Detection and Deflection of Digital Cameras:

An Exploration into Protecting Personal Privacy in the Modern World

By

Jessica Sanford

* * * * * * * * *

Submitted in partial fulfillment
of the requirements for
Honors in the Department of Computer Science

UNION COLLEGE
June, 2016

ABSTRACT

SANFORD, JESSICA Detection and Deflection of Digital Cameras: An Exploration into Protecting Personal Privacy in the Modern World. Department of Computer Science, June 2016.

ADVISOR: John Rieffel

As all forms of technology become more integrated into our daily lives, personal privacy has become a major concern. Everyday devices, such as mobile phones, have surveillance capabilities simply by having a digital camera as part of the device. And while privacy and secrecy seem to go hand in hand, it is not always the case that one does not care about privacy because they have nothing to hide. For example, everything from unflattering photographs to being unknowingly and perhaps criminally surveilled, are ample reasons to desire some means of combatting the not so candid presence of digital cameras in everyday life. There is also the more casual argument of having control over one's public image. For these reasons, we propose a wearable device that offers personal privacy protection. This device should be able to detect for cameras in an area, and then disrupt the photographer's ability to capture their photo. In this project, we explored the implications of creating such a device, and evaluate which approaches to detection and disruption would be possible for such a device.

Contents

List of Figures

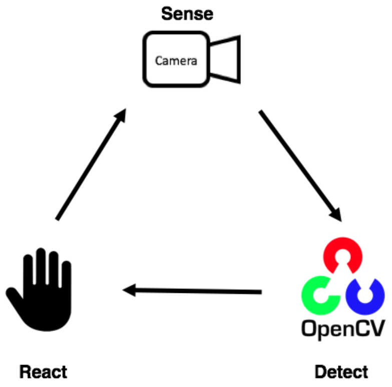


Figure 1: Design foundation of this project. We will sense the area around the user, detect for cameras, and react to the results of detection.

1 Introduction

The purpose of this project is to develop a wearable device that can assist its wearer in gaining control over their personal privacy. We look to develop a device that can sense the users environment, detect if there is anyone trying to photograph the user in this environment, and if such a situation is detected, prevent it from happening. However, this goal also presents and highlights a challenge to the industry of wearable technology in general; the device itself must be casually wearable, and also fashionably and socially acceptable. Therefore while designing our device, we looked to keep the device as invisible as possible, not drawing attention to the user, but rather maintaining their invisibility.

To tackle this problem, we divided it into three separate stages, as seen in Figure 1. Firstly, the device must have a way to sense the area around the user. Then, using this raw sensor data, the device must detect if any cameras have been sensed. After that, the device must react to the output of the detection algorithm, disrupting the photographer's ability to capture the photo of the user.

Fortunately, the first stage is easy. Deciding to use a Raspberry Pi to run our detection algorithm, this step only required that we equip the Pi with a camera. Moving on to detection, we had to investigate an attribute of digital cameras that we could exploit in order to make them easily visible to our device. Fortunately, the two most common types of image sensors, CMOS and CCD, have the quality of being

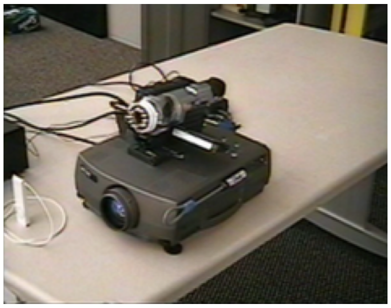


Figure 2: Truong et. al.'s device. Sitting on top of the projector is the Sony HandyCam used for sensing, and the projector is used for blinding cameras in the area of the device. [10]

retro-reflective. Retro-reflectivity refers to the quality of reflecting light directly back at its source. Thus, we have connected an infrared night vision camera to the Raspberry Pi 2 Model B, flanked by two strong infrared LEDs. The intention of these IR LEDs is to produce this retro-reflection from an image sensor which we need in order to isolate cameras in the field of view of our night vision camera. We then process the image feed from the night vision camera using OpenCV run on the Raspberry Pi.

If cameras are then found in the area, we move on to the third stage of the device, which is to react to our detection results, and disrupt any images being taken by the cameras sensed by the our night vision camera. We have explored several methods of image disruption, but at the moment, it appears that radio frequency transmission is the most invisible way of disrupting digital photography. We are still in the process of implementing this, and have therefore implemented user notification as a substitute reaction method. User notification still gives the user the power to control their public image, by making them aware that they are being recorded or photographed. Notification is performed using a piezo buzzer. When a camera is detected, we simply play a very low frequency tone using a piezo buzzer.

