Internet of Things 18 (2022) 100454

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot

Low cost cloud based remote microscopy for biological sciences

Pierre V. Baudin [a,*,1], Victoria T. Ly [a,1], Pattawong Pansodtee [a], Erik A. Jung [a], Robert Currie [b], Ryan Hoffman [c], Helen Rankin Willsey [e], Alex A. Pollen [f,h], Tomasz J. Nowakowski [i,h], David Haussler [b,d,g], Mohammed A. Mostajo-Radji [b,f,h], Sofie R. Salama [b,d,g], Mircea Teodorescu [a,b]

[a] Department of Electrical and Computer Engineering, University of California Santa Cruz, United States of America
[b] Genomics Institute, University of California, Santa Cruz, United States of America
[c] Department of Molecular, Cellular, and Developmental Biology, University of California, Santa Cruz, United States of America
[d] Department of Biomolecular Engineering, University of California, Santa Cruz, United States of America
[e] Department of Psychiatry and Behavioral Sciences, Weill Institute for Neurosciences, University of California, San Francisco, United States of America
[f] Department of Neurology, University of California, San Francisco, United States of America
[g] HHMI: Howard Hughes Medical Institute, University of California, Santa Cruz, United States of America
[h] The Eli and Edythe Broad Center of Regeneration Medicine and Stem Cell Research, University of California, San Francisco, United States of America
[i] Department of Anatomy, University of California, San Francisco, United States of America

ARTICLE INFO

Keywords:
Raspberry Pi
Remote microscopy

ABSTRACT

A low cost remote imaging platform for biological applications was developed. The "Picroscope" is a device that allows the user to perform longitudinal imaging studies on multi-well cell culture plates. Here we present the network architecture and software used to facilitate communication between modules within the device as well as system communication with external cloud services. A web based console was created to control the device and view experiment results. Post processing tools were developed to analyze captured data in the cloud. The result is a platform for monitoring biological experiments from outside the lab.

1. Introduction

The COVID-19 pandemic has changed the work landscape throughout the world. Wherever possible jobs have transitioned to a remote format in compliance with lockdown regulations. Bench scientists were unequally affected by this transition, experiencing a substantial reduction in their ability to produce work compared to computational scientists [1]. This situation is likely to have long-lasting consequences on science careers, particularly for junior investigators. This has motivated the development of new approaches for remote experimentation. These advancements will have lasting benefits long after the pandemic is over. We describe one such development here.

In addition to allowing a greater quantity of work to be done, remote experimentation can increase the quality of work coming out of a lab. There is a "crisis of reproducibility" that exists within biology [2,3]. Scientists and technicians following prescribed protocols are often unable to replicate each other's results. For example, in cell culture experiments, differences in how often temperature and gas regulated incubators are opened or how much time a sample spends out of the incubator for routine manipulation can

* Corresponding author.
E-mail address: pvbaudin@ucsc.edu (P.V. Baudin).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.iot.2021.100454
Received 1 June 2021; Received in revised form 9 August 2021; Accepted 17 August 2021
Available online 11 September 2021

cause varying amounts of stress on the culture. This can affect the metabolism of the cells and cause unpredictable effects on the experimental results. This leads to unaccounted for experimental variability. Increasing automation in lab experiments has been proposed as a way to address this issue. [4]. Remotely operated experimentation entails this type of increased automation.

Techniques for remote operation of lab equipment exist on a wide spectrum of cost and complexity, from fully automated labs utilizing expensive robotic systems [5,6], to Do-it-Yourself (DIY) 3D printed microscopes with basic Internet access [7,8]. The further development of low cost solutions for remote lab control will bring more options within the reach of institutions with limited means [9]. This would allow under resourced labs to reap many of the benefits of the "lab of the future"[4]. Many cost reductions have been made possible by innovations in the Internet of Things (IoT) space [10], ranging from software frameworks to low cost network capable devices [11–14].

The landscape of low cost lab solutions features many contributions from the "maker" community. These enthusiasts are creating designs to manufacture lab equipment with consumer accessible tools [9,15,16]. Several open source microscope designs have been proposed using 3D printing and low cost computing platforms like the Raspberry Pi [7,8,17]. The Raspberry Pi is small low cost computer that runs a fully featured linux based operating system. Starting as low as 5$, the Raspberry Pi substantially reduces the cost of computer clusters. Clusters used to be considered a major investment only used for intense computation tasks. With raspberry pi, it is affordable to build a dedicated cluster for almost any application. [18–20].

In the world of education, remote lab systems have a history of use as a classroom tool alongside fully simulated lab systems [21–23]. Simulated labs have been used as a replacement for, or supplement to, traditional educational lab experience [24,25] with the aim of introducing students without access to the necessary experimental equipment and environment to the experience of the scientific process. However, simulations can never provide students with the experience of actually discovering something new. By contrast, Remote lab experimentation allows students to manipulate live experiments running on real lab equipment from their classroom and home computers, or from their mobile phones. This removes the stale predictability of fully simulated experimentation, giving students a chance to experience the actual scientific process of discovery. Remote microscopy is an important aspect of many of these remote lab experiments [26–28].

We recently described a device for simultaneous longitudinal imaging which we call the "Picroscope" [29]. Here we describe the software and network architecture developed to run this device as well as its integration into an IoT system on the cloud. This system allows researchers to check in on their experiments from anywhere in the world, enabling a variety of remote biology applications.

1.1. Summary

The Picroscope system is comprised of a cluster of network connected devices. A pipeline was developed to facilitate remote operation and to control communication between modules in the system. The result is a web based interface that allows users to control a longitudinal imaging experiment and view results in near real time. This brings high throughput parallel remote microscopy to a price point affordable in many sectors that could not previously access such systems. The device captures stacks of images by sweeping the cameras up and down, taking images at different points along the z-axis. This data type is known as a z-stack. The 3D z-stack image data captured by the system allows it to image both 2D monolayer cell cultures and 3D samples. Our data pipeline is capable of feeding these z-stacks into software that generates Extended Depth of Field (EDoF) composite images [30] to simplify the end user's visual analysis of longitudinal changes in a 3D sample. In this paper we demonstrate the system's functionality with frog embryos, zebrafish, and human cerebral cortex organoids.

2. System design

2.1. Overview

The Picroscope contains several custom boards and 3D printed pieces. These are shown in Fig. 1, the main pieces include the 24 well plate holder, the elevator stage camera array, and the LED illumination boards (one above the culture plate, and one below). The camera array consists of a 6×4 grid of sensors with lenses attached. Two stepper motors are used to raise and lower the camera stage in order to move the focal plane. More detail on the hardware design of the picroscope can be found in [29].

