ChromaCam: Demodulating Colored Light with a High-End Smartphone

Eric Kemmeren
Supervisors: Koen Langendoen, Miguel Chavez Tapia
EEMCS, Delft University of Technology, The Netherlands
20-6-2022

A Dissertation Submitted to EEMCS faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering

Abstract

Visible Light Communication (VLC) has seen drastic improvements in recent years, one approach uses active light sources like LEDs, switching them at hight speeds to send data. Another approach uses the fundamental characteristics of liquid crystals (birefringence and thickness) to transmit data to a single pixel receiver. These characteristics allow for alterations in the color of the light. Little power is being used to transmit this data in comparison with using LEDs, and is significant faster than switching the LCs fully on or off like other research has done. This paper proposes an algorithm to decode the data transmitted by LCs with a high-end smartphone instead of the single pixel receiver. This algorithm called ChromaCam has real time transmitter detection and decoding of data. A demonstration is shown using a prototype, achieving data rates of 31 bits per second.

1 Introduction

In 1880, Alexander Graham was the first person to transmit data wirelessly, using sunlight to transmit the human voice over 200 meters distance. He called this invention the Photophone [1]. After this, wireless communication technology improved a lot and is so common in this day and age that the microwave and radio wave frequency bands are getting crowded. When multiple transmitters and receivers are using the same bandwidth interference can occur where the data can not properly be transmitted. One part of the electromagnetic spectrum is not as commonly used for transmitting data, the visible light frequency band.

With Visible Light Communication (VLC) many applications have already been developed, for example, creating an indoor positioning system [8]. These systems have high energy consumption because of the cost of illumination [2]. These costs can become as large as a few Watts. A solution for this problem was proposed in LuxLink [3]. Ambient light is modulated using liquid crystals to send data. This is received by a photosensor that produces a electrical current depending on the incoming light. This made the transmitter use only around 30mW. The bit rate reached was only 80 bits per second in a range of 60 meters using sunlight. ChromaLux offers a new way to use liquid crystal displays to transmit data [5]. Using the birefringent property of liquid crystal cells it is possible to change the color of polarized light depending on the voltage applied. This allows for smaller adjustments in voltage for faster switching times between bits. The liquid crystal panels work by having a different refractive index for wavelengths of light. In a liquid crystal panel, we can change this refractive index by applying a voltage to it. This way the wavelength of the light can be modulated to allow for sending data encoded in the color of the light.

Just like in LuxLink, Ghiasi, et al. in Chromalux used a photosensor to detect the incoming light. In this paper, we propose another way to detect the incoming light, with a CMOS camera sensor. These cameras experience an effect called the rolling shutter effect [4]. Every subsequent line of pixels in a picture is taken right after each other, instead of at the same line. If an object in the picture is moving or changing fast enough, it results in distortion of the image. This distortion is used to send data to the phone. This type of camera can be found in modern-day phones. It will allow for applications that require no extra hardware on the receiver side. Using the transmitter from Chromalux, this paper will explore how to demodulate the optical signal that is transmitted with a high-end phone camera.

2 Related Work

In this section, other research in visible light communication will be discussed, in particular systems for the transmitter, receiver, and encoding of the data.

2.1 Transmitter

Many different types of transmitters have been developed in the last few years, many use an active led to transmit the data [7; 8; 9; 14; 15]. Others use passive VLC, no light is transmitted at the transmitter. Ambient light is altered to send data. This altering could be changing the intensity of the light, or changing the color. Applications that use visible light communication range from reading tags invisible to the human eye [13] to modulating the amplitude of the light to send data [3].

In this paper the transmitter developed by Ghiasi, et al. is used [5]. The authors discuss how the wavelengths can be selected for the on and off bits, to get the highest throughput. The symbols will have high contrast and a low switching time between these states. The authors also discuss how to choose the optimal number of liquid crystals in a stack. It was noticed that smaller transition periods were found with multiple but less thick cells with a total thickness of $n * w$ instead of a single cell with a thickness of $n * w$. These liquid crystal cells can change the direction of polarization of polarized light, if this light then passes through another polarizer the color will also update. ChromaLux placed this polarizer at the receiver. To remove the need for dedicated hardware on the receiver side, a modification is made here to the transmitter. The polarizer is placed at the transmitter. As humans can not see the change in polarization of light, the trade-off is that extra care needs to be taken to make sure the color modulation is not harmful to observers as flicker in the form of the transmitter changing color could be noticeable.

