Delft University of Technology
Master of Science Thesis in Embedded Systems

Under-Screen Camera Detection

Daniël Jan Hendrik van der Kolk

TU Delft Delft University of Technology

Under-Screen Camera Detection

Master of Science Thesis in Embedded Systems

Embedded Systems Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands

Daniël Jan Hendrik van der Kolk

29^{th} of June 2023

Author
Daniël Jan Hendrik van der Kolk
Title
Under-Screen Camera Detection
MSc Presentation Date

3^{rd} of July 2023

Graduation Committee

Prof. dr. Koen Langendoen (Chair)	Delft University of Technology
Dr. Qing Wang	Delft University of Technology
Dr. Kaitai Liang	Delft University of Technology
Hanting Ye	Delft University of Technology

Abstract

Hidden spy cameras are a growing worldwide threat to people's intimacy and privacy. With the growing interest in full-screen devices and the underlying development of under-screen cameras, a new type of potential security risk is introduced. Recent smartphones such as the ZTE AXON 40 already demonstrate that it's infeasible to detect the camera with the human eye. There exist several techniques to detect hidden cameras, however most of these techniques are not resilient to the unique deployment scenario of the under-screen camera. A recent optical detection technique, which relies on the retro-reflective effect of hidden cameras, is promising but is also greatly hindered by challenges introduced due to reflections from the screen that is placed in front of the under-screen camera. In this work, these challenges are addressed, by proposing a detection principle that exploits the difference in reflective nature between the USC and the screen. Using reflection detection in a sliding window approach, a detection methodology is given to detect the USC. Furthermore, a detection architecture is designed that incorporates the proposed detection principles using a combination of computer vision, image processing and machine learning techniques. Using an off-the-shelf Time-of-Flight sensor, this architecture is implemented into a detection system and evaluated on its robustness and detection accuracy. Experiments on a dataset of 200 videos with a variety of measurement conditions show that this detection system is capable of achieving a USC detection rate of 71.5% while having a false-positive rate of 21.5%. It also proves excellent results while the screen is displaying content.

Preface

I can say that it is with great pleasure to present you this master thesis work. First, I would like to thank my supervisor Qing Wang for his professional guidance and for providing me the opportunity to do this master thesis project on this particular interesting topic. Without Qing, I would not have had the opportunity to gain insights into the research on under-screen cameras, which was completely new to me when I started this thesis. Second, I want to give thanks to my other supervisor Hanting Ye for all the effort he put into the great discussions we had, in helping me to write this thesis, and also for the fun talks we had. I admire his everlasting openness and thoughtfulness during the year I spent on this research. Last, but not least, I want to thank my family, friends and colleagues for helping me, all in their own way, through the busy times I had during this thesis. In particular, I want to thank Koen Goedemondt for connecting me with Qing.

Daniël Jan Hendrik van der Kolk

Delft, The Netherlands
29th June 2023

Contents

Chapter 1

Introduction

Hidden spy cameras recording people in private spaces have increasingly become a worldwide problem. For example in South Korea in 2021 alone, a total of 5541 spycam related crimes were administrated [1]. Especially locations such as Airbnb accommodations have a high interest for attackers to install these hidden cameras. Research in 2019 among rental accommodation visitors showed that 11% has actually found a hidden camera in one of their rental visits in the past [2]. The rise of these spy crimes is not only negatively impacting the general trust in the protection of people's privacy but also has a huge impact on the health and life of the victims. These victims, mostly women, suffer from depression, anxiety and in the worst cases even attempt suicide [3]. To minimize and prevent the impact of these hidden cameras, a great effort has been put into developing detection principles and devices that aid people to find these hidden cameras.

In the last years, a growing interest is shown in the trend of maximizing the screen-to-body ratio of devices such as smartphones, tablets and laptops. The goal is to create *full-screen* (bezel-less) devices that provide several advantages such as (1) a cleaner industrial design; (2) an improved user experience and (3) a more natural viewpoint during videoconferencing [4],[5]. The ultimate goal is to create a *full-screen* device. At present, there exists a variety of full-screen devices such as laptops ((e.g., Thunderobot T-BOOK 14), smartphones (e.g., ZTE AXON 20/30/40, Xiaomi MIX4, and Samsung Galaxy Z Fold 3/4) and even TVs [6]. This full-screen device introduces a new technology, called the *under-screen camera* (USC)[1]. This technology hides the camera behind the screen using a special translucent region. Although the first smartphone (ZTE AXON 20) received strong critiques [10] on the camera's performance, the development of this technology has continued and many new models with a USC have been released by a variety of smartphone manufacturers, showing a strong interest and potential in this new fashion.