The basic workflow for this system is illustrated in Fig. 2. Experiments are triggered through our web based control console (Fig. 3). In the console, the user sets the following parameters: experiment id, number of pictures in z-stack, distance between layers, initial offset distance, light type (Over-the-plate or Under-the-plate), and any additional camera control parameters allowed through the raspistill library [31]. These parameters are passed to the Picroscope through a cloud based messaging service using the Message Queue Telemetry Transport protocol (or MQTT) [32]. Experiment parameters can also be changed on the fly during the course of an experiment through the same console.

An imaging event captured in our system consists of one z-stack per active camera. At the conclusion of each captured event, the Picroscope uploads the results to an S3 Object Store [33] on a server where the pictures become accessible through our image viewer website (Fig. 3).

The image viewer website allows a user to select any well and explore the image set as it changes over time, any individual timestep can also have its z-plane position changed giving the user a virtual focus knob. Even though all 24 cameras are physically connected such that at any given instant they are all looking at the same plane, the image viewer allows users to simultaneously investigate different wells at different focal planes. The result is 24 independent virtual microscopes. This is a benefit of presenting previously captured data. Users can view the most recent images with individual focus control giving near real time views of the contents of each well. This is a considerable cost savings over having a classroom setup with 24 physical microscopes. Students have the experience of controlling their own independent microscope from their mobile phone. This illustrates a unique advantage of computer interfaced remote experimentation.

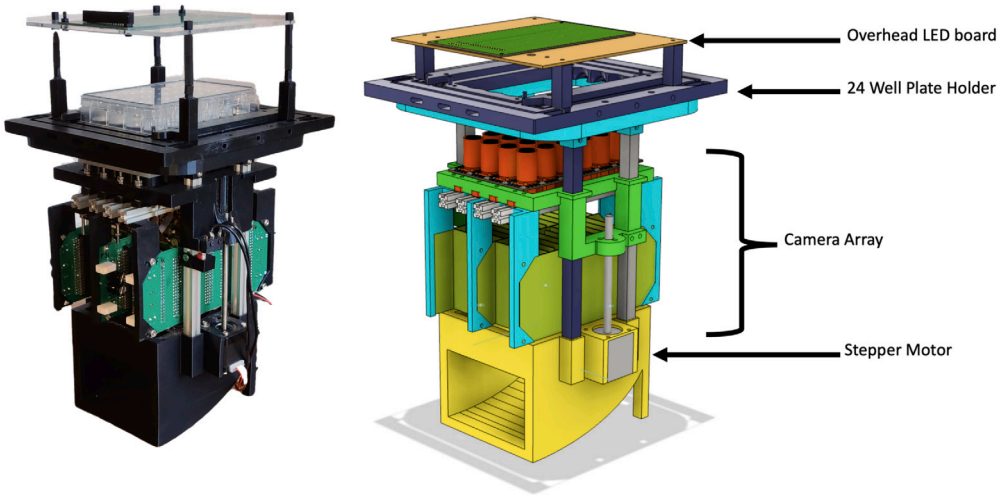

Fig. 1. The Picroscope: The left shows a fully assembled Picroscope. On the right is the computer aided design(CAD) render of the Picroscope. The main components of the picroscope include the over-the-plate illumination board, 24 well cell plate holder with XY stage, and a 24 camera array with stepper motors for vertical movement.

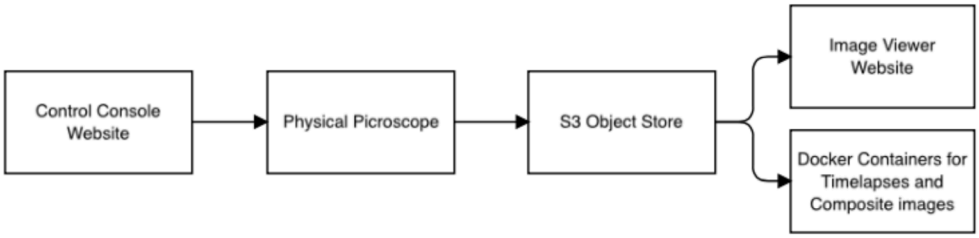

Fig. 2. Basic workflow from user input to experiment output: Control Console (Fig. 3), Physical Picroscope (Fig. 4), Image Viewer Website (Fig. 3), Docker Container (Fig. 8).

Fig. 3. Web Interface: Left: Control Console which allows us to set parameters and control the experiment, Right: Image Viewer where users can view their samples remotely.

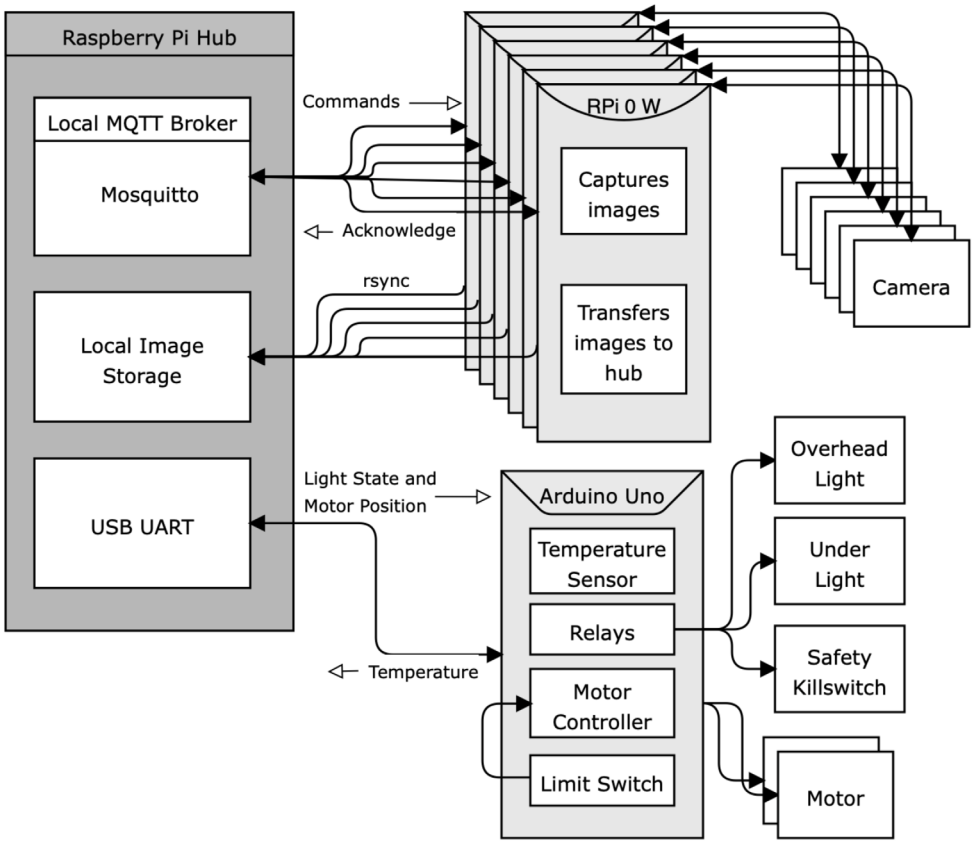

Fig. 4. Data flow between local hardware network:

2.1.1. Device hardware

In addition to the custom pieces previously described, the Picroscope device is comprised of a number of connected sub-systems (Fig. 4). Top level control is handled through the Raspberry Pi 4 based "hub" which communicates with 24 Raspberry Pi Zeros and an Arduino Uno. Fig. 4 illustrates the communication between these devices.

To gather a data point for an experiment, the hub sends commands to 24 Raspberry Pi Zero Ws each of which control one camera. The hub also connects to an Arduino Uno that is responsible for controlling motors and lights. In order to take a picture, the hub pi turns on the light and sends a command containing the input parameters for the raspistill camera API [31] to the pi zeros. During a z-stack capture, pictures are taken and stored on the individual pi zeros. Each Raspberry Pi Zero sends a message back to the hub when its picture has been taken. When all cameras have finished taking pictures, the camera stage is moved upwards by the motors, at which point the next layer of the z stack begins. At the conclusion of the z-stack capture, the stage lowers back to its starting position and the hub sends a command to each camera to begin transferring the images to the hub.

2.2. Communication between system layers

The flow of messages and data in our pipeline is represented in Fig. 5.

The control console webpage (Fig. 3) communicates with our system using the MQTT message protocol. MQTT is a publish/subscribe based protocol in which a message "broker" transfers any messages published on a given topic to all the subscribers of that topic. To pass messages from the webpage to the picroscope, we use a cloud based MQTT broker provided by Amazon IoT. Every picroscope has a unique id and "device shadow" on the Amazon IoT platform. The control console website has access to the device list through Amazon's IoT API. The console displays a list of all active systems with buttons to control each one (Fig. 3A). When a command is sent from the console it is published with the topic being the id of the picroscope we wish to control.

The targeted picroscope receives the start command along with the desired experiment parameters. Parameters can be adjusted on the fly from the control console website. Each picroscope then uses its own locally hosted MQTT broker as a message bus to pass commands to the 24 raspberry pi zero Ws that each control a camera. Each hub also connects through USB to an Arduino Uno which is responsible for controlling the motors and lights as well as temperature safety monitoring and emergency shutoff. Commands to take a picture can be sent to individual cameras or all of them at once. When a camera finishes taking a picture, it sends a message

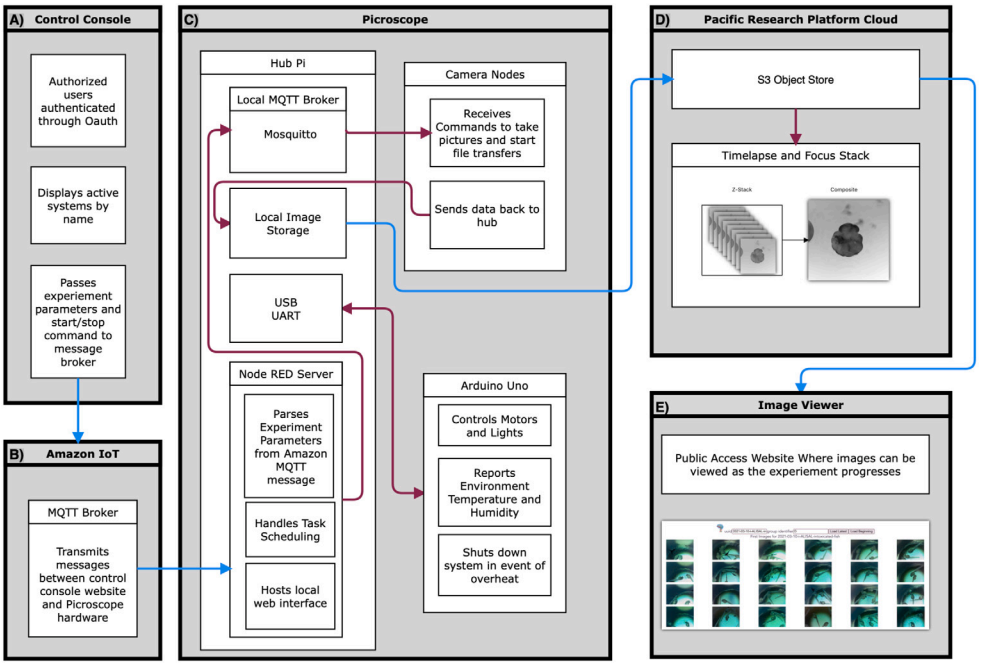

Fig. 5. Message and data flow through the system: Red lines represent communication within a local unit, Blue lines represent communication between physically separate subsystems. (A) Control Console, (B) Amazon IoT, (C) Physical Picroscope, (D) Cloud, (E) Image Viewer.

back to the hub with its camera id, allowing the hub to know when all cameras have finished. Unused wells can be disabled by the hub, allowing a higher maximum throughput for the other enabled cameras.

When a z-stack capture concludes, the pi zeros need to send their data to their assigned hub pi. To accomplish this, we use a custom queuing protocol that initiates file transfers individually on each pi zero W and continues to the next pi when the current transfer finishes or a timeout condition is reached. The protocol is detailed in Fig. 6. This queuing system results in higher throughput than simply starting all transfers in parallel and the queue is not disrupted in the case of non-responsive pi zeros. When the data transfer completes, the result is uploaded to an s3 object store on cloud hardware run on the Pacific Research Platform (or PRP) [34]. The time required for the transfer to complete is the primary limiting factor in determining the maximum data capture frequency we can achieve. Transfer time is primarily determined by z-stack size and number of active cameras. Using smaller z-stacks, or less cameras allows higher maximum throughput while maintaining parallel image capture for each well. This is an important consideration for imaging samples displaying higher frequency dynamics. With 24 cameras capturing 10 layer z-stacks, we are able to capture an entire new z-stack approximately 4 times per hour. If it is necessary to capture higher frequency dynamics, the picroscope can be set to capture short videos instead of z-stacks.