2.2 Receiver

Two types of receivers are commonly used in visible light communication. Photodiodes [2; 3; 5] are simple, cheap and have a fast update speed. They turn the incoming light into a current. This allows for fast communication between the transmitter and receiver. CMOS cameras [4; 8; 14; 15] are exploiting how pictures are taken. Instead of exposing the whole sensor at once, the image is captured line by line. This effect is called the rolling shutter effect. As every line of the picture is exposed at a different time, if the transmitter can transmit fast enough bands of colors can be seen in the picture, an example is shown in figure 1.

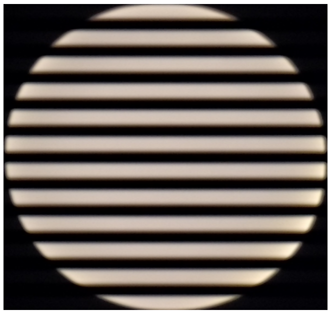


Figure 1: The rolling shutter effect with and LED that is blinking at high speed.

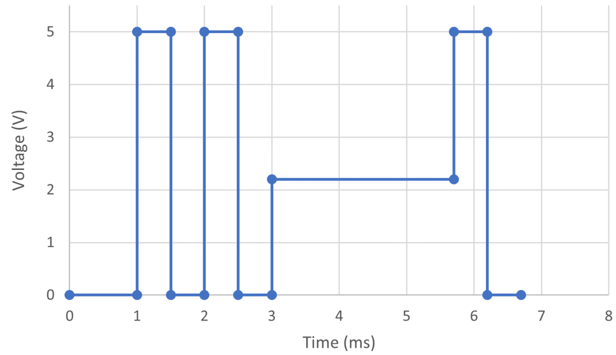


Figure 2: Encoding used with ChromaLux. This assumes 0 volt was already applied for at least 16 milliseconds. Five volt is then applied to get the transmitter to the required color. After this, every millisecond a bit is send. Sending a one corresponds with pulling the receiver high for 0.2 milliseconds, then applying 0 volt for 0.8 milliseconds. A zero corresponds a voltage of 2.2 volt for 1 millisecond.

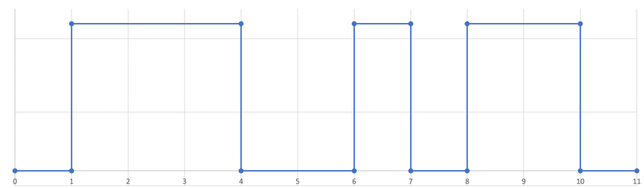


Figure 3: Example of differential Manchester encoding. First five symbols are the header, identified by three consecutive ones. From symbol six to seven is the opposite edge as from from to five, and thus represent a the bit one. The edges from eight and nine, and six and seven are both rising edges, a zero bit is send.

In contrast with ChromaLux [5], where a photodiode was used as a receiver, a high-end smartphone camera is used. This allows us to explore the current limits of this technology. Using the rolling shutter effect continuous data can be received in a single picture [4]. A limitation is present here, as the authors of RollingLight [9] found that two consecutive pictures have an unpredictable time gap. This leads to packet loss. Vasilakis in DynaLight [14] proposes a solution to this problem to make sure that in every frame one packet is received.

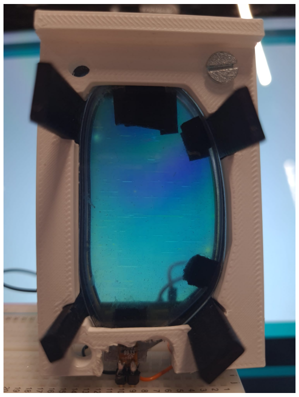
2.3 Encoding

Transmitting the actual data can be done with multiple encoding schemes. On Off Keying (OOK) turns the light on and off, each of these options corresponds with a different bit, 0 or 1. Another way to encode data is Amplitude Modulation (AM), The brightness (or amplitude) of the light is used to send data, commonly used with radio. Just like Frequency Modulation (FM) is used, having data correspond with specific frequencies. In ChromaLux [5] the authors discussed modulation techniques that could be used combining multiple liquid crystal cells called a stack. Setting a voltage over the stack produces a certain color, instead of settings this voltage exactly over the stack, larger voltage swings are used modulating the duty cycle of the voltage to average out over the wanted voltage. ChromaLux found that this achieves faster switching times, but does introduce a small amount of noise.