2 Research Questions

The research questions for this project are as follows: what design flaws can we exploit in imaging technology to render it useless? For example, can we exploit the reflective properties of CMOS and CCD image sensors to then computationally find cameras in a given area? Can the processing needed to sense these

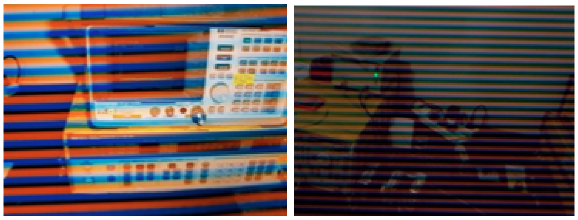


Figure 3: Image disruption generated by RF transmisson on camera using CCD Sensor. [7]

cameras be performed in real time fast enough to be able to then disrupt the cameras in the area? Can this solution be wearable? The main computational focus of this project resides in image processing, and maximizing computational ability at the wearable level.

As mentioned before, we plan to exploit the retro-reflectivity inherent to image sensors in order to detect for digital cameras in an area. After detection, we move to disrupting a camera, if detected, by interfering with its natural interaction with the electromagnetic field, by transmitting radio waves at the frequency of its readout rate.

Camera detection takes several steps of image transformation, which posed an issue for such a limited GPU. In our research we looked for ways to maximize the abilities of this processor to give our image processing more resources to use.

3 Background and Related Work

Similar projects, attempting to foil digital photography, have been developed in both the consumer and the academic realm. Our sensing and detection methods were inspired by Truong et. al.'s approach, in which they use the retro-reflective quality of image sensors, and a night vision camera, to detect cameras in a given area. Their device uses a Sony Handycam in NightVision mode with infrared LEDS rimming the aperture, to sense for cameras an area, and a large project to then blind any cameras that have been detected [10], as shown in Figure 2. The device is meant to be mounted on the wall, in the corner of a room to cover maximum viewing angles. This research informs ours greatly, as our project plan involves taking their sensing concept to the wearable level.

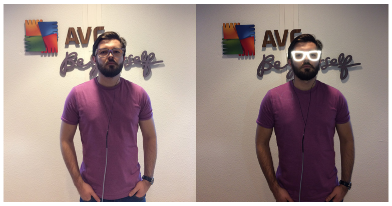


Figure 4: Glasses that foil facial Recognition. [8]

More specifically informing the third stage of our project, is research done by Ariel Schwarz, Zeev Zalevsky, and Yosef Sanhedrai on controlled radio frequency trasmission for image disruption [7]. Their team found that CMOS and CCD sensors each have a readout rate that produces a measurable electromagnetic wave, which they then recieved and measured using a RF receiver circuit and a RF spectrum analyzer. They found that images can be disrupted by transmitting a radio wave in the same frequency as that emitted from the image sensor. The results of their project were very promising, as shown in Figure 3, and significantly disrupted the images being captured by their test cameras. However this project used RF receivers and RF spectrum analyzer for camera detection, making the sensing portion of the project quite bulky and expensive. The disruption portion of the project must be much more concealable, and we therefore are also developing a way to take this project to the wearable level.

Other projects have explored similar themes of protecting personal privacy, such as foiling facial recognition or blinding cameras with their own flash. Researchers at Innovation Labs by AVG Technologies [8] created glasses with embedded infrared LEDs, exploiting the fact that most digital cameras can see this light while humans cannot. This light affects the ability of facial recognition algorithms to correctly locate facial features. There are also projects that use light in a different way, such as the line of clothing called Flashback, created by DJ Chris Holmes, which uses reflective threading to reflect the light from a camera flash and blind the camera with this burst of reflective light. This project was created to thwart paparazzi ambushes, however it only works with flash photography.

Overall, our research is very inspired by that of Truong et. al. and Schwarz et. al., and the design for our device is a hybrid, wearable version of the two approaches. It looks to explore common issues in the wearable technology industry, and its purpose is to solve one of the largest issues in the new technological age of personal computing, the loss of personal privacy.

4 First Approaches at Disruption

Prior to deciding on radio frequency transmission for the disruption method, we researched several other possibilites for creating disruption. Naturally, we first looked to reverse engineering autofocus algorithms to see if we could find a computational or physical flaw naturally occuring in how common digital cameras focus. Digital cameras employ two types of autofocus: active or passive. Active autofocus employs external sensors and measurement tools to measure the distance between objects and the camera lense, such as infrared lasers or sonic waves. Passive autofocus relies solely on the light that comes in through the lense and hits the image sensor to focus on objects in a scene [12]. Most digital cameras employ passive autofocus, because external measurement tools like infrared lasers have difficulty focusing on transparent surfaces, like glass [12]. However some cameras, such as the one on the LG G4 phone, still use active autofocus with infrared lasers. Infrared laser focus makes our job much easier, as it produces a strong infrared light source for our detection method to pick up. Unfortunately, most digital cameras, like that of the iPhone, do not use active autofocus. Therefore, when looking to reverse engineer autofocus, we needed to take a deeper look at what types of passive autofocus exist.

