VitalCore: Analytics and Support Dashboard for Medical Device Integration

Hyonyoung Choi*, Amanda Lor+, Mike Megonegal+, Xiayan Ji*, Amanda Watson*, James Weimer*, Insup Lee*
*Dept. of Computer and Information Science, University of Pennsylvania
{hyonchoi, xjiae, aawatson, weimerj, lee}@seas.upenn.edu
+Penn Medicine, University of Pennsylvania
{Amanda.Lor, Michael.Megonegal}@pennmedicine.upenn.edu

***Abstract*—Medical professionals spend extensive time collecting, validating, reviewing, and analyzing medical device data. These devices use vendor-specific applications with lengthy troubleshooting times, causing extended downtimes where medical professionals have to manually document patient data in the electronic health record (EHR). Manual logging of this data creates delays and leaves it vulnerable to errors, manipulation, and omissions. In this paper, we present VitalCore, a medical device integration platform that supports access to medical device data in real-time. We deploy VitalCore in three applications at Penn Medicine: Medical Device Dashboard, Ventilation Alert, and Anomaly Detector. In the Medical Device Dashboard, we reduced, by up to six times, the amount of time required of medical professionals, clinical engineers, and IT analysts by simplifying the troubleshooting workflow, thus decreasing downtimes and increasing clinical productivity. In Ventilation Alert, we demonstrated the ability to assist medical professionals by alerting them to newly ventilated patients. In Anomaly Detector, we showed that we could predict anomalous patterns in our data with 93% accuracy.**

I. Introduction

The Internet of Medical Things (IoMT) is a complex system of networked medical devices that share medical device data with healthcare professionals to enable new and innovative medical services. These devices range from wearables (e.g., Fitbit) to implantables (e.g., pacemakers) and medical equipment (e.g., magnetic resonance imaging (MRI) machines and ventilators). It is even expected that upwards of 68% of all medical devices manufactured will be connected by 2022 [1]. Further, IoMT is expected to continue growing as forecasts predict it will reach a market value of over $135 billion by 2025 [2]. As the IoMT market continues to grow, the systems that support these devices will need to adapt, bringing new software, hardware, and cybersecurity solutions.

As medical devices come online, systems are developed to support the storage, transmission, and security of medical device data. This has led to non-standardized vendor-specific applications that require specialized training. Thus, medical professionals spend excessive time interfacing with medical devices to collect, validate, review, and analyze medical device data. When these devices malfunction, not only are there extended downtimes for the device, medical professionals have to manually document device data in the electronic health record (EHR), distracting them from direct patient care. Consequently, manual logging of this data leaves it vulnerable to errors, manipulation, and omissions.

The integration of medical equipment in the IoMT has led to massive improvement in the quality of patient care [3]–[7]. It has also led to the coordination of Medical Cyber-Physical System (MCPS) and IoMT to provide better information to the caregiver, detect failures of individual devices, and improve patient safety and treatment effectiveness. Thus, researchers have started developing integration platforms that allow for a large number of medical devices. These platforms focus on bringing old hardware online [8] and interoperability between devices [9]–[11]. VitalCore outperforms these software platforms by ensuring clinical devices are operational while providing a user-friendly dashboard for caregivers without technical backgrounds. Its dashboard is designed with troubleshooting in mind to minimize downtimes creating a more efficient workflow in medical environments.

Currently, Penn Medicine has over 3,000 integrated medical devices over thirteen facilities from seven different vendor networks. Consequently, this extensive network of medical devices has led to the many challenges discussed previously. To address these challenges, we developed VitalCore, a platform to manage clinical devices and proactively keep them operable while improving the workflow for the IT analysts and clinical engineers. VitalCore not only has clinical benefits but technical and research benefits as well. Clinically, manual documentation is reduced, providing time savings and real-time, accurate data is fed to clinical decision support systems. To demonstrate this, we build three applications Medical Device Dashboard, Ventilation Alert, and Anomaly Detector. Technically, troubleshooting efficiency is increased to minimize downtime, and responses are moved from reactive to proactive. This efficiency has led to a decrease of three to six times the time needed for troubleshooting. Additionally, data is archived to support future research and analysis.

