 countries by the percentage of watched time in hours. It is observed that viewers from the United States had the highest percentage of watch time hours (17.9%), indicating higher engagement with viewers from the United States. The higher engagement from the viewers in the United States is correlated to the percentage of viewers. We note that a cause of

TABLE I: Top 10 countries ranked by viewership metrics

Country	percentage of views	percentage watch time (hours)
United States	15.7%	17.9%
India	9.2%	9.1%
Germany	3.7%	3.8%
United Kingdom	2.2%	2.5%
Turkey	1.7%	1.4%
Brazil	1.5%	1.6%
Mexico	1.2%	1.2%
France	1.1%	1.1%
Malaysia	1.0%	1.1%
Canada	1.0%	1.2%

concern is the lack of viewership from nations in the African sub-continent.

2) Channel Accessibility: Fig. 12 shows the distribution of different computing devices used to access the videos. It is observed that 87.2% of the viewers accessed the content from their computers, 9.3% used mobile phones, 3.0% used tablet devices, and 0.5% used their television (TV). The high percentage of users who use their computers is attributed to MATLAB and Arduino software installations on personal computers. The videos were designed such that the viewer can follow along with the videos on their computers. Other devices such as mobile phones and tablets feature lower usage.

3) Gender demographics: The gender-based demographic distribution on YouTube shows that the male viewership was 91.4% and female 8.6%. This data illustrates the gender discrepancy in engineering education since most of the viewers were males as opposed to females. YouTube provides insufficient data on other gender identities.

4) Top user traffic sources: Table II depicts the top 6 external sources of viewer traffic to the YouTube channel and the associated percentage of watch hours. Google Search rakes in the highest percentage of viewers with 66.8%. Viewers interested in learning about interfacing Arduino and MATLAB look up resources online for step-by-step tutorials. Being the most popular search engine, Google Search is the number one source of bringing viewers to the channel. The second external source to the channel is the `MATLABArduino.org` website. Since the videos are embedded into the website,

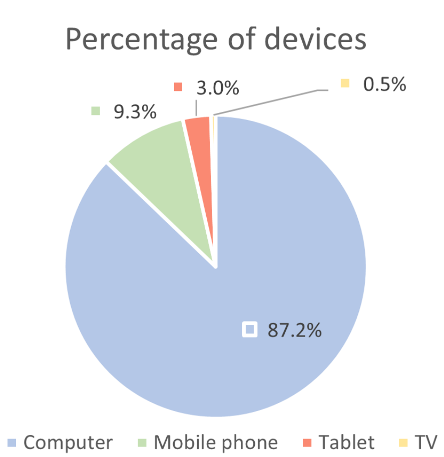

Fig. 12: Types of devices used to access the video lectures

TABLE II: Top viewer traffic sources

Source	Percentage of views	Percentage watch time (hours)
Google Search	66.8%	66.6%
matlabarduino.org	13.6%	14.1%
Google	5.2%	5.9%
YouTube	1.5%	1.2%
Facebook	1.2%	1.2%
Instructables.com	0.4%	0.4%

viewers visiting the website will often watch the videos on the website. The YouTube website itself was responsible for only 1.5% of the viewers. The organic nature of viewership indicates the relevance of the content provided on the website and the YouTube channel.

5) Viewership hours: Viewership hours signify the duration the viewer watched the videos. It is a crucial metric in determining the engagement of the viewers with the video content. A key challenge online video lectures face is the average view duration for a video, which is often less than the total length of the video. Studies have shown that shorter videos are much more engaging than longer videos [24]. While utmost care is taken to create videos shorter than 10 minutes (the average video length on this YouTube channel is approximately 6 minutes), it is observed that the average view length is approximately 3 minutes. Based on this observation, later videos were shorter than the earlier videos on the channel. Therefore, the shorter videos have witnessed a higher than average view duration.

B. Qualitative Analysis

Viewers are free to leave comments under the videos on YouTube. To date, the YouTube channel has witnessed approximately $5.9K$ subscribers. A qualitative analysis of the YouTube comments left by the viewers is provided, along with highlighted comments showing recurring themes. These themes include compliments on the content, questions regarding the particular video content, constructive feedback, and source code requests.