Once uploaded to s3, the images are accessible over the Internet. When an experiment is started, the picroscope generates a file called “manifest.json”. The manifest serves as a text based map indicating where in the s3 object store each image is stored. The manifest is updated with a new timestamp every time a new z-stack is captured. The manifest can be interpreted in such a way that you can generate the URLs for every picture without needing to query the object store (querying the object store takes a substantial amount of time). The image viewer website interprets the manifest to generate an interactive display of the images from a given experiment id. The manifest is also used when pulling the data into our dockerized scripts which we use to generate timelapse videos, focus stacked composite images and perform other image analysis.

A simplified view of the system has 3 distinct interface layers, each of which can be interacted with in separate locations from each other during an experiment. These are the Physical, Control, and Monitoring layers. They are represented in Fig. 7. The separation of these layers means that one can place a physical sample on the scope while elsewhere someone else remotely sets the control parameters of an experiment for the viewing needs of a third separate group. All layers could likewise be interacted with by the same person if that better serves the needs of the experiment

2.2.1. Timelapse processing and Extended Depth of Field

In addition to being viewable through our web interface, the pictures on the server are can be fed into scripts for generating focal plane stacked composite images with FIJI [35] using the Extended Depth of Field plugin [30]. This plugin allows us to generate a single image containing the best focused features from each layer of the z-stack. We can also use a script to generate timelapse videos from experiments. Generating timelapse videos with focus stacked frames allows for easy visual analysis of longitudinal changes in 3 dimensional samples.

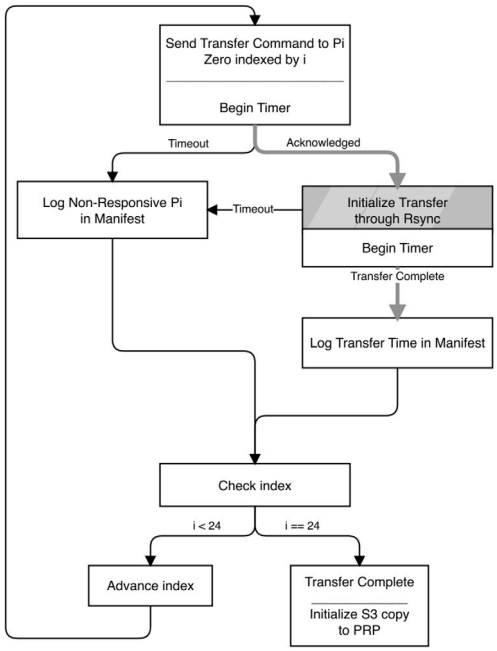

Fig. 6. File Transfer Queuing Protocol: White boxes with black arrows represent actions taken by the Pi Zeros, Grey Box and arrows indicate messages and actions taken by Pi Zero, Black arrows and Boxes indicate messages and actions taken by the Hub Pi.

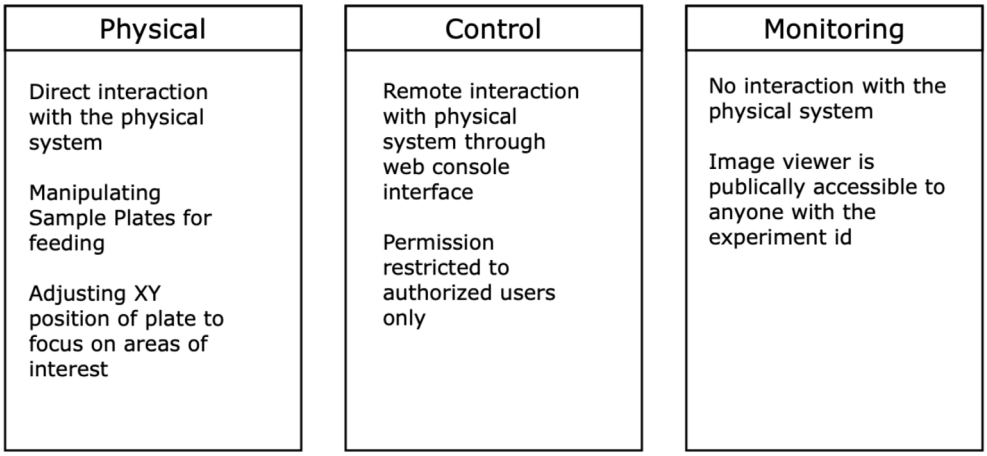

Fig. 7. Separate Interface Layers:.

The containerization of these programs with Docker allows us to run them in an automated fashion and easily deploy them with cloud service providers. [36]

3. Applications

3.1. In incubator application: Visualizing 3D Human Cortical Organoids

This experiment utilized a picroscope placed in an incubator at UC Santa Cruz, monitoring of results and control of the device was performed by users remotely primarily from their homes in Santa Cruz

Cortical organoids generated from aggregates of pluripotent stem cells have been shown to recapitulate aspects of early embryonic brain development, providing an in vitro model for studying species specific brain development in organisms including humans [37,38]. Here, human cerebral organoids were generated and plated on laminin coated 24 well plates and allowed to adhere to the surface. Outgrowths and cellular migration were restricted to the 2D plane of the plate's surface and captured/monitored over

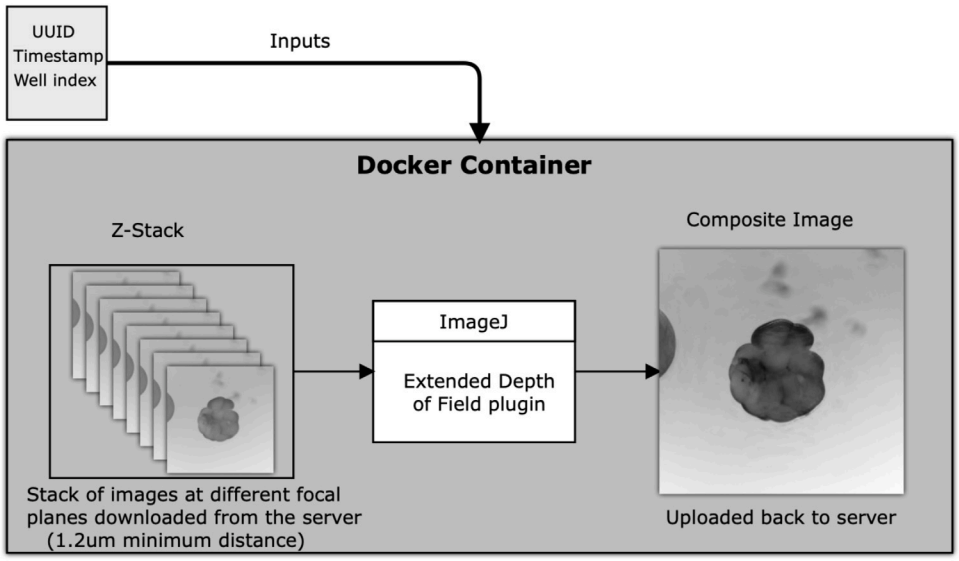

Fig. 8. Extended Depth of Field: Docker Containers are used to generate composite images with FIJI.