However, this new type of camera also brings new types of potential security risks. Since the translucent screen of full-screen devices equipped with a USC allows light to pass through while also being able to display content, it makes the camera unnoticeable for the human eye, especially when content such as videos is displayed on the device. This makes the USC a new attack vector in the field of spy camera crimes. As this technology is not only limited to smartphones, any screen can potentially contain a spy camera, making it unfeasible to detect with the naked eye. Another potential threat is the possibility of impersonating people using deepfakes. Good quality deepfakes currently rely on the scale and variance of the training set, requiring several thousands of images with diverse facial variations [11]. Because humans naturally will look to screens, the USC is also capable of recording more detailed and diverse information of the victims compared to the typical hidden camera, making it a good candidate to create strong deepfakes of people.

[1]It is also referred to as Under-Display Camera (UDC) in the literature [[7], [8], [9]]

Because detecting hidden cameras is a challenging task without the aid of specialized technology, recent academic works propose to detect the presence of hidden cameras in an automated fashion, leveraging commercially available devices. One popular technique used is analyzing the wireless traffic they generate [12]. However, this can only detect the presence of the hidden camera, and not their exact location. Another way is using thermal analysis [13]. However, the screen itself will also heat up, which will affect the detection of the camera. In light of the above limitations, the following question is asked: *can a solution be proposed that only leverages existing commercial devices to automatically detect and localize under-screen cameras?* This research will focus on a detection technique that is based on optical reflections [14]. This technique leverages the fact that cameras show strong reflections, also known as *retro reflections*, when light hits the camera's surface. Using a Time-of-Flight (ToF) sensor, a widely available device on the commercial market as well as in modern smartphones, this retro-reflective effect can be detected [15]. There are however challenges when applying this technique to USCs. The first challenge that arises is that the screen of the USC device itself also has strong reflectivity, which will bring a lot of disturbing noise into the detection models. The second challenge is that the screen greatly limits the Field-of-View (FoV) in which the retro reflections from the USC are visible. To combat these challenges, this work proposed a novel detection model that exploits the difference in the reflective nature between reflections originating from the USC and reflections originating from the screen. A proof-of-concept detection system is implemented using an off-the-shelf ToF sensor and is evaluated on a variety of conditions using several state-of-the art full-screen smartphones to assess its performance and robustness.

1.1 Research objective

The objective of this thesis is to provide clear insights into the feasibility of detecting the presence of under-screen cameras. The thesis should bring forth the challenges of detecting USCs and core principles that can be used to build a detection model capable of predicting whether a screen contains an under-screen camera. This solution should be robust under a variety of conditions. More specifically, it should work while content is displayed on the screen.

1.2 Research challenges

The research challenges of this thesis are formulated as follows:

1. *Design a detection model that is able to detect the presence of under-screen cameras.* Detecting the USC is a challenging task, because of its hidden nature and the reflectivity of the screen. Therefore, a clear detection model should be designed that can overcome these challenges.

2. *Design a USC detection system that is robust under a variety of conditions.* Because under-screen-cameras can be deployed in a variety of conditions, another challenge of this research is to design a detection system that is able to operate under a variety of conditions. Especially the detection system should work while the screen is displaying content, because this is a challenging condition to detect the camera with the naked eye.

1.3 Contributions

The following list summarizes the contributions brought forth by this thesis:

1. An in-depth review of state-of-the-art research into hidden-camera detection techniques, alongside the functional principles of USCs and ToF sensors.
2. An exploration into the feasibility of discerning USCs utilizing optical reflections.
3. A novel design of a detection model that is able to address the challenges that come with detecting USCs based on retro-reflections. The detection model solves the challenges by exploiting the difference in reflective nature between reflections from the screen and reflections from the USC. It provides a clear methodology and uses a combination of in-screen reflection detection and a sliding window to capture this behavioral difference.
4. A detection architecture integrating the proposed detection model. The detection architecture comprises screen detection, reflection detection, and decision modules, leveraging computer vision and machine learning techniques.
5. An implementation of a detection system based upon the proposed USCs detection architecture using an off-the-shelf ToF sensor. Comprehensive experiments were conducted using state-of-the-art full-screen smartphones. The detection system demonstrates a USC detection rate of 71.5% while maintaining a false positive rate of 21.5%. The model performs well under a range of conditions, successfully detecting a USC even when the screen is actively displaying content.