There are two types of passive autofocus algorithms that digital cameras use: contrast and phase detection. Contrast autofocus attempts to focus a photo by maxmizing the intensity difference between adjacent pixels on the image sensor [9]. Phase detection autofocus splits incoming light up into two photos, using two focus pixels [9]. It then compares these two images until they line up. Both of these approaches obviously rely heavily on the amount of light coming into the camera in order to focus. Therefore, they are most easily disrupted by too much or too little light.

We experimented with different approaches at disruption meant to exploit weaknesses in these algorithms. Since phase detection autofocus is increasingly becoming the most common autofocus method in consumer level cameras, we looked at ways of disrupting the image alignment required to achieve this type of focus. Since we did not want our disruption method to draw attention to the user, we avoided flooding the camera lense with light, as Truong et. al. did. Instead, we tried creating motion with infrared light. Using an Arduino Nano to program infrared LEDs, we tried several different blinking patterns, at varying speeds to see if we could create a pattern difficult for a camera to focus on. We hypothesized that we could

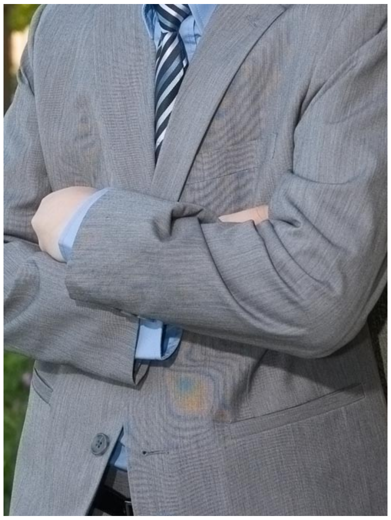


Figure 5: Moiré pattern on a man's striped suit. The curved yellow and blue lines are the superimposed pattern. [5]

create a blinking pattern that would make it difficult for the phase detection autofocus to align the two images it uses together. We tested this method with several different types of cameras, including that of the iPhone 6, the Samsung Galaxy 6, and an older handheld digital camera, the HP Photosmart E317. While the HP camera had trouble focusing, the other two cameras did not. Since our project focus was to disrupt digital photography for the most common digital cameras used today, we decided the small amount of disruption we received from the HP was not promising enough to continue using this as our method of disruption.

We also investigated natural situations that prove difficult for digital cameras to capture. Most obviously, low lighted and brightly lighted conditions are very difficult to capture clear photographs in, but we found that these were difficult conditions to control with a wearable device. Low light conditions are almost impossible to produce without direct access to external factors producing light in the room, and bright light conditions draw too much attention to the user for the goals of this project. Even bright light outside the visible spectrum is difficult to bring to the wearable level, as it requires substantial amounts of power. Nevertheless, we believed there was some promise in a naturally occuring pattern called the moiré pattern. The moiré pattern is the resulting superimposed pattern produced from inprecisely overlaying

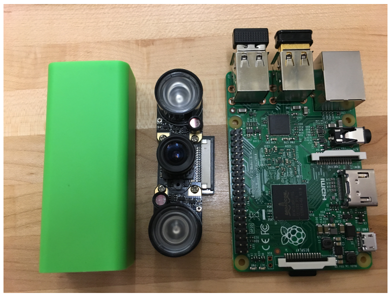


Figure 6: Sensing and detection unit. Left is a portable power supply. Middle is the night vision camera used for sensing other cameras. Right is the Raspberry Pi Model 2 B used for detection processing.

two identical grid like patterns [1]. The moiré pattern appears in digital photography as the result of the pixel grid of the image sensor trying to reassemble an already grid like structure. The result is the sumperimposed pattern like the one depicted in seen in Figure 5. While this pattern represents a clear disuption that is the natural result of the inability of digitization to handle grid-like surfaces, we found that it was not disruptive enough for our purposes, and decided to focus our efforts on radio frequency transmission as the method our device. We reserve the moiré pattern for future research, where we would look for ways to amplify its effect.

Since we wanted our approach to disruption to be invisible to the public, we decided that the approach that Schwarz et. al. took best suited our needs for this device. In any case, future work for this device will involve much more research into disruption techniques.

5 Sensing and Detection

In order to identify reflections from CMOS and CCD sensors, we needed to be able to uniquely identify the light source we were producing to create the reflections. Therefore, for the sensing part of this project, a night vision camera made the most sense. Unlike other cameras, a night vision camera can see infrared light. In addition to being able to operate in low light conditions, the infrared LEDs illuminate outside of the visible spectrum of light, which makes the device less visible to the public. For our camera, we chose the SainSmart Infrared Night Vision Surveillance Camera for the Raspberry Pi. The camera comes with two 1W infrared leds, as pictured in Figure 6.