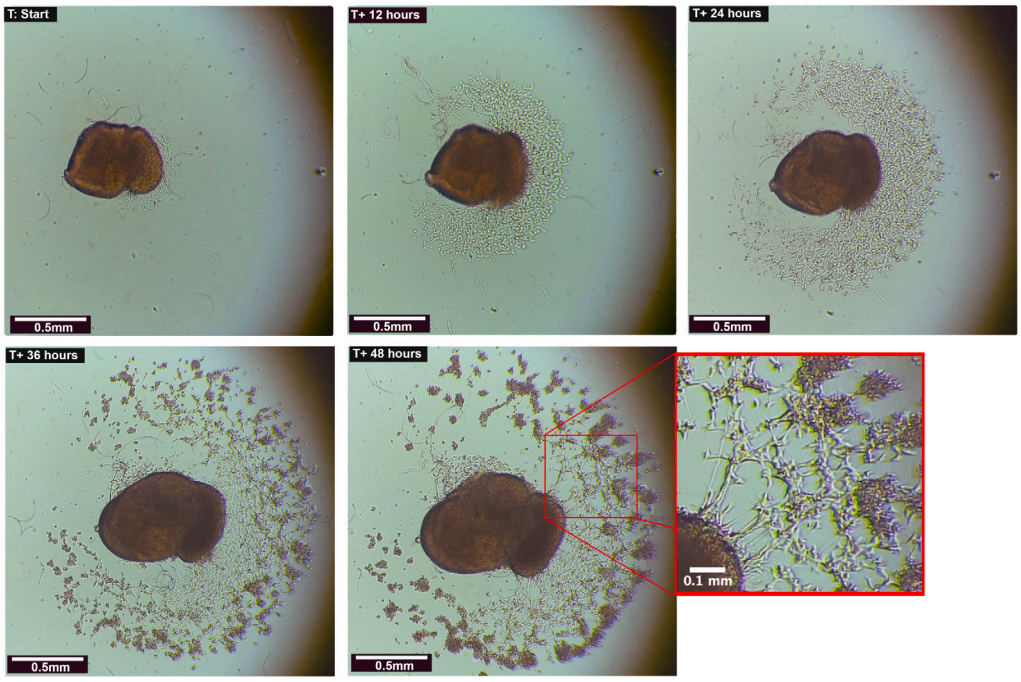

Fig. 9. Human Cortical Organoid with Outgrowths: Organoids were plated on laminin coated 24 well plates and allowed to adhere to the surface. Outgrowths and cellular migration were restricted to the 2D plane of the plate's surface and captured/monitored over the course of 21 days. Significant morphological changes occurred within the first 48 h after imaging began.

the course of 21 days with the entire Picroscope device inside a temperature and humidity controlled CO2 incubator. Samples from that experiment can be seen in Fig. 9

3.2. Live whole organism imaging: Zebrafish

In a pilot program, a picroscope was used in a project for AP biology students at a local high school. These students were taking remote classes and did not have access to in-person lab facilities. The students used the Picroscope to measure survival and behavioral changes of zebrafish under the influence of varying concentrations of exogenous chemicals including caffeine and

Fig. 10. Zebrafish Circulation Video included in supplemental material.

ammonia. In terms of the interface layers described in Fig. 7 The physical interaction occurred at UC San Francisco. Remote control was performed by class facilitators 80 miles away at UC Santa Cruz. And the students themselves monitored the resulting data from their homes in Salinas, 110 miles away from the physical device.

In the set up phase of this experiment, we captured video showing fluid circulation inside a live zebrafish (Fig. 10). This demonstrates the video capture capability which can be used to observe higher frequency dynamics than would be possible with z-stack image capture.

Feedback regarding the student experience was gathered using surveys with free response answers to questions. All users responded very positively to questions regarding usability, interest, and excitement in doing remote scientific experiments and indicated interest in pursuing future projects with the Picroscope. This experiment demonstrates the educational potential of remotely managing live whole organism studies on the Picroscope.

3.3. Extended Depth of Field functionality: Frog embryos

In another study controlled remotely from Santa Cruz with physical hardware set up at UC San Francisco, we monitored development of *Xenopus tropicalis* frog embryos into larvae. We were able to image all stages of development as the embryo grew into a moving larva. Figs. 11 and 12 show 2 interesting periods of development with different time scales of change. At the conclusion of this experiment, each z-stack timestep was fed through the Extended Depth of Field plugin in FIJI [30]. Running FIJI through a docker container allowed the process to be scripted and run on a remote server. We then generated a timelapse video (Fig. 13) of these composite images. Each frame contains the in focus pieces of each of the 10 layers in the z-stack. When the embryo develops into the larval stage, it starts to move. The movement causes visible artifacts to appear in that section of the video, since the larvae moves between layer captures, demonstrating a drawback of this approach when imaging moving organisms.

4. Discussion

The development of remote capability for this device offered many unique challenges from a design perspective. When the system is reliant on the internet for reporting what is going on with a cell culture, loss of contact can mean many different things. These can range from a short term internet outage to a catastrophic system failure. Loss of contact in this way presents major issues if one were to deploy the system in an unmanned remote location. We attempt to address this issue by anticipating the various failure modes and setting up external monitoring systems that can alert us if they occur and in some cases take automatic action.