Specifically, our contributions are:

1) Development of VitalCore, a vendor-neutral platform that, to our knowledge, is the first of its kind in the industry that manages clinical devices and proactively keeps them operable while improving the workflow for IT analysts and clinical engineers.
2) The VitalCore system architecture supports the development and deployment of various applications, includ-

978-1-6654-3965-7/21/$31.00 ©2021 IEEE
DOI 10.1109/CHASE52844.2021.00016

ing user-friendly dashboards, clinical alert systems, and anomaly detection.

The remainder of this paper is structured as follows: First, we summarized the work related to this paper in Section II. In Section III, we describe the VitalCore system. Then, Section IV describes applications in which VitalCore is being used. In Section V, we evaluate our system. Finally, we conclude our paper in Section VI.

II. Related Work

The tendency of migrating medical devices online has become more and more prominent in the world of the Internet of Medical Things. Devices such as ventilators [3], pulmonary monitors [4], medical equipment in ambulances [6], and surgical devices in operating rooms [5] are being brought online. Additionally, integrating devices with an online platform has made remote health monitoring more convenient. Devices such as ECGs [12], [13], insulin pumps [14], and heart rate monitoring via Apple Watches [7] track a patient's health in the comfort of their own home and send this data back to clinicians for further analysis. As IoMT brings these monitoring devices online, the resulting Medical Cyber-Physical System (MCPS) has the ability to provide more intelligent information to clinicians and caregivers, detect failures of devices, and improve patient safety and treatment effectiveness.

Researchers and engineers have begun developing integrated systems that manage a large number of heterogeneous medical devices spanning the domains of hardware and software. For instance, Prudenzi et al. [8] implemented a hardware system that installed a Raspberry Pi 3 near medical devices of interest and connected them to an online supervisory system. Asare et al. provides a dongle to connect previously unconnected medical devices [15]. In addition, software frameworks tackle the interoperability challenges between devices. OpenICE [9] is an open-source Integrated Clinical Environment (ICE) that assists in research for connecting IoMT devices. Expanding on that, OpenICE-lite [10] provides security guarantees and real-time data visualization and analysis. HIP [11] is an end-to-end software integration platform that generalized the wireless body sensor framework to test for correctness and performance in health applications. VitalCore outperforms other software platforms because it not only maintains clinical devices and ensures that they are operational but also provides a user-friendly GUI dashboard for caregivers without technical backgrounds. This is a critical functionality to maintain efficient workflows in medical environments where users may not be technology experts.

III. VitalCore System

The overall architecture of VitalCore is depicted in Figure 1. VitalCore takes as input the HL7 data feed streaming from medical devices. This data is fed into the stream processor for processing and routing. Then, data is stored to be displayed in the dashboard and for future analysis. Next, the communication protocols provide the processed data to the applications. Applications can send additional data back to VitalCore, in which the communication protocols can route this data to our storage.

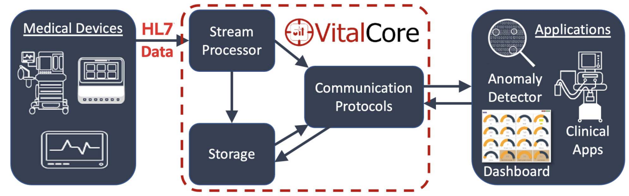


Fig. 1: VitalCore System Architecture

a) HL7 Data: Health Level Seven, commonly known as HL7, is a set of widely supported international standards that promote the transfer of medical data between software applications used by various healthcare providers. By providing commonly supported data transfer guidelines, medical data can be exchanged across EHRs and other software applications without ambiguity and risk of misinterpretation. In our test environment, we received the HL7 Data stream not directly from the medical devices but from the integrated middleware system. However, VitalCore supports any medical devices that are capable of networking or support a network integration adapter (e.g., Capsule Neuron [16]).