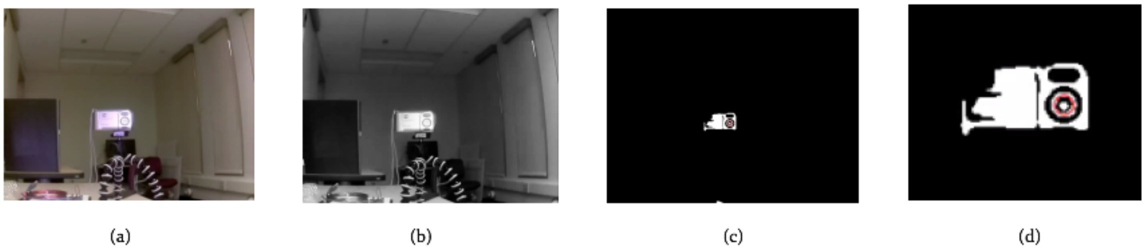


Figure 7: (a) Raw image. (b) Grayscale image. (c) Thresholded image. (d) Close up of camera being detected. Red circle indicates a camera sensor was found.

Essentially, looking for reflections becomes the simple task of looking for the brightest clusters, or blobs [4], of pixels in the video feed coming through our night vision camera. In order to detect these blobs, the video feed is passed into an OpenCV python script to process the camera detection. First, we convert the image to grayscale. This essentially makes the the photo black and white, as as seen in Figure 7b. This turns each pixel in the image into shade of gray, between the values of 0, which is black, and 255, which is white. From the grayscale image, we then threshold out pixels below our expected brightness range, which through experimentation we found to be most successful between 240 and 255. We also did additional filtering for the size of the blobs and their shape, to filter out any other lights or reflective surfaces that may also appear in the threshold. When an image sensor is detected, a red circle is drawn around the sensor, as seen in Figure 7d.

We recognized that attempting to perform real time image processing on the Raspberry Pi's 900MHz quad-core ARM Cortex-A7 CPU would be difficult, therefore we researched several ways to maximize the processors ability. When noticing a bit of lag while watching the video output, we tried to expand the memory dedicated to the GPU. On the Raspberry Pi, the RAM dedicated to the GPU is shared with that dedicated to the ARM CPU. Therefore, the more memory given to the GPU, the less RAM the processor has. Originally, there are 16MB dedicated to the GPU. We increased this to 320MB. We saw some enhanced improvement when increasing the GPU memory, but not enough. We then tried to see how far we could push the processor, by overclocking it. Although the Raspberry Pi has a 900 MGHz processor, its default clock cycle is set to 700MHz. When pushing the processor to 800MHz, we saw minimal improvement, but when pushing it to the full 900MHz, we began to have boot problems, could not run our program at all.

Overall, we think that in the future, perhaps looking at a more powerful processor would give us the speed reliability we need to ensure our device reacts in enough time to disrupt the photo being taken.

5.1 Issues with this Method of Detection

When replicating the research done by Truong et. al. we began to realize that this method of detection was not as reliable as we initially perceived it to be. In their research, Truong et. al. makes the claim that they are able to see the CCD or CMOS image sensor's reflection, with plenty of ambient light, by illuminating an area with their infrared LEDs. Looking more closely at the optics of a consumer level camera, one realizes that being able to see the reflection of light directly from the image sensor is impossible. Firstly, the image sensor of a camera is placed behind many layers of optics designed to capture light rays and focus them on the sensor, where they will then be digitized. These optics alone make it impossible to claim that the image sensor is uniquely producing the reflection.

Secondly, every consumer level camera, unless its intended use is to see infrared light, has an infrared cut filter that resides right in front of the image sensor, blocking direct access to the sensor for reflective purposes. The cut filters ensure image quality. Since a CMOS or CCD sensor employs photosensitive elements to capture light and then translate it into an electronic signal, these sensors are vulnerable to a much wider range of light than the human eye [2]. Therefore, a filter to cut out infrared light is almost always placed in front of the image sensor on digital cameras, to prevent the infrared light from being pixelated, shaded, and included in the image, as this would cause the image to look much different than the human eye would see it [2].

There are two types of infrared cut filters, absorptive and reflective. Absorptive filters are made of a special type of optical glass that absorbs near infrared raditation [6]. Reflective filters bounce infrared light back out of the camera lense. Absorptive filters would completely foil this attempt at detection, because no light is being reflected at all. Even reflective filters are still preventing infrared light from reaching the image sensors, so therefore one cannot truly claim that the reflections produced by the infrared LEDs in this detection set up are coming directly from the image sensor.