1.4 Thesis content

This thesis aims to provide a full overview of the detection principles, system design and architecture and evaluation of the detection system that is proposed to detect under screen cameras. Furthermore, a survey on related work in the field of hidden camera detection and their challenges is given. The contents can be summarized as follows:

1. *Introduction* provides the motivations behind detecting under screen cameras, and describe the main challenges which will be addressed in this work.
2. *Background and Related Work* will provide a brief investigation into the technological principles of USCs and a review of state-of-the-art research in hidden camera detection. Also, background knowledge on the principle of retro reflections is given.
3. *Feasibility and Challenges* first shows the feasibility of detecting the USC using retro reflections but will also addressing the difficulties and challenges that arise when using optical detection methods to under screen cameras.
4. *Detection Model* will present the core principles that should be used to detect under screen cameras using optical reflections. Furthermore, a methodology is given that integrates the core principles into a detection model.
5. *System Architecture* details the architecture of a detection system based upon the detection model and its methodology given in Chapter 4.
6. *Implementation* will describe an implementation of the system architecture that will be used to evaluate the detection model.

7. *Evaluation* will outline the methods used to test and verify the proposed solution implemented. First an evaluation is performed on the detection system using a simple setup in order to investigate the models effectiveness and limitations and to optimize the models parameters. Second a more generic evaluation of the detection system is done under a variety of setups in order to verify the robustness of the detection system.

8. *Discussion and Future Work* contains a discussion of the results and provides future work recommendations.

9. *Conclusion* summarizes the results obtained in the thesis.

Chapter 2

Background and Related Work

2.1 Under-screen cameras

The development of USCs is at this moment the latest advancement in the technological arms race to create bezel-less or full-screen devices. Although the development of USCs is still its early stages, it has drawn quite some attention from several smartphone manufacturers. The first official public release (ZTE AXON 20) was in September 2020 [16] and currently the amount of smartphone models containing this technology that have been released has passed ten [17]. While its appearance, a camera placed behind the screen, remains the same, there is large variety in how the technology is functionality implemented. This section aims to explain the current state-of-the art USC technology and will lay out the applications it has and briefly tell its future expectations.

2.1.1 Operating principle

Devices equipped with USCs generally utilize translucent Organic Light-Emitting Diode (OLED) screens to facilitate content display while minimizing light loss [18]. This is achievable due to the translucent cathodes and organic layers that constitute OLEDs. However, the transmittance of OLEDs is not sufficient to produce high-quality images, necessitating a different approach to the design of the translucent screen's pixel layout to improve image quality.

As depicted in Figure 2.1, reproduced from [19], this design approach involves replacing a substantial portion of the screen's pixels with translucent areas. Various pixel layouts are possible, such as removing 3 out of 4 pixels, significantly enhancing the signal-to-noise (SNR) ratio of the image.

Nevertheless, situating the camera behind the screen negatively impacts the quality of images captured by the USC. Firstly, the display obstructs a notable portion of the incident light from the scene, thereby diminishing the SNR ratio of the captured image. Secondly, the display introduces diffractive blur to the captured image, a phenomenon that presents significant challenges in terms of filtering and removal. To mitigate these effects, pixel layout optimization can be deployed to minimize the impact of diffraction [20].

Improving the design of the translucent display remains a critical area of focus in the quest to enhance image quality. Another prevalent approach involves the utilization of deep neural networks (DNNs) to bolster image quality [7], [8], [9]. This approach involves training neural networks to identify and correct complex blurring and diffractive patterns introduced by the translucent screen region.

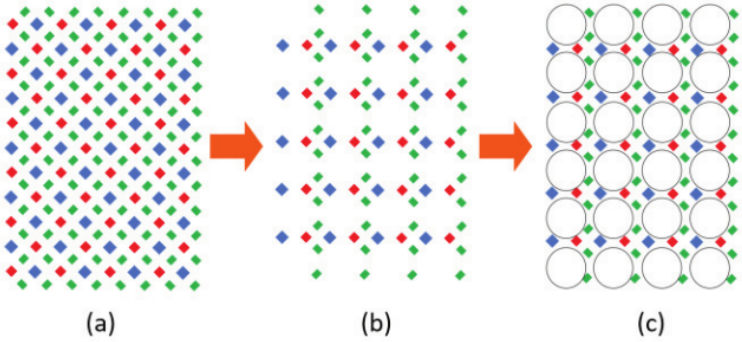


Figure 2.1: **(a) Regular pixel layout; (b) Pixel layout after removing one half of pixels; and (c) Transparent area for removed pixels (reproduced from [19])**

In addition to image restoration efforts, recent studies have examined the potential use of under-screen sensors to facilitate visible light communication tasks [6], [21]. However, the primary focus of this thesis lies in examining the potential risks associated with USCs, such as their possible utilization as spy cameras. We specifically concentrate on the detection of USCs in commercially available full-screen devices, an aspect that is largely impervious to the diverse screen pixel layout optimization strategies employed by various manufacturers, which has never been investigated.