b) Stream Processor: The Stream Processor accomplishes two tasks: it generates and updates meta-data and routes the data to storage and the communication protocols. To generate metadata, we extract the device identifier and the arrival time of the message to VitalCore. With this information, we generate two meta-data tables: the timestamp history for each device and the latest timestamp for each device. Then, the data and meta-data are sent to storage and the applications.

c) Storage: VitalCore uses TimescaleDB for HL7 Data and MongoDB for any other data. HL7 data is time-series as each medical device is sending its HL7 data repeatedly over time. Hence, there are performance and scalability benefits to storing HL7 data in a time-series database. MongoDB stores our non-time-series data such as user accounts, meta-data, application-specific data, etc. While these databases are separate, we use common identifiers to provide connections between the data.

d) Communication Protocols: Communication Protocols include REST-API (REpresentational State Transfer API) and Web-Socket. REST-API uses queries to access data from storage. But, some of the applications need a real-time stream of HL7 data. In this case, HL7 Stream Processor forwards the HL7 data directly to the applications via a Web-Socket. The streaming data is directly sent from the HL7 Stream Processor bypassing the database. This data can also be sent back from the applications to the communication protocols for further routing to storage or other applications.

e) Applications: In VitalCore, we prioritized flexibility and modularity to promote the support and creation of many applications. Thus, applications with varied functionalities can be built on top of the VitalCore system. This allows for custom, tailored applications that meet the specific needs of clinicians and IT staff to be built. In this paper, we will discuss

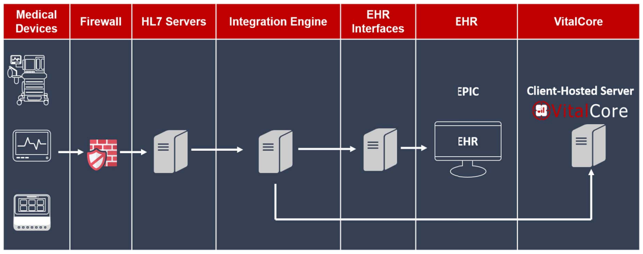


Fig. 2: VitalCore Integration

the following scenarios: medical device dashboard, ventilation alert, and anomaly detection. Detailed functionalities of the applications will be explained in Section IV.

IV. Applications

VitalCore is used in three applications: Medical Device Dashboard, Ventilation Alert, and Anomaly Detector. The Medical Device Dashboard provides clinicians access to real-time, accurate data in the EHR with a user-friendly GUI. Further, it streamlines the troubleshooting process for IT staff. Ventilation alert showcases the integration of medical devices, in this case, ventilators, to send data in real-time to the EHR and generates alerts sent to medical professionals. The anomaly detector detects anomalies in the usage patterns of medical devices and groupings of medical devices.

A. Medical Device Dashboard

The medical device dashboard shown in Figure 3 is a graphical user interface (GUI) designed to allow users (e.g., IT analysts and clinical engineers) to find and identify essential information (e.g., device name, location, vendor, etc.) within the timespan of a minute. To tailor the dashboard to those using it, we analyzed the usage patterns of our users to identify important functionalities that support the navigation of existing tools. After discovering the most valuable features, we created a mock-up user interface (UI) which was employed to collect user feedback. From this feedback, we redesigned our UI and tested it with real-time data.