In the event of a system failure, it is critically important to ensure that the issue does not lead to a loss of the biology experiment on top of it. Implementing failsafes for potential hardware issues is important. Earlier iterations of the picroscope caused incubator environments to overheat and cook the cell cultures. It can be quite frustrating for a researcher to spend time growing and plating a culture, only for an instrument to cook it. This scenario led to the implementation of a overheat safety shutoff system, a temperature and humidity sensor was added to the picroscope. On the software side, a user can select and modify the safety shutoff temperature

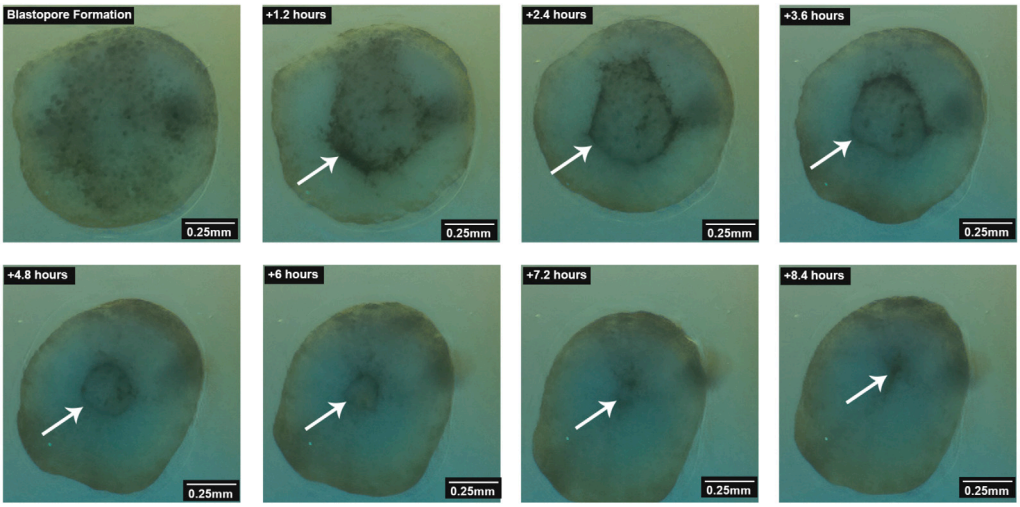

Fig. 11. Blastopore Closure Selected images during blastopore closure, these are samples from every 5 timesteps during this stage. The video in Fig. 13 shows all frames.

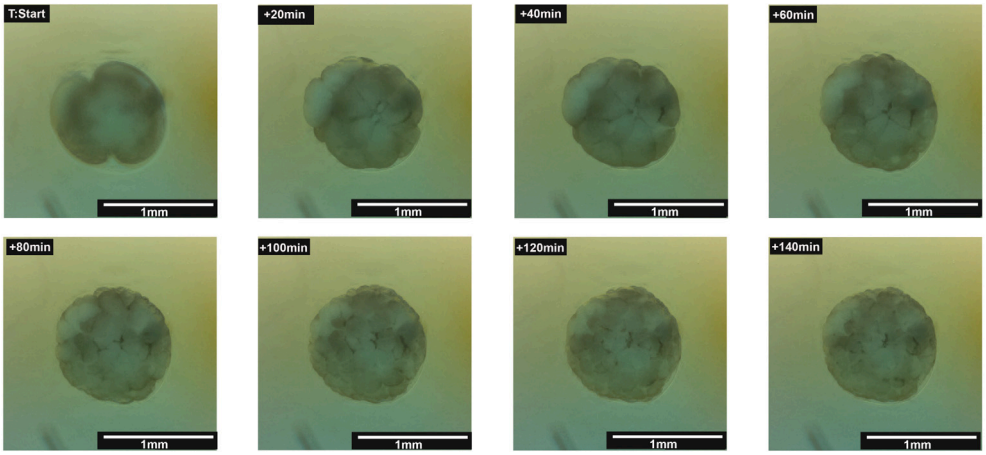

Fig. 12. Early Cell Divisions The first captured images in our data set show cell divisions as the embryo becomes more complex and individual cells shrink in size.

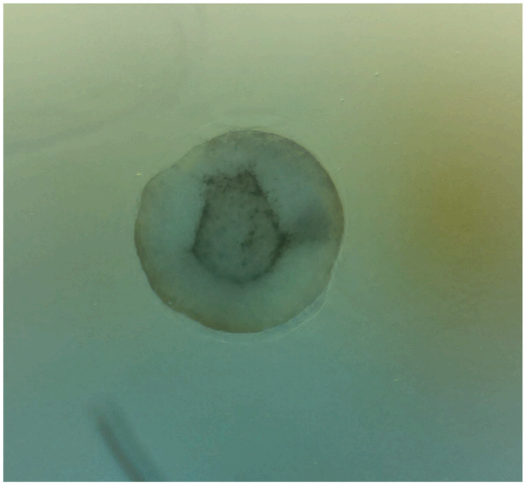

Fig. 13. Frog embryo development timelapse video. Video included in supplemental material.

remotely and if that threshold is reached, all power to the system is shut off. The hub pi remains powered as it is located outside of the incubator environment. The hub remaining active allows the user to remotely reset the system if it has been deemed safe to continue.

In the event of a more minor issue, such as an internet outage, the system is designed to continue operating with only the local wireless network. In this state, users can no longer change experiment parameters, but the system will continue capturing data and storing it locally on the hub pi until the internet comes back, at which point all data gathered during the outage will be uploaded to the server

While we have described the efforts taken to ensure the system does not cause interference with experiments, it should be noted that the regular operation of this system can actually save experiments that are going wrong for other reasons. Using a picroscope to monitor a culture remotely means that the user can notice issues with the cells sooner and be more likely to be able to take corrective action.

The IoT framework described here, is system specific for the picroscope. Currently our research group has been developing a general use IoT cloud framework for running and monitoring several different pieces of lab equipment including the picroscope. This includes a container orchestration environment for efficiently running our image processing docker containers in an automated fashion. This framework is described in [39].

5. Conclusion

The Covid-19 pandemic changed the work landscape for many of us. The development of the Picroscope was highly motivated by the access limitations we were presented with. The resulting system has helped our research group continue to produce work during this difficult time period.

The Picroscope has been designed from the ground up as an extensible platform. Development of various compatible add-ons are in progress for new features including fluorescence microscopy. Our end goal is a general use parallel experiment system allowing remote control, sample manipulation, feeding, and imaging.

With this system we have provided a low cost solution (costing 88$ per well) [29] for biologists to work remotely with greater ease. The result is a sensor-per-well parallel imaging system capable of brightfield microscopy that can be deployed inside a standard CO2 incubator. By having one camera per well, we have an array of microscopes available to researchers allowing them to remotely monitor the development of the biological samples over a long period of time.

Having access to this system allows researchers to easily monitor long term morphological changes in their cell cultures without needing to interfere with their incubator environments. Using Picroscopes also allows for seamless collaboration between researchers at different institutions, allowing them to easily compare cultures as they grow. We envision deployment of many of these systems at once in our lab and collaborator's labs to help push us into an interconnected open source bio-lab of the future.

Anybody interested in building a picroscope system will find resources to do so at our research group's website http://braingeneers.gi.ucsc.edu/.