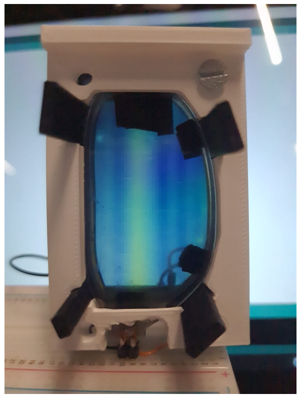
3 Harmful encoding

Research has shown that flickering between 3 and 200 Hz could potentially be harmful to humans [6]. These frequencies could be harmful to part of the population that suffers from epilepsy or could induce a headache in everyone. Even though the transmitter is an adaption from ChromaLux, the same encoding scheme will not be able to be used. Before the light hits the second polarizer the color does not change, only the polarization of the light. ChromaLux placed this right before the receiver, thus no extra care needed to be taken in making sure no harmful flickering occurred as humans. In this research, the second polarizer is placed at the transmitter. Care now needs to be taken to make sure the frequency of switching between colors is at least 200 Hz. Following this, a single color can not be displayed for more than $1/f = 1/200Hz = 5ms$. The encoding in ChromaLux does not do this, as can be seen from the state machine in figure 2. As the voltage does not change on the display when zeros follow each other, the maximum of 5ms per color could be reached with enough zeros.

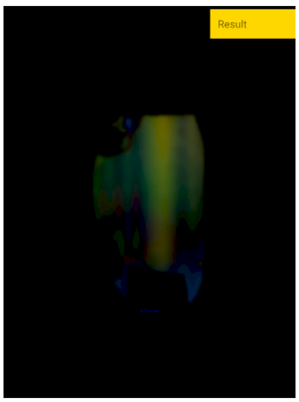
Manchester encoding can be used to mitigate this effect. This was used in Retro-VLC [10] and uses the edges to denote bits, instead of a one being a high voltage over the LCs, a one is a change from a low voltage to a high voltage. Differential Manchester encoding is an improvement where instead of the bit one always being the transition from a low voltage to a high voltage, it is the opposite of the last transition. This has the benefit that what the low and high voltages correspond to on the LC stack does not need to be known, as the difference in transitions will correspond with the data. In figure 3 an example of Manchester encoding can be seen, it can be seen the longest time a single symbol is displayed is in the header, three times the symbol length. From this it follows that a single symbol can not be displayed for longer than $5/3 = 1.66ms$. This corresponds with a frequency of $f_{sym} = 1/1.66 = 600Hz$. As long as this criterion is met, no harmful frequencies will be transmitted.



(a) High exposure time.



(b) Low exposure time, modulation becomes visible.



(c) Difference between high and low exposure image.

(d) OTSU filter applied to black and white image of difference.

Figure 4: Process of searching for the transmitter. Two pictures are taken, one with high and one with low exposure time. The difference is taken between these two pictures. This picture is converted to black and white, and an OTSU filter is applied.

4 Demodulation

The transmitted data is encoded in differential Manchester encoding as discussed in section 3. To decode this a picture is taken with a CMOS camera exploiting the rolling shutter effect [4].

4.1 Packet

Each of the pictures contains at least a single packet. As discussed in DynaLight, a packet should fit at least twice in the picture [14]. Together with repeating every packet once, a packet cannot be partially in one picture, and partially in the next. One of the two equal packets is always received correctly.

To determine the start and end of a packet, a header is used. This header contains a pattern that can not occur in Manchester encoding normally, a zero followed by three consecutive ones and ending with a zero. This can be seen in the DFA in figure 3. To determine the start of the packet the widest band in the picture must be found, and the end will be the start of the next packet.

4.2 Finding the data

When taking a picture, only part of the image will be of the transmitter. Image processing needs to be applied to detect where the data in the image can be found. In other works that use an LED as a transmitter, the image was passed into an OTSU filter, this filter turns a gray picture into a black and white image by making the brightest part of the image white, and everything else black. The contour of the transmitter was found to be white while other parts of the image were black. This will not work as the transmitter is more of an 'encoder'. It is not transmitting light but only encodes it with data. In the process, it is losing about 57 percent of the light intensity [5]. As the transmitter is now darker than the surrounding, another detection algorithm needs to be proposed. In this paper, we propose a new algorithm to detect the transmitter, ChromaCam.