First, we observed the troubleshooting workflow of the IT analysts with the goal of saving their time by improving the workflow with an integrated approach to medical device management. We found that when a device needed troubleshooting, the IT analysts, in general, performed the seven steps shown in Figure 4. Among these steps, we identified where improvements could be made. Three steps were deemed unnecessary and time-consuming. We determined that they could be accounted for in a single login to the VitalCore system: identifying device info, retrieving server information, and launching HL7 tests. This reduced our troubleshooting workflow to four steps, as shown in Figure 4. VitalCore reduced the need for analysts to use an excel spreadsheet, look for login credentials and server names, run lengthy searches, or contact other teams for support. Further, analysts were limited in the past by relying on vendor solutions and tools that were specific to each vendor's medical technology. For example, one vendor's app displayed the HL7 data output status for their technology, while other vendors did not. VitalCore reduced the need to learn to use multiple solutions and provides a single, standardized platform for our users.

B. Ventilation Alert

Respiratory Therapists (RT) in the ICU manage patients on ventilators based on clinician-developed treatment plans. A large portion of their job revolves around monitoring these patients to evaluate their treatment. When a patient is not reacting as expected, the RT troubleshoots the issue and consults with clinicians to make changes to the ventilator settings. To free up the RT to focus on the more critical, troubleshooting portion of their job, a telemedicine respiratory therapist (eRT) is stationed at the virtual intensive care unit (VICU) to remotely monitor newly intubated patients. When a patient is not reacting well to treatment, the eRT contacts the RT for troubleshooting.

Currently, ventilators do not send a start status message to the EHR, and thus the eRT is not notified through the VICU. The eRT relies on calls from the onsite nurse or respiratory therapist (RT) or validated data in the EHR, which is not real-time. As a result, patient monitoring via the VICU is delayed. To solve this problem, ventilation start time can be extracted from the real-time HL7 messages sent every minute from the ventilators. Within these messages, a variable ID, expired tidal volume (TV), can be used as an indicator that a patient has been intubated. Expired TV is the volume of air that a patient breathes out. If an expired TV has a value greater than zero, the receive time for the message is noted as the ventilation start time. To alert the eRT, an intubation alert is sent as a text message to the eRT's phone.

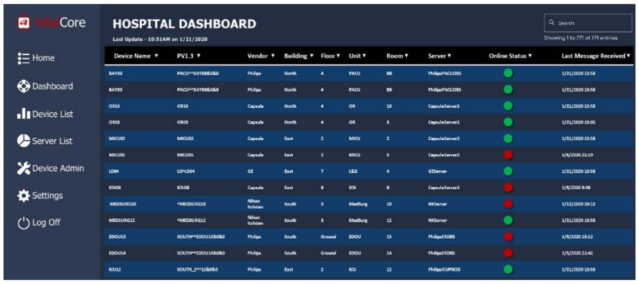
When only using the expired Tidal Volume to determine start time, we noticed many false positives, largely with patients who were already ventilated. To minimize these false positives, the following logic was implemented:

1) If the patient-name is different from the other devices, update the patient-name and always send an alert
2) If the patient-name is the same as the other devices and the previous stop time is less than an hour, skip the alert.
3) Otherwise, send an alert

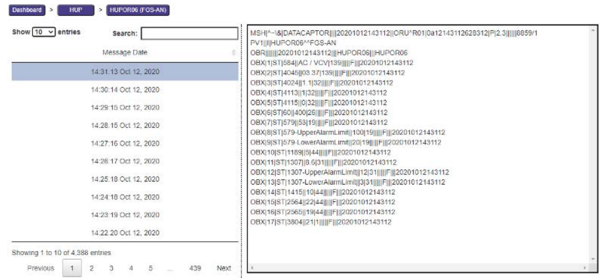
The alert system is currently under evaluation for its effectiveness and safety.

