LOW-COST SECURITY SYSTEM FOR A UIC LABORATORY USING MOTIONEYEOS OPEN-SOURCE SECURITY SOFTWARE AND RASPBERRY PI SINGLE-BOARD COMPUTERS

Anthony Dudlo[1]
[1]University of Illinois at Chicago
College of Mechanical and Industrial Engineering
Chicago, IL
adudlo@outlook.com

The University of Illinois at Chicago College of Mechanical and Industrial Engineering expands and adds value to science by performing and publishing novel research. Research at any of UIC's laboratories generally represents significant investments of time and money, is usually an individual or entity's intellectual property, and may even be a health hazard. As such, all laboratories are kept under lock and key. If a laboratory is left unlocked, it is prudent to have a redundant security system to inform the laboratory administrator of potential security breaches.

A motion-activated camera system is a reasonable means of laboratory security, but can also be expensive, with high startup costs and monthly maintenance fees. With a little work, it is possible to implement a low-cost, low-maintenance security camera system with many mainstream features that would cost a premium if purchased at retail price.

MotionEyeOS is an open-source Linux distribution that can be installed on single board computers. This software can be used to make standalone security camera or large security camera networks that broadcast over internet protocol to a server. MotionEyeOS has the following noteworthy features: multi-camera functionality, local storage of images and videos, stream over internet protocol, upload-to-cloud functionality, motion detection, and working schedule. When broadcasting over internet protocol, a system of wirelessly connected single-board computer camera units running MotionEyeOS can provide inexpensive active and passive security comparable to professionally managed security systems. This procedure illustrates the steps necessary to build and implement such a system.

Raspberry Pi Zero W computers are ideal camera platforms for a MotionEyeOS security system because they are inexpensive, easy to package, and use a wireless local area network card for an internet connection. A Raspberry Pi 3B is a suitable camera server due to a bank of four USB ports, a graphics processor, a wireless local area network card to connect with the Pi Zero W cameras, and an ethernet port for optimal internet connectivity. The Pi 3B server can double as a security camera because the board features two camera connections. The small nature of these devices makes them inconspicuous and easy to install wherever is convenient. Not accounting for camera housings, a system of two Pi Zero W cameras broadcasting to a Pi 3B server/camera combination costs approximately $145.

Keywords: MotionEyeOS, Raspberry Pi, Security Camera

1. INTRODUCTION

Domestic and professional security have been critical to individuals, communities, and nations since presumably the dawn of mankind. Security solutions have included but not been limited to animal husbandry, communal constabularies, military establishments, and so on. The advent of wireless cellular technology changes the pace of security around the globe, as individuals can, for a premium of time or money, secure their homes and workplaces with sophisticated automated systems that nearly make guard personnel obsolete.

This procedure is an inexpensive demonstration of how to use readily available developer equipment to implement a three-camera security system capable of saving images and live video feeds to both local and cloud storage, motion sensitivity, and automatically notifying the administrator of possible security breaches via email.

2. MATERIALS

MotionEyeOS is a lightweight open source computer operating system purpose-built as a video security system and designed for use with single board computers. A single board computer is a small, inexpensive, complete computer built on one circuit board. Single board computers generally have the following features factory installed: processors, random access memory (RAM), input/output ports like universal serial bus (USB), and occasionally graphics processors, wireless and ethernet connectivity, and even Bluetooth. This procedure employs three single board computers running MotionEyeOS - two Raspberry Pi Zero W units serve as remote cameras, and one Raspberry Pi 3B unit that is a camera and a server.

2.1 Raspberry Pi Zero W

Each Raspberry Pi Zero W unit, or Pi Zero, is an independent "stream to internet protocol (IP)" camera running its own copy of MotionEyeOS. The Pi Zero is small – about half the size of a credit card. Figure 1 illustrates the Pi Zero's architecture, suggesting great versatility when implementing a security system.

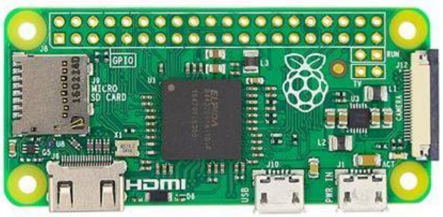


FIGURE 1: A RASPBERRY PI ZERO W SINGLE BOARD COMPUTER.