In addition to these two facts about camera optics, we found that this method of detection, as Truong

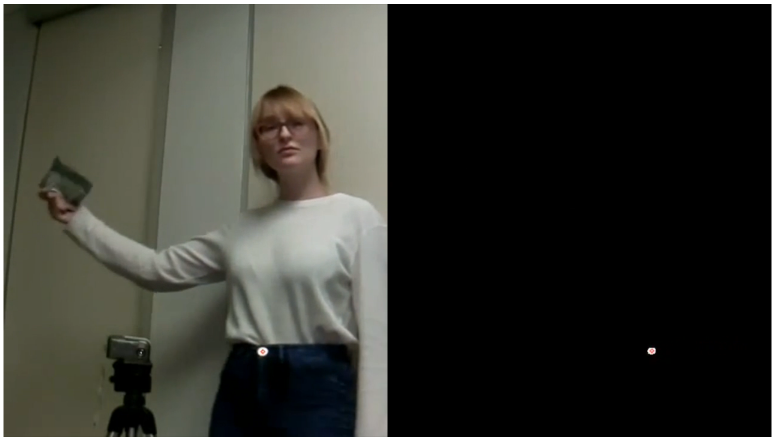


Figure 8: The same camera as pictured in 7, but it is not picked up using the same detection parameters.

et. al. concede, suffers from conditional inconsistencies that make it difficult to design detect parameters. For example, changes in distance between the cameras in an area and camera sensor unit cause a camera that was once picked by our detection unit a closer distance to then be missed by the detection unit at a further distance, as shown in Figure 8. Even when changing conditional circumstances, such as increasing the amount of infrared light used illuminate the area, we found that the camera could not be picked up at this distance. Additionally, we tried changing thresholding parameters and blob detection parameters, but we could not get the sensor to be picked up by our detection script. Regardless, we have concluded that achieving consistent detection under different conditions can not be accomplished.

We also tested this with several different types of cameras, one of them being that of the iPhone 6. As you can see in Figure **??**, the iPhone is not detected at all. After doing some research, we found that the iPhone uses a Hybrid IR Filter, which is combinational infrared cut filter, employ both the absorptive and reflective technique. We suspect that the absorptive qualities of the filter contribute to the inability of our detection unit to pick up any infrared reflections from the camera.

This method of detection is not a reliable method for detecting cameras in an area. In addition to the inconsitencies and flaws mentioned above, this method of detection also picks up many false positives. For example, in Figure 8, the button of the author's pants is picked up as an image sensor reflection, when it is clearly not an image sensor. False positives seriously impact the validity of our system, especially because of our two reaction methods. The first method, transmitting radio frequency to disrupt camera functionality, poses the issue of unnecesarrily transmitting RF in the area of the user. While once fully realized, the device will not pose a threat to the user, it is still much safer to limit any radio frequency exposure for long periods of time. In the worst case scenario of false positives, we could potentially be detecting an "image sensor" that is really just a reflection from a static object, like a drinking glass positioned near the user, and therefore this detection would last for a long period of time. The other method of disruption, user notification, would simply cause unneccessary paranoia for the user in the case of false positives. Therefore, this method of detection still needs development, and in the future we would look for other ways to detect cameras in an area.

6 Disruption

The disruption phase of this project refers to the reaction part of the loop in the original design shown in Figure 1. We outline our two methods of disruption, one implemented, user notification, and one still in development, the radio frequency transmission.