1) Appreciative comments: Here are examples of positive feedback about the content of the video. *"The tutorial is awesome, thank you so much for sharing"*, *"This was brilliant. Thanks a lot. It will help me visualize the motion of a rotating body to illustrate my research results very well. "* The positive comments indicate that viewers were satisfied with the video content and earned valuable information from them.
2) Questions related to the content: Here is an example of a viewer satisfied with the content and looking to advance to other concepts related to the ones explained in the video related to plotting accelerometer data. *"Excellent work friend, I am also developing something very similar but I want to plot 3D the g-forces from the accelerometer, where a chart changes color depending on the force, with your experience in this work you've done and it's great can you help me? "*
3) Constructive feedback: *"Appreciated your video! But you should explain step by step slowly and going over*

the code properly. The video feels like a rushed presentation more than a tutorial. Thanks anyway." In this comment, the viewer conveyed that they found the video was rushed and did not take its time in explaining the concepts. The valuable feedback was taken into account for several future videos. *"This video is useful, but it would be perfect if it explained the stop_call_temp graph function, I don't even know where to start from to develop this and its one of the most generic things that could have been explained in this tutorial!"* The viewer shared their inability to proceed because some concepts were not explained in the video. Based on this, additional content was added to the video to address such situations.

4) Source code requests: Significant number of messages request for the code shown in the videos. Sufficient care was taken to go over each line of code in the video itself. As a matter of policy and in the spirit of active learning, these videos are meant to encourage the audience to create and further curate their code repository. In the same vein, the viewers are encouraged to write their code. The YouTube comment section is continuously monitored and any questions asked in the comments are quickly addressed by the video creators.
5) Some viewers expressed difficulties in following the videos. The viewers left comments saying that they felt the videos were rushed due to the speaking style of the video voiceover and an American accent. This feedback was taken into account, and the video voiceover was slowed down. The YouTube feature of providing automatic closed captioning added an extra layer of accessibility for the viewers.

C. Key Challenges

Operating an educational initiative such as `MATLABArduino.org` comes with its challenges. Some of these challenges are enumerated below:

1) MATLAB is a ubiquitous scientific software. At the same time, it is a paid software, which inherently limits the number of students who can use the educational content.
2) Continuous MATLAB software updates mean that the educational content development team has to update videos continuously, requiring significant financial and time commitment. This commitment is challenging for an educational laboratory located at a university.
3) The videos use specific sensor modules. While every effort is put in selecting the most popular sensor modules that are also low-cost, accessibility of these modules worldwide cannot be guaranteed. Additionally, sensor chips are phased out over time and replaced by newer chips. Keeping up with such hardware updates also requires financial and time commitment.
4) A significant viewership of this channel comes from outside of North America. Therefore, providing localized content is essential to engage students who do not speak English as their first language. Localization requires additional script writing, voiceover, and video editing.
5) Historically, YouTube has not reached the Chinese student audience. This observation indicates reaching the Chinese audience using other platforms, which is an additional overhead that the educational content team has to manage.

VI. Conclusions

MatlabArduino.org and its companion YouTube channel pioneered the use of MATLAB for embedded systems education starting in 2013. This paper provides an overview of the content provided on the website and the YouTube channel. We also report on the YouTube viewership trends in terms of country-wise percentage viewership, computing devices used to watch the videos, gender demographics, top user traffic sources, and viewership hours. Two examples of in-class use of this website in undergraduate engineering classes were provided. The growing number of YouTube subscribers and viewer engagement indicates the relevance of this content. As online education becomes increasingly pervasive, the insights presented here may serve as a guide for the design of future open-education initiatives.