CRediT authorship contribution statement

Pierre V. Baudin: Worked on hardware and software of the Picroscope, Performed Biological experiments, Wrote the manuscript with contributions from all authors. **Victoria T. Ly:** Worked on hardware and software of the Picroscope, Performed Biological experiments, Wrote the manuscript with contributions from all authors. **Pattawong Pansodtee:** Worked on hardware and software of the Picroscope. **Erik A. Jung:** Worked on early prototypes. **Robert Currie:** Worked on the web console pipeline. **Ryan Hoffman:** Performed Biological experiments, Conceived the experiments. **Helen Rankin Willsey:** Performed Biological experiments, Conceived the experiments. **Alex A. Pollen:** Supervised the team and secured funding. **Tomasz J. Nowakowski:** Supervised the team and secured funding. **David Haussler:** Supervised the team and secured funding. **Mohammed A. Mostajo-Radji:** Performed Biological experiments, Conceived the experiments, Supervised the team and secured funding, Wrote the manuscript with contributions from all authors. **Sofie R. Salama:** Supervised the team and secured funding. **Mircea Teodorescu:** Supervised the team and secured funding, Wrote the manuscript with contributions from all authors.

Declaration of competing interest

One or more of the authors of this paper have disclosed potential or pertinent conflicts of interest, which may include receipt of payment, either direct or indirect, institutional support, or association with an entity in the biomedical field which may be perceived to have potential conflict of interest with this work. For full disclosure statements refer to https://doi.org/10.1016/j.iot.2021.100454. The authors have written patents covering the technology described in this article. A.A.P. is on the board of Herophilus. The authors declare no other conflict of interest.

Acknowledgments

This work is supported by the Schmidt Futures Foundation SF 857 (D.H.). Research reported in this publication was also supported by the National Institute Of Mental Health of the National Institutes of Health under Award Number R01MH120295 (S.R.S.) and the National Science Foundation under award number NSF 2034037 (M.T.). We would like to thank Jeremy Linsley and Wiktoria Leks for providing us zebrafish for this study. In addition, we would like to thank Arnar Breevoort for providing experimental support. H.R.W. was supported by grant U01MH115747-01 from NIMH to Matthew State. M.A.M.-R. was partially supported by grant TL1 TR001871 from the NIH National Center for Advancing Translational Sciences. D.H. is an investigator with the Howard Hughes Medical Institute.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.iot.2021.100454.

References

[1] K.R. Myers, W.Y. Tham, Y. Yin, N. Cohodes, J.G. Thursby, M.C. Thursby, P. Schiffer, J.T. Walsh, K.R. Lakhani, D. Wang, Unequal effects of the COVID-19 pandemic on scientists, Nat. Hum. Behav. 4 (9) (2020) 880–883.
[2] F. Sayre, A. Riegelman, The reproducibility crisis and academic libraries, Coll. Res. Libr. 79 (1) (2018) 2.
[3] M. Baker, Biotech giant publishes failures to confirm high-profile science, Nature 530 (7589) (2016) 141, http://dx.doi.org/10.1038/nature.2016.19269, URL http://www.nature.com/articles/nature.2016.19269.
[4] B. Miles, P.L. Lee, Achieving reproducibility and closed-loop automation in biological experimentation with an IoT-enabled lab of the future, SLAS Technol. Transl. Life Sci. Innov. 23 (5) (2018) 432–439, http://dx.doi.org/10.1177/2472630318784506.
[5] G. Lippi, G. Da Rin, Advantages and limitations of total laboratory automation: a personal overview, Clin. Chem. Lab. Med. (CCLM) 57 (6) (2019) 802–811, http://dx.doi.org/10.1515/cclm-2018-1323.
[6] B.T. Schmidt, J.M. Feduska, A.M. Witt, B.M. Deasy, Robotic cell culture system for stem cell assays, in: T. Evans (Ed.), Ind. Robot Int. J. 35 (2) (2008) 116–124, http://dx.doi.org/10.1108/01439910810854601.
[7] T. Aidukas, R. Eckert, A.R. Harvey, L. Waller, P.C. Konda, Low-cost, sub-micron resolution, wide-field computational microscopy using opensource hardware, Sci. Rep. 9 (1) (2019) 1–12, http://dx.doi.org/10.1038/s41598-019-43845-9, URL https://www.nature.com/articles/s41598-019-43845-9.
[8] A. Maia Chagas, L.L. Prieto-Godino, A.B. Arrenberg, T. Baden, The €100 lab: A 3D-printable open-source platform for fluorescence microscopy, optogenetics, and accurate temperature control during behaviour of zebrafish, Drosophila, and Caenorhabditis elegans, Plos Biol. 15 (7) (2017) e2002702, http://dx.doi.org/10.1371/journal.pbio.2002702.
[9] T. Baden, A.M. Chagas, G. Gage, T. Marzullo, L.L. Prieto-Godino, T. Euler, Open Labware: 3-D printing your own lab equipment, PLoS Biol. 13 (3) (2015) e1002086.
[10] S. Ornes, Core concept: The Internet of Things and the explosion of interconnectivity, Proc. Natl. Acad. Sci. 113 (40) (2016) 11059–11060.
[11] Z. Chaczko, R. Braun, Learning data engineering: Creating IoT apps using the node-RED and the RPI technologies, in: 2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET), IEEE, Ohrid, Macedonia, 2017, pp. 1–8, http://dx.doi.org/10.1109/ITHET.2017.8067827, URL http://ieeexplore.ieee.org/document/8067827/.
[12] Node- RED, URL https://nodered.org/.
[13] D.B.C. Lima, R.M.B. da Silva Lima, D. de Farias Medeiros, R.I.S. Pereira, C.P. de Souza, O. Baiocchi, A performance evaluation of raspberry pi zero w based gateway running MQTT broker for IoT, in: 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), IEEE, 2019, pp. 0076–0081.
[14] A. Patel, P. Devaki, Survey on NodeMCU and Raspberry pi: IoT, Int. Res. J. Eng. Technol. (IRJET) 6 (4) (2019) 5101–5105.
[15] M. May, A DIY approach to automating your lab, Nature 569 (7754) (2019) 587–589.
[16] M. Meyer, Build your own lab: Do-it-yourself biology and the rise of citizen biotech-economies, J. Peer Prod. 2 (online) (2012) 4–p.
[17] J.D. Brock, R.F. Bruce, M.E. Cameron, Changing the world with a Raspberry Pi, J. Comput. Sci. Coll. 29 (2) (2013) 151–153.
[18] P. Abrahamsson, S. Helmer, N. Phaphoom, L. Nicolodi, N. Preda, L. Miori, M. Angriman, J. Rikkilä, X. Wang, K. Hamily, et al., Affordable and energy-efficient cloud computing clusters: The bolzano raspberry pi cloud cluster experiment, in: 2013 IEEE 5th International Conference on Cloud Computing Technology and Science, 2, IEEE, 2013, pp. 170–175.
[19] J.a. Saffran, G. Garcia, M.A. Souza, P.H. Penna, M. Castro, L.F. Góes, H.C. Freitas, A low-cost energy-efficient raspberry pi cluster for data mining algorithms, in: European Conference on Parallel Processing, Springer, 2016, pp. 788–799.
[20] S.J. Cox, J.T. Cox, R.P. Boardman, S.J. Johnston, M. Scott, N.S. O'brien, Iridis-pi: a low-cost, compact demonstration cluster, Cluster Comput. 17 (2) (2014) 349–358.
[21] I. Gustavsson, J. Zackrisson, L. Håkansson, I. Claesson, T.L. Lagö, The VISIR project – an open source software initiative for distributed online laboratories, 2007, URL http://urn.kb.se/resolve?urn=urn:nbn:se:bth-9135.
[22] G. Alves, M. Marques, C. Viegas, M.C. Lobo, R. Barral, R. Couto, F. Jacob, C. Ramos, G. Vilão, D. Covita, J. Alves, P. Guimarães, I. Gustavsson, Using VISIR in a large undergraduate course: Preliminary assessment results, in: 2011 IEEE Global Engineering Education Conference (EDUCON), 2011, pp. 1125–1132, http://dx.doi.org/10.1109/EDUCON.2011.5773288, ISSN: 2165-9567.
[23] M. Blazquez-Merino, A. Macho-Aroca, P. Baizán-Álvarez, F. Garcia-Loro, E. San Cristobal, G. Diez, M. Castro, Use of VISIR remote lab in secondary school: Didactic experience and outcomes, in: M.E. Auer, R. Langmann (Eds.), Smart Industry & Smart Education, in: Lecture Notes in Networks and Systems, Springer International Publishing, Cham, 2019, pp. 69–79, http://dx.doi.org/10.1007/978-3-319-95678-7_8.
[24] B. Balamuralithara, P. Woods, Virtual laboratories in engineering education: The simulation lab and remote lab, Comput. Appl. Eng. Educ. 17 (1) (2009) 108–118, http://dx.doi.org/10.1002/cae.20186.
[25] R.K. Scheckler, Virtual labs: a substitute for traditional labs? Int. J. Dev. Biol. 47 (2–3) (2003) 231–236.
[26] M.G. Jones, T. Andre, R. Superfine, R. Taylor, Learning at the nanoscale: The impact of students' use of remote microscopy on concepts of viruses, scale, and microscopy, J. Res. Sci. Teach. Off. J. Natl. Assoc. Res. Sci. Teach. 40 (3) (2003) 303–322.
[27] C. Wallace, C. Conway, A. Ray, S. Robinson, Remote-access scanning electron microscopy for K-12 students: The bugscope project nine years on, Microsc. Microanal. 14 (S2) (2008) 858–859, http://dx.doi.org/10.1017/S1431927608087461, URL https://www.cambridge.org/core/journals/microscopy-and-microanalysis/article/abs/remoteaccess-scanning-electron-microscopy-for-k12-students-the-bugscope-project-nine-years-on/1E6B1FE0EBCAC984DFCFED84C37FEBE7.