This algorithm uses the fact that high-end cameras nowadays can reach frame rates up to 960 fps [12]. This frame rate is high enough to insert calibration frames when needed and at the same time not lose too many packets. These frames are taken with a higher exposure time, such that they will not show the rolling shutter effect. Together with the next picture that does show this effect, the difference between the two pictures is taken, the result can be seen in figure 4c. This image is converted to black and white, and an OTSU filter is applied. In figure 4 the whole process can be found. With the result of the OTSU filter, the boundaries of the transmitter are detected. This results in the red bounding box in figure 5. To track this location the colors and frequency of the modulation are captured.

The time between frames can be as little as one millisecond, thus the location of the transmitter will not differ greatly. Instead of searching the whole frame for the transmitter, only near the last known location of the transmitter needs to be searched for a signal with a known color and frequency. When the transmitter is found, the location will be updated and the data is demodulated. If the transmitter is not found, another calibration frame is inserted. As the calibration frame does take valuable time, even at 960 frames per second, as few as possible calibration frames should be used. In section 6 we discuss how to decrease the number of calibration frames needed.

4.3 Getting the data

To demodulate the data every vertical bar of the transmitter is averaged to a single RGB value, the three color channels can then be plotted separately, depending on what color the transmitter is oscillating between, one of these channels will hold the encoded data. Figure 5 shows the green channel holding the encoded data as this channel has the highest average change in value. To obtain the data from this graph, the header must be found. As can be seen from the DFA in 3, the header contains three consecutive ones. This will result in the widest band in the image, in figure 5 it results in the green band. As we know this part is the result of three consecutive ones, the width of a single symbol is also known. The data can now be decoded by matching this length to the rising and falling edges in the graph. Using this the differential Manchester encoded data can be interpreted.

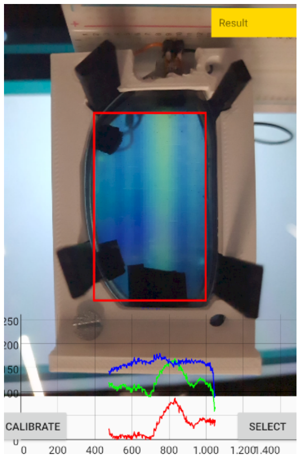


Figure 5: Picture where the location of the transmitter is detected, and the RGB channels are plotted.

To increase the throughput of ChromaCam, the optimal colors for the transmitter to modulate between need to be found, the optimal colors will have the highest change in contrast for the smallest change in time. This will decrease the overall width of the bands that encode the data, and more data will fit into a single frame. In ChromaLux [5] a way to determine these colors is discussed, this will be adapted to work with a camera.

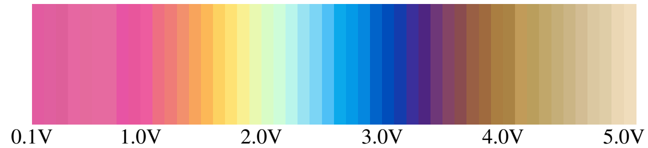
5 Platform

To determine the performance of ChromaCam, two seperate platforms are used, a transmitter and a receiver.

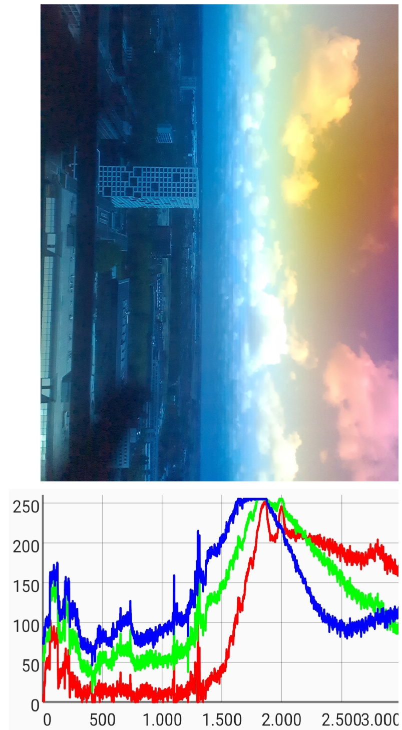
5.1 Transmitter

The transmitter consists of four liquid crystal cells with a height of seven centimeter modulated by an Arduino Uno operation at 5.0V. To determine the optimal voltages for sending the bits, the voltage over the liquid crystal cells versus the color is plotted in figure 6a. To get the fastest switching times a picture of a single transition is taken. The steepest slope of a single channel is the point where the deviation in color per change in voltage is the largest. If the colors in figure 6b are matched with the voltages that correspond to those colors in 6a it is found that between 2.2V and 2.6V the red channel has the highest voltage difference for the smallest deviation in voltage. These are the voltages for which this liquid crystals switch the fastest, achieving the highest bandwidth and contrast.