VII. Acknowledgements

This development of this manuscript was supported in part by the National Science Foundation under Grant DRL-2010259. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] M. Kassab, J. DeFranco, and P. Laplante, "A systematic literature review on Internet of things in education: Benefits and challenges," *Journal of Computer Assisted Learning*, vol. 36, no. 2, pp. 115–127, 2020.
[2] R. S. Abd-Ali, S. A. Radhi, and Z. I. Rasool, "A survey: the role of the Internet of Things in the development of education," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 1, pp. 215–221, 2020.
[3] H. Aldowah, S. U. Rehman, S. Ghazal, and I. N. Umar, "Internet of things in higher education: A study on future learning," in *Journal of Physics: Conference Series*, vol. 892, no. 1. IOP Publishing, 2017, p. 012017.
[4] P. Abichandani, "MATLAB Arduino." [Online]. Available: "http://www.matlabarduino.org/serial-communication.html"
[5] Mathworks, "MATLAB for students." [Online]. Available: https://www.mathworks.com/products/matlab.html
[6] A. H. Assi, *Engineering education and research using MATLAB*, 2011.
[7] Arduino, "Arduino." [Online]. Available: https://www.arduino.cc/
[8] Y. A. Badamasi, "The working principle of an arduino," in *2014 11th international conference on electronics, computer and computation (ICECCO)*. IEEE, 2014, pp. 1–4.
[9] D. Nichols, "Arduino-based data acquisition into excel, labview, and matlab," *The Physics Teacher*, vol. 55, no. 4, pp. 226–227, 2017.
[10] S. Roodt and D. Peier, "Using youtube© in the classroom for the net generation of students," in *Proceedings of the Informing Science and Information Technology Education Conference*. Informing Science Institute, 2013, pp. 473–488.
[11] N. Montes, N. Rosillo, M. C. Mora, and L. Hilario, "A novel real-time matlab/simulink/lego ev3 platform for academic use in robotics and computer science," *Sensors*, vol. 21, no. 3, p. 1006, 2021.

[12] J. Sheng, "Teaching devices and controls for computer engineering and systems students using arduino and matlab/simulink," in *2018 IEEE 14th International Conference on Control and Automation (ICCA)*. IEEE, 2018, pp. 318–323.
[13] R. Gonzalez, C. Mahulea, and M. Kloetzer, "A matlab-based interactive simulator for mobile robotics," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*. IEEE, 2015, pp. 310–315.
[14] R. Beneder, M. Lechner, and P. Schmitt, "Increasing the accessibility of measurement equipment for undergraduate courses dedicated to embedded systems," in *2017 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 2017, pp. 161–164.
[15] J. C. Martinez-Santos, O. Acevedo-Patino, and S. H. Contreras-Ortiz, "Influence of arduino on the development of advanced microcontrollers courses," *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, vol. 12, no. 4, pp. 208–217, 2017.
[16] T. Acosta, P. Acosta-Vargas, J. Zambrano-Miranda, and S. Lujan-Mora, "Web accessibility evaluation of videos published on youtube by worldwide top-ranking universities," *IEEE Access*, vol. 8, pp. 110 994–111 011, 2020.
[17] A. Shoufan, "Active distance learning of embedded systems," *IEEE Access*, vol. 9, pp. 41 104–41 122, 2021.
[18] O. M. Oteri, "The arduino e-kit as applied in engineering, science and technology e-learning," in *2020 Sixth International Conference on e-Learning (econf)*. IEEE, 2020, pp. 285–289.
[19] Arduino, "UNO R3." [Online]. Available: https://store.arduino.cc/usa/arduino-uno-rev3
[20] Mathworks, "Arduino support from MATLAB." [Online]. Available: https://www.mathworks.com/hardware-support/arduino-matlab.html
[21] "Zigbee/ieee 802.15.4 alliance," p. May 17, 2021. [Online]. Available: https://zigbeealliance.org/solution/zigbee/
[22] *Raspberry Pi*, 2021. [Online]. Available: https://www.raspberrypi.org/
[23] A. Augustin and contributors, "Websockets," p. May 17, 2021. [Online]. Available: https://websockets.readthedocs.io/en/stable/
[24] P. J. Guo, J. Kim, and R. Rubin, "How video production affects student engagement: An empirical study of mooc videos," in *Proceedings of the first ACM conference on Learning@ scale conference*, 2014, pp. 41–50.