Each Pi Zero is a wireless local area network (LAN) and Bluetooth- equipped single board computer, and has a 1GHz single-core processor, 512MB of RAM, two micro USB ports, a mini high definition multimedia interface (HDMI) port, a micro camera slot and a micro secure digital (SD) card for storing the operating system.

The Pi Zero needs between 5 and 10 watts to run properly, which is a function of the system administrator's demand. If a Pi Zero is running both wireless and Bluetooth adapters, sending a signal to a monitor via HDMI, using a camera, mouse, and keyboard, the machine will draw more power than just running its wireless adapter and a camera. This means each unit can be powered by a spare mini USB cell phone charger without a problem.

2.2 Raspberry Pi 3B

The Raspberry Pi 3B, or "Pi 3B" for short, is a slightly larger and more capable single board computer when compared to the Pi Zero. Like the Pi Zero, the Pi 3B is equipped with LAN and Bluetooth- adapters, along with the micro SD card for the operating system. The Pi 3B also features a 1.2 GHz quad core processor, a graphics processing unit (GPU), two camera slots, a full gigabyte of RAM, a full size HDMI slot, four USB ports, a micro USB port, an ethernet port for hard-wiring into a network, a 40-pin general purpose input/output (GPIO) header, and a barrel jack.

Although the Pi 3B shown in Figure 2 has a tremendous array of features, the machine has power requirements similar to the Pi Zero and requires a 5-volt 2.5-amp power supply. This means each unit can be powered by a spare mini USB cell phone charger without a problem.

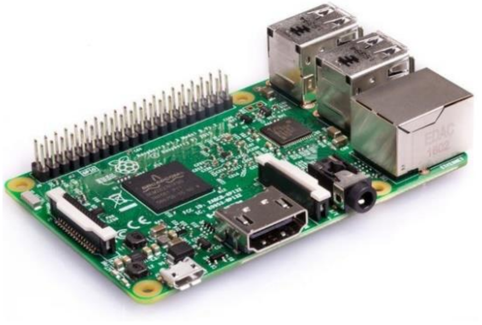


FIGURE 2: A RASPBERRY PI 3B SINGLE BOARD COMPUTER.

2.3 Raspberry Pi Camera

Raspberry Pi cameras, or Pi cameras for short, are generally inexpensive modules that connect to the Pi 3B and Pi Zero via ribbon cables. An example of a Pi camera is shown in Figure 3 with Pi 3B and Pi Zero specific connectors.

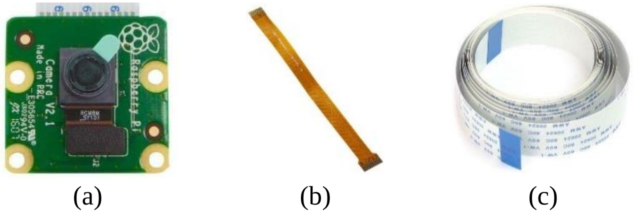


FIGURE 3: A RASPBERRY PI CAMERA (a) CONNECTS TO THE PI ZERO W USING A PROPRIETARY CONNECTION (b) AND CONNECTS TO THE PI 3B USING (c) A GENERIC CAMERA RIBBON CABLE.

Although inexpensive, Pi cameras are readily available at a variety of resolutions up to full high-definition 1080 progressive scan (1080P) and have a variety of configurations such as low light and infrared. These cameras are generally about the size of a quarter, making them unobtrusive and easy to package.

3. PROCEDURES

Programming knowledge and experience is not required for this project, but having a general understanding of Windows 10 and web searches is essential.