The Arduino Uno uses the build in timers one and two to send the data. Timer two controls the output pin to the liquid crystal cells, it uses Fast PWM mode to send a PWM signal with a frequency of 10kHz. The duty cycle can be edited by setting specific registers. The registers and what to set them to can be found in the datasheet of the ATmega328P [11]. To get the voltage of 2.2V and 2.6V, because the maximum voltage is 5.0V the duty cycle will be 44% and 52% of period of the 10kHz signal. Timer one will be used to send the symbols, every 0.5ms an interrupt service routine is called. This routine will change the duty cycle of timer two depending on the next symbol in the packet following the DFA found in 3. If the next symbol is a zero, timer two will have a duty cycle of 44%, a duty cycle of 52% is selected if the next symbol is a one.



(a) Colors that are produced by the LC shutter with a stable voltage from 0.1V to 5V with steps of 0.1V.



(b) Single transition from 0V to 5V.

Figure 6: When a voltage is applied to the liquid crystal cells, the red green and blue channel do not all respond equally fast. To determine the fastest changing channel a single transition is plotted. in (b) it can be seen that the red channel has the largest slope between two colors. These colors can be matched with voltages in found in (a).

5.2 Receiver

The Samsung Galaxy S20 is used as a receiver. The phone can go up to 960 frames per second at 1280 by 720 pixels with a field of view of 53 degrees. As the S20 contains a fast enough processor, demodulation can be done on the phone itself in real time. To process the pictures, the OpenCV library is used. This library is able to process pictures and videos, it also contains algorithms to follow objects. The library is extensively used for the difference between two pictures, an OTSU filter and tracking the transmitter.

Figure 7: Setup used with testing, the transmitter is posted against the window

6 Evaluation

To evaluate ChromaCam, the platform is tested for stability and data rate. Natural light is used by placing the transmitter in front of a window. During testing the light intensity fluctuated between 5 kLux when it is a cloudy day and 7 kLux on a sunny day. Figure 7 shows the setup.

6.1 Bit rate

The bit rate of the system depends on a multitude of factors, most important are the package size, distance to the transmitter, and symbol rate of the transmitter. If the packet size can be increased, more bits can be decoded in a single picture of the camera. The packet size can not be increased indefinitely, as the packet will no longer fit on the transmitter. For this same reason, the distance from the camera to the transmitter is an important factor. The size the transmitter takes on the screen decreases with the distance. Lastly, if we increase the symbol rate of the transmitter, a larger packet can fit in a single picture. This is however limited by the shutter speed of the camera and the switching speed of the transmitter.

Symbol Rate

To find the limit of how fast the transmitter can send a single symbol and the camera can receive this, the distance is minimized. For increasing symbol rates of the transmitter, the width of a single transition is measured. The result can be found in figure 8. After about 6000 symbols per second, no distinct transitions could be seen. Equation 1 is the regression line that conforms with the found data, with a width per transition of l_{sym}, and a symbol rate of s.

$$l_{sym} = 122217 * s^{-1.039} \tag{1}$$

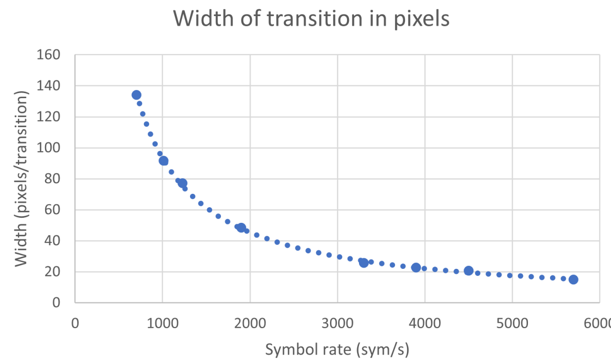


Figure 8: Width of a single transition plotted against the symbol rate of the transmitter.