C. Anomaly Detector

An important factor in changing reactive troubleshooting to proactive is monitoring anomalies in the usage pattern of medical devices. For example, a medical device sends a message every minute during business hours but sends a reduced number of messages after business hours and during weekends. But when unexpected events occur, such as an emergency for a patient or a network outage, we should detect and respond appropriately to the event. VitalCore detects these anomalous situations in real-time to bring nursing or IT personnel to investigate them. Moreover, some anomalies appear in several devices within the same group (e.g., room,



(a) Dashboard Home



(b) HL7 Messages

Fig. 3: Medical Device Dashboard

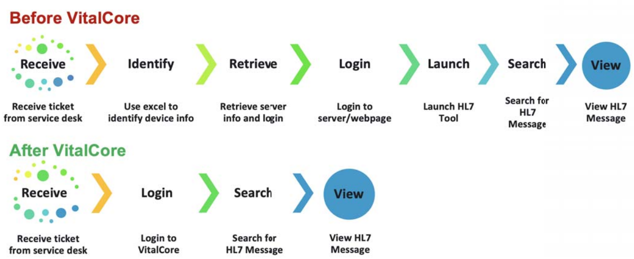


Fig. 4: Troubleshooting Workflow

floor, nursing units). VitalCore detects these patterns as well, allowing for a more comprehensive view of the anomalies.

We train machine learning models to learn the normal usage patterns of devices. We monitor the usage patterns of 60-second intervals, weekday, and weekend, where each pattern has its own model. The trained models retain a compressed representation of the patterns and use it to reconstruct the input. After that, we compute the Mean Absolute Error (MAE) between the reconstructed input and the raw input. Then, during training, we fed all training instances to the model and choose the maximum reconstruction error as the threshold T to determe anomalies. During testing, each new instance i received in real-time is fed to the trained models where the maximum reconstruction error err_i among all pattern models is calculated. If $err_i > T$, we declare anomaly. Otherwise, we consider it normal.

V. Evaluation

We evaluate our system in each of the applications described in the previous section. First, we consider the impact the Medical Device Dashboard has on the troubleshooting workflow. Second, we assess the ventilator alert system. Finally, we analyze the anomaly detector.

A. Medical Device Dashboard

A goal of VitalCore is to increase troubleshooting efficiency to minimize downtimes. To do this, we simplified the troubleshooting workflow as shown in Figure 4. We evaluate the improved workflow by comparing the time it takes to troubleshoot using VitalCore to using vendor-specific software. Figure 5 shows a direct comparison between multiple vendors and VitalCore. Overall, we see a decrease of three to six times the amount of time(4.5 minutes to 50 seconds) needed to troubleshoot for these vendors exemplifying the benefits of using VitalCore during the troubleshooting process.

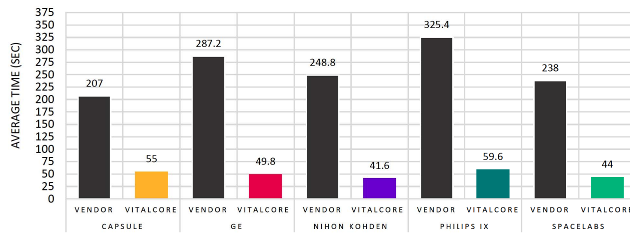


Fig. 5: Troubleshooting Time Comparison

B. Ventilation Alert

With device integration, respiratory therapists spend five minutes per ventilator check to validate ventilator data in the EHR. When there is a disruption in the HL7 data flow from the ventilator to the EHR, respiratory therapists must manually document readings and settings, which takes three times longer (15 minutes) per ventilator check. The longer it takes for analysts to troubleshoot and restore data flow, the more time respiratory therapists must spend on manual documentation instead of directly caring for their patients.

We conducted a pilot study using 139 ventilators where a telemedicine respiratory therapist (eRT) is stationed at the virtual intensive care unit (VICU) to remotely monitor newly intubated patients. Over the course of three months, 3196 alerts were sent in total. Of these alerts, we were able to successfully filter out 872 false alert messages by the filtering logic given in the previous section. While we filtered out most of the false alerts, it was reported by the eRT's that not all of them were filtered out. In general, these were patients that were already intubated. While it is not difficult for the eRT to ignore this message, we plan to improve our filtering logic further to distinguish between new and continued intubation.