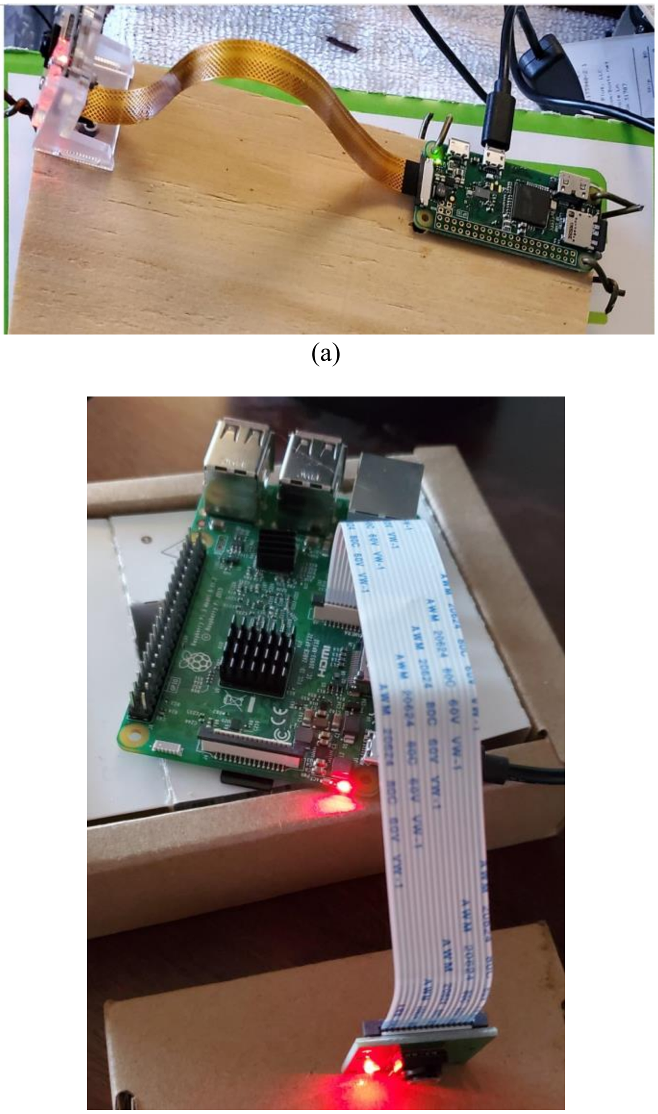
3.1 Procurement of Materials

All materials for this procedure can be readily purchased on mainstream websites like Amazon, Micro Center, and eBay. Beware that many retailers ration sales of the Pi Zero to one unit per customer per day due to artificial price ceilings. This project's price breakdown is in Table 1.

TABLE 1: ESTIMATED PROJECT BUDGET

Item	Quantity	Individual Cost ($)
Raspberry Pi Zero W	2	10
Raspberry Pi 3B	1	35
16 GB Micro SD Card	3	5
Raspberry Pi Power Supply	3	10
Raspberry Pi Camera and cables	3	15
Total Cost ($)		145

After obtaining all the components for this project, each camera unit must be assembled. Fully assembled Pi camera units are shown in Figure 4.



(a)

(b)

FIGURE 4: A PHOTOGRAPH OF (A) A RASPBERRY PI ZERO CAMERA UNITS AND (B) A RASPBERRY PI 3B CAMERA ROUTER COMBO. EACH UNIT CONSISTS OF A PI COMPUTER, PI CAMERA AND CAMERA HOLDER, PI CABLE, PI POWER SUPPLY, AND MICRO SD CARD

Due to size restrictions, the Pi Zero requires a proprietary cable to connect to a Pi Camera. The difference between the Pi camera cables is evident, as the smaller cable is gold and the wider cable is white. If desired, the Pi 3B can use an ethernet cable for better internet connectivity. After connecting the Pi cameras and power supplies to both Pi Zero units and the Pi 3B, the MotionEyeOS operating system must be imaged onto the micro SD cards.

3.2 Installing MotionEyeOS

Several software packages are used in this procedure. All software packages used are trivial to find online and readily available for free internet download. Software necessary to complete this procedure includes:

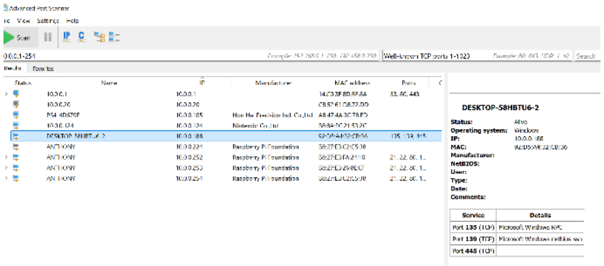
- MotionEyeOS operating system [1]
- SD Memory Card Formatter for Windows [2]
- Etcher disk imager [3]
- wpa_supplicant.conf [4]
- Advanced Port Scanner [5]
- PuTTY (optional) [6]

MotionEyeOS is available for free download in a GitHub repository on the internet. A quick web browser search will locate the MotionEyeOS repository.