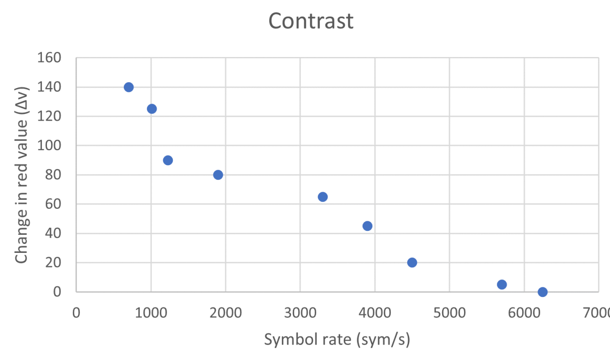


Figure 9: Difference between high and low symbol of the red channel at certain symbol rates.

As the symbol rate increases, the contrast between the high and low symbols decreases. This is shown in figure 9 where the difference in the red channel is plotted against the symbol rate. This decrease in contrast can be explained by how the transmitter is still changing its color. Because ChromaCam uses Manchester encoded data, where we are looking for falling and rising edges, the exact value does not matter. As long as the contrast is higher than the noise coming from the transmitter, data can be decoded.

Packet Size

To be able to reliably decode data, a symbol rate of 3000 symbols per second is used for sending the packets in ChromaCam. With this symbol rate, equation 1 shows that a single transition has a width of about 30 pixels. From figure 3 it is known that a packet is at least five symbol lengths long, and every bit added adds two symbol lengths. For a packet size l_{packet} of b bits, equation 2 shows the size of the packet.

$$l_{packet} = l_{sym} * (5 + 2 * b) \tag{2}$$

$$d_{whole} = \frac{\frac{l_{transmitter}}{2}}{\tan(\frac{FoV}{2})} \tag{3}$$

The Samsung Galaxy S20 has the whole transmitter in

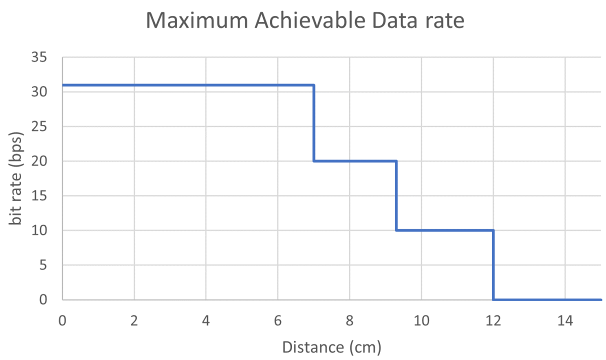


Figure 10: Datarate plotted against the distance to the receiver.

Figure 11: Example of the transmitter at seven centimeter sending data at 3000 symbols per second.

frame at a distance of seven centimeters, with general geometry equation 3 is found, it generalizes the distance the whole transmitter is in frame for a camera with a field of view FoV and the height of a transmitter $l_{transmitter}$. For every distance smaller than this distance the whole transmitter is in frame. As the resolution of the receiver has a limit of 720 pixels, keeping in mind that at least two packets need to fit in a frame as seen in section 4.1, the max length of a packet is $720/2 = 360$ pixels. Rewriting equation 2 shows that at a distance of seven centimeters three bits can fit into a single packet. For every factor x with which the distance is increased, the max size of a packet is decreased by this same factor x. From this, it follows that if one bit is transmitting with every packet, the maximum distance is twelve centimeters.

On average decoding a single frame takes about 95 milliseconds, with a packet size of three bits, this means a bit rate of 31 bits per second can be achieved by ChromaCam with a stable link at a distance of seven centimeters. At a distance of twelve centimeters, the bit rate is reduced to ten bits per second. Figure 10 shows the bit rate as a function of the distance. An example of the transmitter at a distance of seven centimeters is shown in figure 11.

6.2 Calibration Frames

The OpenCV library proved useful for tracking objects. After an initial calibration frame, if the transmitter was visible in a frame, the tracking algorithm called "MOSSE" was able to locate it. Only when the transmitter was out of frame a calibration frame needs to be inserted again. Thus using this library proves useful for the reliability of the link between transmitter and receiver.