[28] Z. Hossain, E.W. Bumbacher, A.M. Chung, H. Kim, C. Litton, A.D. Walter, S.N. Pradhan, K. Jona, P. Blikstein, I.H. Riedel-Kruse, Interactive and scalable biology cloud experimentation for scientific inquiry and education, Nature Biotechnol. 34 (12) (2016) 1293–1298.
[29] V.T. Ly, P.V. Baudin, P. Pansodtee, E.A. Jung, K. Voitiuk, Y. Rosen, H.R. Willsey, G.L. Mantalas, S.T. Seiler, J.A. Selberg, et al., Development of a low-cost system for simultaneous longitudinal biological imaging, BioRxiv (2021).
[30] B. Forster, D. Van De Ville, J. Berent, D. Sage, M. Unser, Extended depth-of-focus for multi-channel microscopy images: A complex wavelet approach, in: Proceedings of the Second IEEE International Symposium on Biomedical Imaging: From Nano To Macro (ISBI'04), Arlington VA, USA, 2004, pp. 660–663.
[31] Raspistill - Raspberry Pi Documentation, URL https://www.raspberrypi.org/documentation/usage/camera/raspicam/raspistill.md.
[32] D. Locke, Mq telemetry transport (mqtt) v3. 1 protocol specification, IBM Dev. Tech. Libr. 15 (2010).
[33] What is Amazon S3? - Amazon Simple Storage Service, URL https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html.
[34] L. Smarr, C. Crittenden, T. DeFanti, J. Graham, D. Mishin, R. Moore, P. Papadopoulos, F. Würthwein, The pacific research platform: Making high-speed networking a reality for the scientist, in: Proceedings of the Practice and Experience on Advanced Research Computing, 2018, pp. 1–8.
[35] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, et al., Fiji: an open-source platform for biological-image analysis, Nature Methods 9 (7) (2012) 676–682.
[36] D. Merkel, Docker: Lightweight Linux containers for consistent development and deployment, Linux J. 2014 (239) (2014).
[37] M. Eiraku, K. Watanabe, M. Matsuo-Takasaki, M. Kawada, S. Yonemura, M. Matsumura, T. Wataya, A. Nishiyama, K. Muguruma, Y. Sasai, Self-organized formation of polarized cortical tissues from ESCs and its active manipulation by extrinsic signals, Cell. Stem. Cell. 3 (5) (2008) 519–532.
[38] M.A. Lancaster, J.A. Knoblich, Generation of cerebral organoids from human pluripotent stem cells, Nat. Protoc. 9 (10) (2014) 2329–2340.
[39] D.F. Parks, K. Voitiuk, J. Geng, M.A. Elliott, M.G. Keefe, E.A. Jung, A. Robbins, P.V. Baudin, V.T. Ly, N. Hawthorne, D. Yong, S.E. Sanso, N. Rezaee, J. Sevetson, S.T. Seiler, R. Currie, K.B. Hengen, T.J. Nowakowski, S.R. Salama, M. Teodorescu, D. Haussler, Internet of things architecture for high throughput biology, BioRxiv (2021) http://dx.doi.org/10.1101/2021.07.29.453595.