Once downloaded, the MotionEyeOS software is installed on each micro SD. It is important to use the SD Memory Card Formatter for Windows program to first wipe all data from each micro SD card, even if the cards are brand new. MotionEyeOS will connect to the internet via wireless connection if and only if a special program entitled "*wpa_supplicant.conf*" is downloaded separately from GitHub, modified with the user's network ID and password, and mounted on each micro SD card prior to the system's initial startup. After mounting wpa_supplicant, the Etcher disk imager software mounts MotionEyeOS to each micro SD card and the system can be brought online. If plugged into a monitor, each camera system will launch with a typical Raspberry Pi screen showing launch progress, SD card partition data, and eventually an IP address. Once each camera is powered up and broadcasting, each individual IP address can be located as noted above by either plugging each unit into a monitor or by using a port scanning software.

3.3 Locating Each Camera on the Network

The quickest way to locate each camera on the network is to use the free-to-download program "Advanced Port Scanner." This software locates all computers using a network. Each Raspberry Pi unit will be named "Raspberry Pi Foundation" on the network, as shown in Figure 5.



0.0.0.1-254 | Well-known TCP ports 1-1023

Status	Name	IP	Manufacturer	MAC address	Ports
	10.0.0.1	10.0.0.1		14:C0:3F:8D:8F:8A	53, 80, 443
	10.0.0.20	10.0.0.20		C8:52:61:D8:72:DD	
	PS4-4D679F	10.0.0.105	Hon Hai Precision Ind. Co.,Ltd.	A8:47:4A:3C:78:FD	
	10.0.0.124	10.0.0.124	Nintendo Co.,Ltd	B8:8A:EC:21:53:2C	
	DESKTOP-58HBTU6-2	10.0.0.188		92:D5:A4:32:CB:36	135, 139, 445
	ANTHONY	10.0.0.224	Raspberry Pi Foundation	B8:27:EB:C2:C5:08	
	ANTHONY	10.0.0.252	Raspberry Pi Foundation	B8:27:EB:FA:21:10	21, 22, 80, 1...
	ANTHONY	10.0.0.253	Raspberry Pi Foundation	B8:27:EB:26:0E:CF	21, 22, 80, 1...
	ANTHONY	10.0.0.254	Raspberry Pi Foundation	B8:27:EB:C2:C5:90	21, 22, 80, 1...

DESKTOP-58HBTU6-2

Status: Alive
Operating system: Windows
IP: 10.0.0.188
MAC: 92:D5:A4:32:CB:36
Manufacturer:
NetBIOS:
User:
Type:
Date:
Comments:

Service	Details
Port 135 (TCP)	Microsoft Windows RPC
Port 139 (TCP)	Microsoft Windows netbios-ssn
Port 445 (TCP)	

FIGURE 5: A SCREENSHOT OF ADVANCE PORT SCANNER

Simply typing the IP address of each unit into an internet browser will bring the user a MotionEyeOS sign-in page, where the default login name is "admin" and the password is blank. Once signed in, the administrator can set static IP addresses so each camera's network address is constant.

3.4 Setting Up the Camera Hub

MotionEyeOS allows the administrator to add remote camera views to any camera on the network. Camera views are added to the hub's dashboard by inputting the previously discussed static IP addresses in under the camera hub's "Add a Camera" option. This allows the user to sign into a single master camera while still having complete access to each independent camera on the network. It is recommended to implement such a hub on the most powerful unit in the system for performance purposes. For this procedure, the Pi 3B is the default choice for a camera hub due to superior hardware. The MotionEyeOS dashboard showing all three cameras broadcasting over IP is shown in Figure 6. Note the local IP address in the browser search bar. This can be saved to favorites for easy viewing of all three cameras with the push of a button. The overlay containing the time and camera name is configurable for each camera, as is each camera's name, frame rate, orientation, and motion sensitivity. Note the change in overlay text in Figure 7. Each camera can also be configured individually to store data to fit the administrator's requirements.

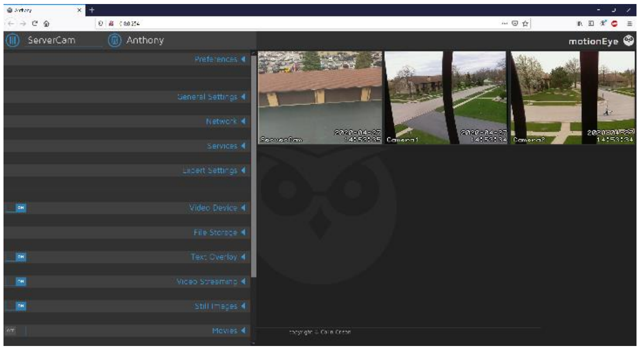


FIGURE 6: A SCREENSHOT OF THE HUB SHOWING THREE SECURITY VIDEO STREAMS.