6.1 User Notification

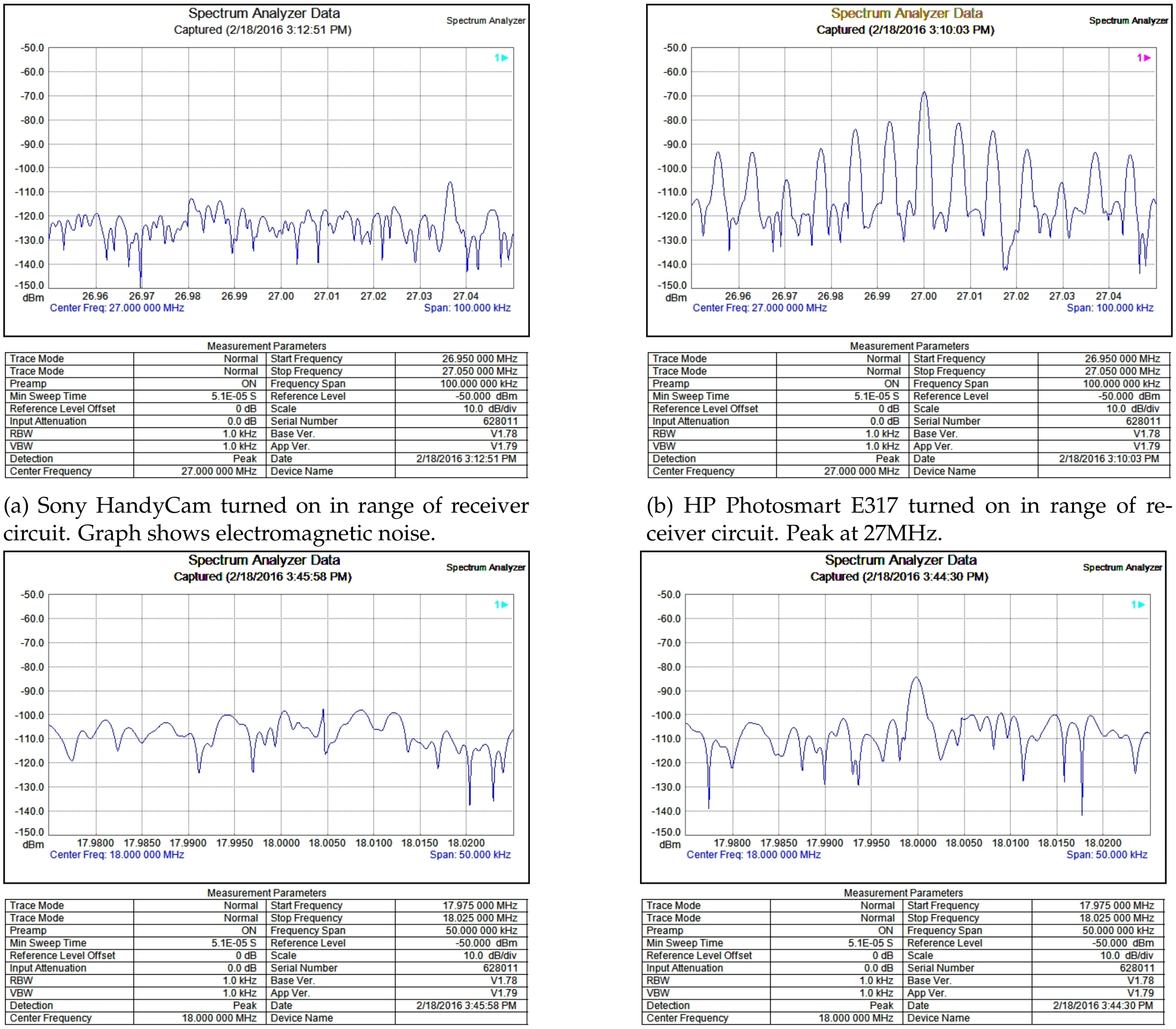
We understand user notification of cameras in an area as a form of disruption because it makes the user aware that a camera is pointed at them, and with this information, they can choose to avoid their photographer. We implemented user notification using a piezo buzzer that plugged directly in to the GPIO pins of the Raspberry Pi. A piezo buzzer is a small audio signaling device. We use the device to react to a detected camera. If a camera is detected, the piezo buzzer plays a low frequency tone, notifying that the device has seen a camera. Otherwise, it remains silent. While the tone makes the device more noticeable to the public, it is a standard electronic beeping, so it easily assimilates into the daily noise of ringtone and text message notification alerts.

6.2 Radio Frequency Transmission

The radio frequency transmission method of disruption, designed after the research done by Schwarz et. al. [7], focuses on exploiting another inherent quality of digital cameras that we can use to disrupt their function: their readout rate. A digital camera's readout rate is the time required to digitize a pixel of the image sensor, and is often expressed in terms of frequency [11]. This readout rate can be detected using a spectrum analyzer and radio receiver circuit. Schwarz et. al. found that transmitting another signal at this same frequency causes visible disruption in the cameras field of view. In our research, we first attempted to replicate their results for detection to determine what frequency range would be best for our transmission purposes.

While our main goal is to disrupt cameras that are most common to current day, like those embedded in mobile phones, we tested several different cameras to first see if we could replicate the results that Schwarz et. al. reported, and also to test to see if modern cameras, such as those in mobile phones, still conform to the frequencies reported in their research. For our experiments, we used an Anritsu MS2721A spectrum analyzer and small loop antenna as the receiver circuit. We tested three cameras: the HP Photosmart E317, the Sony HandyCam, and the iPhone 6s iSight camera. For the first two cameras, we were able to produce results very similar to those of Schwarz et. al. As seen in figure 9b, the spectrum analyzer detected a peak at 27MHz. The peak detected represents the camera's readout rate, because it is only detected when the camera is turned on. This replicates the results of Schwarz et. al., in which they found digital camera readout rates to be between 12MHz and 24MHz. The readout rate of the HP Photosmart E317 is very close to that of the cell phone camera Samsung 944 CMOS sensor used by Schwarz et. al., whose readout rate showed at peak at 24Mhz [7]. The Sony HandyCam showed very similar results, seen in Figure 9d, to the Kodak CX7330 CCD sensor frequencies, producing a main peak at 18MHz, and a secondary peak, not pictured below, at 5Khz (differing slightly from the Kodaks secondary peak at 6.8KHz) [7].