On average the extra steps to locate the transmitter when calibration is needed takes an extra 45 ms. This time is mostly calculating the difference between the two pictures taken. This adds about 50% to the processing time of a frame, and thus reduces the bit rate by one third if a calibration frame was needed every time.

7 Future Work

The achieved bit rate is at the time limited by the processing speed of the individual frames. If the tracking procedure can be improved and be made faster, large improvements in the bit rate can be found up to 750 bits per second, as this is the maximum the transmitter supports.

Another place for improvements is the effect of an optical zoom found in the newest phones. This could potentially lead to large improvements in the distance data can be decoded from.

One problem found during evaluation of ChromaCam was background noise. The color of the light coming from the receiver depends on what is behind the liquid crystal stack. The colors the liquid crystals are switching between can be obscured depending on the color of the incoming light. Research can be done to make the colors of the LCs more vibrant and thus better readable, even with background noise.

8 Responsible Research

In this section the reproducibility and integrity of this paper is discussed. As discussed by S. Verma, the performance of light-to-camera communication depends on many factors, that makes direct comparison not possible. ChromaLux also discussed the difference in liquid cells and found a difference of 0.5 millisecond in some cells [5]. This difference can make the system unusable. To overcome these difficulties, every setting is documented and the steps to optimize the performance for a particular setup is shown. Furthermore, every step in the algorithm is clearly explained. Libraries that were used are mentioned and are open source. This should make it possible for other researchers to reproduce the ChromaCam algorithm.

9 Conclusion

In this work, we looked into how to demodulate an optical signal with a high-end smartphone. An algorithm called ChromaCam was proposed to locate the transmitter. This algorithm takes the difference between two consecutive pictures taken with different shutter speeds to locate the transmitter, then uses common tracking algorithms from OpenCV to follow the transmitter. This transmitter works based on the basis of passive visible light communication. It is not transmitting light, instead it is modifying and changing the color of the incoming light. To test ChromaCam, a prototype was used. At a distance between transmitter and receiver of seven centimeters, a bit rate of 31 bits per second was able to be reached. This proves the viability of ChromaCam.

References

[1] Alexander Graham Bell. The photophone. *Science*, os-1(11):130–134, September 1880.

[2] Jona Beysens, Ander Galisteo, Qing Wang, Diego Juara, Domenico Giustiniano, and Sofie Pollin. DenseVLC. In *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*. ACM, December 2018.

[3] Rens Bloom, Marco Zúñiga Zamalloa, and Chaitra Pai. LuxLink. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*. ACM, November 2019.

[4] Christos Danakis, Mostafa Afgani, Gordon Povey, Ian Underwood, and Harald Haas. Using a CMOS camera sensor for visible light communication. In *2012 IEEE Globecom Workshops*. IEEE, December 2012.

[5] Seyed Keyarash Ghiasi, Marco A. Zúñiga Zamalloa, and Koen Langendoen. A principled design for passive light communication. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. ACM, September 2021.

[6] IEEE recommended practices for modulating current in high-brightness LEDs for mitigating health risks to viewers.

[7] T. Komine and M. Nakagawa. Fundamental analysis for visible-light communication system using LED lights. *IEEE Transactions on Consumer Electronics*, 50(1):100–107, February 2004.

[8] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. Luxapose. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, New York, NY, USA, September 2014. ACM.

[9] Hui-Yu Lee, Hao-Min Lin, Yu-Lin Wei, Hsin-I Wu, Hsin-Mu Tsai, and Kate Ching-Ju Lin. RollingLight. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, May 2015.

[10] Jiangtao Li, Angli Liu, Guobin Shen, Liqun Li, Chao Sun, and Feng Zhao. Retro-VLC. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*. ACM, February 2015.

[11] Microchip Technology. *8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash*, January 2015.

[12] Samsung. How does super slow-mo video work on galaxy s20, s20+, s20 ultra, and z flip?

[13] Zhao Tian, Charles J. Carver, Qijia Shao, Monika Roznere, Alberto Quattrini Li, and Xia Zhou. PolarTag. In *Proceedings of the 21st International Workshop on Mobile Computing Systems and Applications*. ACM, February 2020.

[14] Michail Vasilakis. *DynaLight: A Dynamic Visible Light Communication Link for Smartphones*. PhD thesis, Delft University of Technology, November 2015.

[15] Shashwat Verma. *Analysing the Performance and Stability of LED-to-Camera Links*. PhD thesis, Delft University of Technology, December 2017.