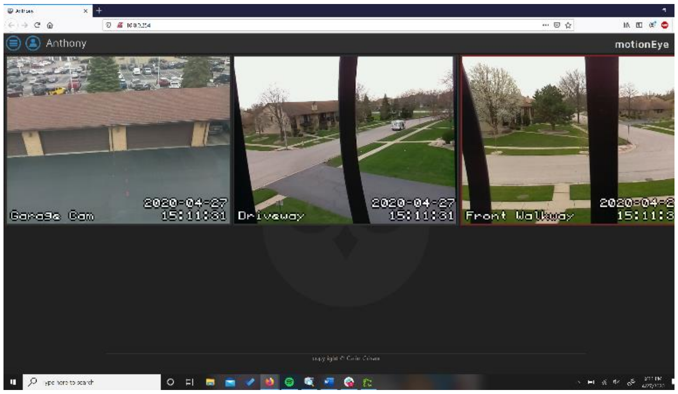
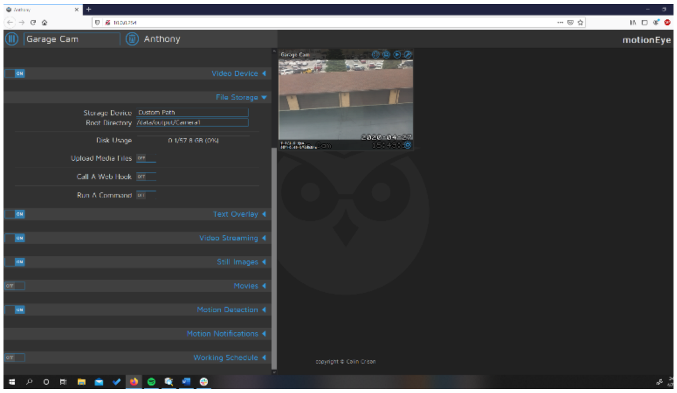


FIGURE 7: A SCREENSHOT OF THE HUB SHOWING CUSTOMIZED VIDEO TEXT OVERLAYS.

3.5 Storing Video and Images Locally

MotionEyeOS allows the administrator to store video footage locally in several convenient methods, and this procedure focuses on two. The camera's operating system is lightweight and takes up minimal disk space, so it is possible to store footage and images directly on each unit's micro SD card. Selecting the SD card's name in the "File Storage" tab, while leaving the "Root Directory" unchanged will begin storing footage and images directly to a partition specifically set aside for data storage. If the administrator desires a specific name for each data file, changing the "Storage Device" setting to "Custom path" will allow for the creation of a new local folder to save data. In this context, the "Root Directory" is "/data/output/Camera1", as shown in Figure 8. Footage can be viewed directly in MotionEyeOS from the web browser or downloaded to a Windows or Mac for later viewing.

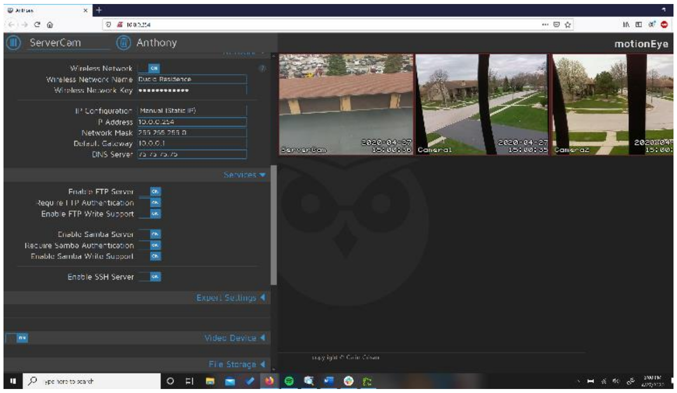


Storage Device: Custom Path
Root Directory: /data/output/Camera1
Disk Usage: 0.1/57.8 GB (0%)
Upload Media Files: OFF
Call A Web Hook: OFF
Run A Command: OFF

FIGURE 8: A SCREENSHOT OF THE MOTIONEYEOS SERVICES TAB, SHOWING SSH ENABLED.