Unfortunately, we could not pick up any peaks for the iPhone 6s, when turning on and off the camera. Even when turning on and off the entire device, and toggling on and off cell phone service, there were no significant peaks detected. We believe this is due to there being too much electromagnetic noise generated by the iPhone's processor when the device is turned on. The level of noise changed when the iPhone was



Measurement Parameters

Trace Mode	Normal	Start Frequency	26.950 000 MHz
Trace Mode	Normal	Stop Frequency	27.050 000 MHz
Preamp	ON	Frequency Span	100.000 000 kHz
Min Sweep Time	5.1E-05 S	Reference Level	-50.000 dBm
Reference Level Offset	0 dB	Scale	10.0 dB/div
Input Attenuation	0.0 dB	Serial Number	628011
RBW	1.0 kHz	Base Ver.	V1.78
VBW	1.0 kHz	App Ver.	V1.79
Detection	Peak	Date	2/18/2016 3:12:51 PM
Center Frequency	27.000 000 MHz	Device Name	

(a) Sony HandyCam turned on in range of receiver circuit. Graph shows electromagnetic noise.

Measurement Parameters

Trace Mode	Normal	Start Frequency	26.950 000 MHz
Trace Mode	Normal	Stop Frequency	27.050 000 MHz
Preamp	ON	Frequency Span	100.000 000 kHz
Min Sweep Time	5.1E-05 S	Reference Level	-50.000 dBm
Reference Level Offset	0 dB	Scale	10.0 dB/div
Input Attenuation	0.0 dB	Serial Number	628011
RBW	1.0 kHz	Base Ver.	V1.78
VBW	1.0 kHz	App Ver.	V1.79
Detection	Peak	Date	2/18/2016 3:10:03 PM
Center Frequency	27.000 000 MHz	Device Name	

(b) HP Photosmart E317 turned on in range of receiver circuit. Peak at 27MHz.

Measurement Parameters

Trace Mode	Normal	Start Frequency	17.975 000 MHz
Trace Mode	Normal	Stop Frequency	18.025 000 MHz
Preamp	ON	Frequency Span	50.000 000 kHz
Min Sweep Time	5.1E-05 S	Reference Level	-50.000 dBm
Reference Level Offset	0 dB	Scale	10.0 dB/div
Input Attenuation	0.0 dB	Serial Number	628011
RBW	1.0 kHz	Base Ver.	V1.78
VBW	1.0 kHz	App Ver.	V1.79
Detection	Peak	Date	2/18/2016 3:45:58 PM
Center Frequency	18.000 000 MHz	Device Name	

(c) Sony HandyCam turned off in range of receiver circuit. Graph shows electromagnetic noise.

Measurement Parameters

Trace Mode	Normal	Start Frequency	17.975 000 MHz
Trace Mode	Normal	Stop Frequency	18.025 000 MHz
Preamp	ON	Frequency Span	50.000 000 kHz
Min Sweep Time	5.1E-05 S	Reference Level	-50.000 dBm
Reference Level Offset	0 dB	Scale	10.0 dB/div
Input Attenuation	0.0 dB	Serial Number	628011
RBW	1.0 kHz	Base Ver.	V1.78
VBW	1.0 kHz	App Ver.	V1.79
Detection	Peak	Date	2/18/2016 3:44:30 PM
Center Frequency	18.000 000 MHz	Device Name	

(d) Sony HandyCam turned on in range of receiver circuit. Peak at 18Mhz.

Figure 9: Graphings of electromagnetic frequencies picked up by spectrum analyzer. Peaks demonstrate digital camera readout rates.

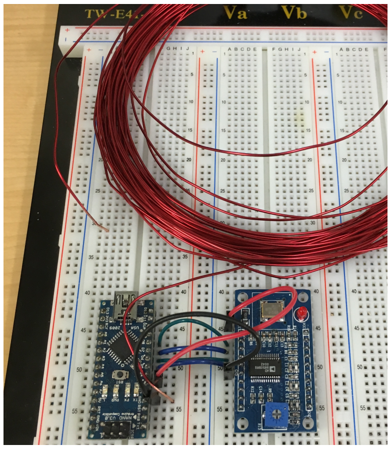


Figure 10: Leftmost is the Arduino Nano. Rightmost is the NooElec signal generator. The red coil is a crude antenna used for testing purposes.

placed in range of the receiver circuit, but again, no significant peaks were detected. Therefore, for mobile phones, we believe that this method of disruption may be a trial and error stage, and in future development we will attempt experiment with information that we could collect from less noisy camera units.

Once we were able to replicate the results from Schwarz et. al., we moved on to creating a unit that could transmit these frequencies. Firstly, we needed a signal generator that could produce frequencies at least up to 27MHz, and was also small enough to integrate into wearable form. Therefore we chose the NooElec AD9850 DDS Signal/Function Generator Module, pictured in Figure 10, which can transmit frequencies up to 40MHz and is 2.7 x 2.2 x 1.1 inches in dimensions, which is just a bit wider than Arduino Nano. For this portion of the unit, we successfully used an Arduino Nano to program the NooElec AD9850 to generate sine waves at any frequency up to 40MHz. We later plan to use serial communication from the Raspberry Pi to the Arduino Nano to replace the piezo buzzer notification unit with the NooElec signal generator. Using the Arduino Nano allows us to offload some of the processing power from the Raspberry Pi needed to generate the signal, while still remaining small in size. Next, we will look into designing a transmitter and antenna unit small enough to be part of the device while still being powerful enough to transmit the signal.