C. Anomaly Detector

The anomaly detector module of VitalCore identifies the abnormal usage pattern of medical devices while avoiding raising excessive false alarms that cause alarm fatigue. We train machine learning models to learn the normal usage patterns of devices. We monitor the usage patterns of 60-second intervals, weekday, and weekend, where each pattern has its own model. Our training data is collected over five monitoring devices, which are composed of 48,096 samples of constant 60-second-intervals pattern and 6,791 samples of reduced activity after business hours on weekdays and weekends. Note, no abnormal sample is fed to the models at training. In other words, we want the models to learn the normal pattern and test their performance on abnormal instances. During testing, we provide the trained models with 48,969 samples, of which 8% are anomalous. Each model classifies each instance as anomalous or not. If any of the trained models declare a test instance to be an anomaly, we consider the classification result to be anomalous.

TABLE I: Performance Comparison of Anomaly Detection Algorithms. ACC: accuracy, F1: f1-score, PRE: precision, REC: recall, FPR: false positive rate, FNR: false negative rate, T: training time.

	ACC	F1	PRE	REC	FPR	FNR	T
Autoencoder	**0.93**	**0.87**	**0.84**	**1.00**	**0.08**	**0.00**	**787.41**
1-class SVM	0.81	0.57	0.42	1.00	0.21	0.00	280.04
Matrix Profile	0.99	-	0.00	0.00	0.00	1.00	295.88
tsmoothie	0.99	-	0.00	0.00	0.00	1.00	-

To choose the most suitable anomaly detection algorithm for each application of VitalCore, we tested four state-of-the-art algorithms, namely convolutional autoencoder [17], one-class SVM [18], matrixprofile [19] and tsmoothie [20], and evaluate their performance as shown in Table I. We chose these algorithms because they are benchmark algorithms used for unsupervised time-series anomaly detection. One-class SVM sacrifices accuracy for efficiency in run time. Matrix profile gives higher accuracy with a longer run time. Tsmoothie runs offline hence the time is not listed here for comparison. Matrix profile and tsmoothie give zero for precision, recall, and false positive rate, because they do not fit our dataset well and fail to predict any anomaly. Overall, the convolutional autoencoder provides the best performance for our application as it exhibits the minimum false alarm rate while performing highly in overall accuracy, recall, and false-negative rate.

VI. Conclusion

In this paper, we presented VitalCore, a medical device integration platform that supports clinical decisions by reducing the manual documentation of medical device data and providing access to real-time, accurate data in the EHR. We deployed VitalCore in three real world applications at Penn Medicine: Medical Dashboard, Ventilation Alert, and Anomaly Detector. After evaluation, we found that VitalCore reduced the amount of time required of medical professionals, clinical engineers, and IT analysts by up to six times when troubleshooting. Further, we could accurately and in real time extract ventilation information and alert appropriate personnel. Finally, we detected anomalies in device usage to change troubleshooting responses from reactive to proactive.

Acknowledgment

This research was supported in part by the National Institute of Health under grant R01EB029767.

References

[1] "Medtech and the internet of medical things," https://www2.deloitte.com/uk/en/pages/life-sciences-and-healthcare/articles/medtech-and-the-internet-of-medical-things.html, accessed: June, 2021.

[2] "Projected size of the internet of things (iot) in healthcare market worldwide from 2016 to 2025," https://www.statista.com/statistics/997959/worldwide-internet-of-things-in-healthcare-market-size/, pubished: December, 2016.

[3] R. L. Read, L. Clarke, and G. Mulligan, "Ventmon: An open source inline ventilator tester and monitor," *HardwareX*, vol. 9, p. e00195, 2021.