USB flash drive storage is generally more convenient in a Microsoft or Mac setting. In the context of MotionEyeOS, storing footage locally to a USB flash drive for later viewing is a process that requires a few extra minutes to format a USB drive to be used with Linux. This procedure begins with downloading a software called PuTTY.

For PuTTY to work properly, MotionEyeOS must be configured under the "Services" tab to allow the use of a secure shell server, or SSH, shown in Figure 9. PuTTY allows the administrator to remotely access MotionEye's Linux command line and format the USB drive to work properly in this context. The Linux formatting commands are readily available with a trivial internet search.



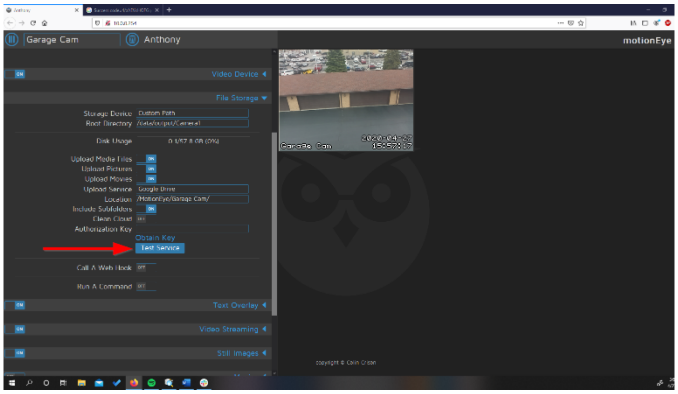
Wireless Network: ON
Wireless Network Name: Ducic Residence
Wireless Network Key: ●●●●●●●●●●●●
IP Configuration: Manual (Static IP)
IP Address: 10.0.0.254
Network Mask: 255.255.255.0
Default Gateway: 10.0.0.1
DNS Server: 75.75.75.75
Enable FTP Server: ON
Require FTP Authentication: ON
Enable FTP Write Support: ON
Enable Samba Server: ON
Require Samba Authentication: ON
Enable Samba Write Support: ON
Enable SSH Server: ON

FIGURE 9: A SCREENSHOT OF THE MOTIONEYEOS SERVICES TAB, SHOWING SSH ENABLED.

Once the USB drive is formatted and inserted into the Pi 3B camera hub, the procedure to save data locally is identical to saving to the micro SD card.

3.6 Storing Video and Images to a UIC Google Drive

MotionEyeOS offers compatibility with several cloud-based services, including Google Drive, Google Photo, and Dropbox. The University of Illinois at Chicago uses Gmail as an email service, which comes with a Google Drive account. Each camera must be individually configured for cloud storage under the dashboard's "File Storage" tab. Options entitled "Upload Media File" and "Upload Service" must be selected. This procedure is uploading to Google Drive. The location box generates a new pathway in the administrator's Google Drive account, in this case "*/MotionEye/Garage Cam/*". Selecting the "Test Service" button as shown in Figure 10 prompts the user to enter Google Drive credentials, copy and paste a Google authorization key into the File Storage tab, and will finally inform the administrator of a successful link between MotionEyeOS and the cloud storage service.

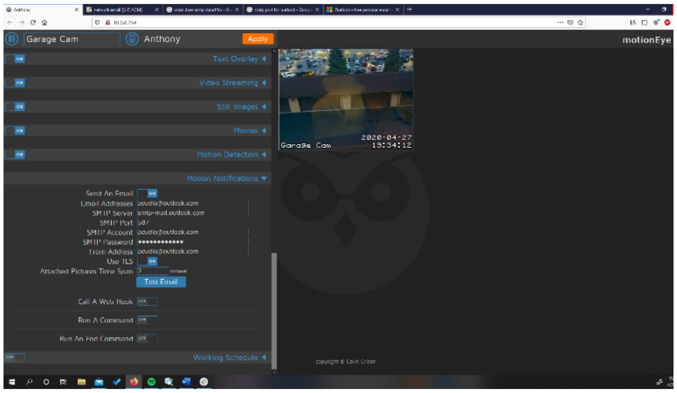


Storage Device: Custom Path
Root Directory: /data/output/Camera1
Disk Usage: 0.1/57.8 GB (0%)
Upload Media Files: ON
Upload Pictures: ON
Upload Movies: ON
Upload Service: Google Drive
Location: /MotionEye/Garage Cam/
Include Subfolders: ON
Clean Cloud: OFF
Authorization Key:
Call A Web Hook: OFF
Run A Command: OFF

FIGURE 10: A SCREENSHOT OF THE MOTIONEYEOS FILE STORAGE MENU ILLUSTRATING THE CLOUD STORAGE SETUP PROCEDURE.