6.2.1 Antenna Issues

The only issue with fully realizing this radio freqency disruption portion of the project is the size of the antenna. An antenna's size is inversely proportional to the frequency it is trying to transmit. This means that the smaller the frequency, the larger the antenna. A half wave length dipole antenna, the antenna used by Schwarz et. al., has a length defined by the following equation [3]:

$$l = 39/f$$

where l is the length of the antenna in meters, and f is the frequency in megahertz to be transmitted.

Therefore an antenna that is capable of transmitting 27MHz, would need to be 1.4 meters. This antenna is quite long to be incorporated into this wearable device, therefore in the future we look to experiment more with antenna design. Since we still haven't developed an antenna, we have not yet created any disruption with this method.

7 Evaluation and Results

Overall, this project is still in the development stages. Our method of detection had inherent design flaws which became apparent when attempting to replicate the research of Truong et. al. While we did see some success in detecting cameras with this method, particularly with the 2005 HP Photosmart E317, this success was not consistent across all devices and across all environment conditions, such as distance from the device, and light levels in the environment. We also discovered that the original claims made by Truong et. al., that direct reflections from CCD and CMOS sensors can be seen and detected with this method, were not precisely correct. There are too many layers of optics placed in front of the sensor to ensure that any reflection is coming from the sensor and not from the lenses in front of it. Finally, this method of detection produces too many false positives to be considered reliable, especially for our system design.

We were able to sucessfully implement the user notification reaction method, but we have not yet reached the stage where we can perform user testing. The buzzer sounds when the unit believes a camera has been detected, but as we mentioned above, our detection results are not as reliable as desired. Our other reaction method, radio frequency transmission, still requires further development to take it to the wearable level. As a result, we're not yet able to measure the success of this method as a means of disruption of digital photography. In addition, if this device were to become a truly wearable product, we would need to get FCC certification to fully implement testing.

8 Conclusions

In conclusion, our device, although still in development, looks to protect the privacy of its user, and to also be as inconspicuous as possible. Through our research we were able to learn what does not work for camera detection, and now know we will have to redesign this portion of the project. Our reaction methods show promise for creating true disruption of digital photography, and we will continue the development of these methods. Overall, through the development of this device, we hope to offer freedom from an issue that ranges from an everyday annoyance to a personal threat: the protection of personal privacy. Our research is novel in the fact that it is attempting take a project like this to a wearable scale, and therefore is the first step towards a personal device that can achieve this goal. Developing such a project for the consumer level is still reserved for future work, as the steps taken here were necessary research in the development of such a device. In conclusion, we believe the research done here greatly informs the progress of developing such a device, and look forward to new investigations in the area.

References

[1] Wolfram Alpha. Moiré pattern, 2016.

[2] Y.S. Chang and T.W. Liu. Infrared cut filter, August 23 2012. US Patent App. 13/101,557.

[3] James W. "Russ" Healy. Antenna here is a dipole. *QST*, pages 24–26, 6 1991.

[4] Satya Mallick. Blob detection using opencv (python, c++), 2015.

[5] Nadine Ohara. Really bad moiré pattern, 2010.

[6] Precision Micro Optics. Ir cut filters, 2016.

[7] A. Schwarz, Z. Zalevsky, and Y. Sanhedrai. Digital camera sensing and its image disruption with controlled radio-frequency reception/transmission. In *Microwaves, Communications, Antennas and Electronics Systems (COMCAS), 2011 IEEE International Conference on*, pages 1–6, Nov 2011.

[8] AVG Technologies. Invisibility glasses, 2015.

[9] Toshiba. Image sensor technology: Phase detection auto-focus (pdaf), 2015.

[10] Khai N. Truong, Shwetak N. Patel, Jay W. Summet, and Gregory D. Abowd. Preventing camera recording by designing a capture-resistant environment. In *Proceedings of the 7th International Conference on Ubiquitous Computing*, UbiComp'05, pages 73–86, Berlin, Heidelberg, 2005. Springer-Verlag.

[11] Thomas J. Wellers and Michael W. Davidson. Concepts in digital imaging technology: Digital camera readout and frame rates, 2015.

[12] Xin Xu, Yinglin Wang, Jinshan Tang, Xiaolong Zhang, and Xiaoming Liu. Robust automatic focus algorithm for low contrast images using a new contrast measure. *Sensors*, 11(9):8281, 2011.