[4] H. Nguyen, R. Ivanov, S. B. DeMauro, and J. Weimer, "Repulmo: A remote pulmonary monitoring system," *SIGBED Rev.*, vol. 16, no. 2, p. 46–50, Aug. 2019. [Online]. Available: https://doi.org/10.1145/3357495.3357501

[5] M. Kasparick, M. Schmitz, B. Andersen, M. Rockstroh, S. Franke, S. Schlichting, F. Golatowski, and D. Timmermann, "Or. net: a service-oriented architecture for safe and dynamic medical device interoperability," *Biomedical Engineering/Biomedizinische Technik*, vol. 63, no. 1, pp. 11–30, 2018.

[6] B. Almadani, M. Bin-Yahya, and E. M. Shakshuki, "E-ambulance: real-time integration platform for heterogeneous medical telemetry system," *Procedia Computer Science*, vol. 63, pp. 400–407, 2015.

[7] M. V. Perez, K. W. Mahaffey, H. Hedlin, J. S. Rumsfeld, A. Garcia, T. Ferris, V. Balasubramanian, A. M. Russo, A. Rajmane, L. Cheung *et al.*, "Large-scale assessment of a smartwatch to identify atrial fibrillation," *New England Journal of Medicine*, vol. 381, no. 20, pp. 1909–1917, 2019.

[8] A. Prudenzi, A. Fioravanti, and M. Regoli, "A low-cost internet of things integration platform for a centralized supervising system of building technology systems in hospitals," in *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*, 2018, pp. 1–6.

[9] D. Arney, J. Plourde, and J. M. Goldman, "Openice medical device interoperability platform overview and requirement analysis," *Biomedical Engineering/Biomedizinische Technik*, vol. 63, no. 1, pp. 39–47, 2018.

[10] R. Ivanov, H. Nguyen, J. Weimer, O. Sokolsky, and I. Lee, "Openice-lite: Towards a connectivity platform for the internet of medical things," in *2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC)*. IEEE, 2018, pp. 103–106.

[11] J. Woodbridge, H. Noshadi, A. Nahapetian, and M. Sarrafzadeh, "Hip: Health integration platform," in *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2010, pp. 340–345.

[12] Z. Yang, Q. Zhou, L. Lei, K. Zheng, and W. Xiang, "An iot-cloud based wearable ecg monitoring system for smart healthcare," *Journal of medical systems*, vol. 40, no. 12, pp. 1–11, 2016.

[13] H. Xia, I. Asif, and X. Zhao, "Cloud-ecg for real time ecg monitoring and analysis," *Computer methods and programs in biomedicine*, vol. 110, no. 3, pp. 253–259, 2013.

[14] Z. A. Al-Odat, S. K. Srinivasan, E. Al-qtiemat, M. A. L. Dubasi, and S. Shuja, "Iot-based secure embedded scheme for insulin pump data acquisition and monitoring," *arXiv preprint arXiv:1812.02357*, 2018.

[15] P. Asare, D. Cong, S. G. Vattam, B. Kim, A. King, O. Sokolsky, I. Lee, S. Lin, and M. Mullen-Fortino, "The medical device dongle: An open-source standards-based platform for interoperable medical device connectivity," in *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, 2012, pp. 667–672.

[16] "Neuron, medical device integration, capsule technologies," https://capsuletech.com/neuron, accessed: June, 2021.

[17] "Keras documentation: Timeseries anomaly detection using an autoencoder," https://keras.io/examples/timeseries/timeseries_anomaly_detection/.

[18] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, J. C. Platt *et al.*, "Support vector method for novelty detection." in *NIPS*, vol. 12. Citeseer, 1999, pp. 582–588.

[19] C.-C. M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, H. A. Dau, Z. Zimmerman, D. F. Silva, A. Mueen, and E. Keogh, "Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile," *Data Mining and Knowledge Discovery*, vol. 32, no. 1, pp. 83–123, 2018.

[20] "tsmoothie: A python library for time-series smoothing and outlier detection in a vectorized way." https://github.com/cerlymarco/tsmoothie.