3.7 Enabling motion notifications

MotionEyeOS has several different options for motion notifications - this procedure focuses exclusively on email notifications. The "Motion Notifications" option requires a user's email credentials and information on the email service provider. A trivial internet search is necessary to find port and server information for the administrator's email provider. Figure 11 illustrates the "Motion Notification" tab on the dashboard.



Send An Email: ON
Email Addresses: oudio@outlook.com
SMTP Server: smtp-mail.outlook.com
SMTP Port: 587
SMTP Account: oudio@outlook.com
SMTP Password: ••••••••••••
From Address: oudio@outlook.com
Use TLS: ON
Attached Pictures Time Span: 2 seconds
Call A Web Hook: OFF
Run A Command: OFF
Run An End Command: OFF

FIGURE 11: A SCREENSHOT OF THE MOTIONEYEOS MOTION NOTIFICATION MENU.

4. RESULTS AND DISCUSSION

MotionEyeOS has built-in features that allow the user to customize the security system's duty cycle, image orientation, motion sensitivity, storage medium, and risk notification method. Security footage can be obtained in video or still image format, and the administrator can enable automatic or manual image surveillance.

If data is being stored to a camera's local SD card or USB flash drive, adhering to a regular data-backup schedule is critical to maintain service continuity. High quality images and long videos take up significant disk space, and data may be lost if not archived on a regular basis. If the administrator requires video to be saved locally, manipulation of motion sensitivity, image and video quality, and maximum video length must occur to optimize camera output with respect to the user's archive cycle.

Implementing a MotionEyeOS camera system that saves to the cloud requires extensive initial administrator attention to achieve proper results. During initial testing, high resolution images and videos obtained with high motion sensitivity settings took several days to upload to Google Drive and took over 24 hours to send email notifications. This latency may be a function of internet speed but is more likely due to the inexpensive nature of the camera system. Upgrading each computer to the latest model Raspberry Pi would add graphics processing units, RAM, and faster wireless adapters with the added benefit of ethernet connectivity. Although the project's cost would double, the system utility may increase proportionally.

5. CONCLUSION

When seeking an inexpensive and easy-to-implement security system for a university laboratory setting, the MotionEyeOS operating system in concert with a network of wireless Raspberry Pi single board computers is difficult to compete with. MotionEyeOS software works on a user-specified duty schedule, offers continuous recording or motion detection to save disk space, and features multi camera functionality. Multiple modes of storing footage and images locally or cloud-based in combination with stream over IP functionality makes a useful home security network with a minimal learning curve for implementation. A fully functional security network including all the hardware to support two Raspberry Pi Zero W camera setups and one Raspberry Pi 3B camera server combination costs about $145.

6. FUTURE WORK

This procedure surveys a fraction of MotionEyeOS' functionality. Options like calling a web hook, running commands, and saving data over a shared network were not explored. These options and several others may increase the system's utility and user experience.

This procedure also used some of the most inexpensive Pi cameras on the market, leading to grainy and low-quality images. Using slightly more expensive hardware capable of working in low-light situations may increase system's utility and user experience.

This procedure did not incorporate any kind of protective structure to house the cameras and shield them from the elements. Designing a weatherproof housing in CAD is trivial, and highly recommended if these units are to see actual service.

ACKNOWLEDGEMENTS

I would like to thank Dr. Pranav Bhounsule for his expert advice and encouragement throughout this project, along with Dr. Yeow Siow for introducing me to Dr. Bhounsule.

REFERENCES

[1] Crisan, C., 2020, URL *https://github.com/ccrisan*.

[2] SD Card Formatter for Windows, 2020, URL *https://www.sdcard.org/downloads/formatter_4/.*

[3] Blaena Etcher Corp. 2020, URL *https://www.balena.io/etcher/.*

[4] DB Tech, 2019, URL *https://www.youtube.com/watch?v=O5ifTks4w4U.*

[5] Famatech Corp. 2020, URL *https://www.advanced-port-scanner.com/.*

[6] PuTTY. 2019, URL *https://www.chiark.greenend.org.uk/~sgtatham/putty/*