2.2 Hidden camera detection

Because of the worldwide concerns people have in the possibility of being spied, a variety of techniques have been developed and research has been done in the past years to detect the presence of hidden cameras. As a feasibility study for this research, these techniques will be examined on their applicability to the detection of USCs. First a review will be done on the current state-of-the art techniques. Secondly, a more in-depth examination on the use of optical reflections as detection technique will be done, as it will be shown that this method is on forehand the most suited method to detect USCs.

2.2.1 State-of-the-art techniques

To explore the state-of-the-art techniques used in hidden camera detection research, recent papers and surveys on hidden camera detection have been examined. The techniques that are used in this research can be placed in three different categories:

- *Network Traffic* - The idea behind this type of research is to analyze the network traffic and detect patterns that are related to spy cameras. In this way the near presence of a hidden camera can be determined, although it will not disclose its location. This is a challenging research because of the large amount of devices generating network traffic nowadays. Also, this research is limited to wireless cameras that continuously stream their data to the attackers. Using fine-grained techniques this research is however very successful within these constraints. Dinhnguyen Dao et Al. [12] developed a detection system called *DeepDeSpy* which convolutional neural network that is able to real-time detect patterns

inside the channel state information caused by physical movements that camera records. This model achieved a detection accuracy of 96%.

- *Thermal Emissions* - In this type of research, the heat dissipation of cameras is used to detect their presence. The idea of using thermal cameras has been proposed as such a solution [22]. While being a feasible approach, it comes with the downside that it heavily relies on user expertise to distinguish the camera. However, a recent approach called *HeatDeCam* [13], proposed a novel solution leveraging a user-friendly thermal-based detection model. In this work, a neural network is deployed that is capable of detecting the unique heat dissipation patterns of spy cameras. The model is evaluated on a training set of 22506 thermal and visual images, collected from 11 spy cameras setup in a variety of environments. For this dataset, the model achieved an accuracy of 95%.

- *Optical Reflections* - A common way to detect hidden cameras is using the principle of optical reflections. The idea here is that a camera lens, when illuminated by light, will show strong, bright reflections which makes it distinguishable from its surroundings. A variety of commercial hidden camera detectors use this principle [23] and it's a common theme in online articles about hidden-camera detection [24]. The drawback is that it's hard to use these tools effectively because the reflections are only visible from a limited FoV. Also, it's difficult for an inexperienced person to differentiate reflections of cameras from other reflective materials. A recent work called *LAPD* [15], created a smartphone application that addresses these challenges and leverages the use of the smartphone's ToF sensors to detect these reflections. An automatic detection model is proposed that aims to detect reflections of hidden cameras while ignoring reflections from other objects. In a real-world experiment with 379 participants, this method achieves an 88.9% detection rate while so naked eye yields only a 46.0% detection rate.

This research focuses on predicting and verifying if a camera screen contains a hidden camera. Detection methods based on network traffic analysis are therefore not of interest as they are eventually limited to only detecting the presence of a hidden camera in near surroundings rather than providing a screening or validation functionality for specific objects. Detection methods based upon thermal emissions also have limitations when using these to detect USCs. Although research shows that an OLED screen on its own has homogeneous temperature distributions while electronic components placed behind the screen cause non-uniform temperature distributions [25], it will be hard to correlate the non-uniform distributions to heat radiations of a camera. The increase of research and technology [26], [27] focusing on placing electronics beneath the screen in order to increase the spatial efficiency of devices will make this even more challenging or even infeasible for future screen devices. Optical detection methods rely on the principle of creating reflections in the camera lens and being able to observe these reflections. The reflections are created by a hidden camera detection device emitting light into the camera lens. In order to record videos, a camera lens should be able to capture light. Therefore, if a hidden camera is recording people, it will also record (and reflect) the light emitted by the hidden camera detection device.

2.2.2 Retro reflections

Because this research will focus on detecting hidden cameras using the principle of optical reflections, this section will provide a more in-depth analysis of the operating principles of this technique. In Figure 2.2a a simplified, conceptual model is shown of a typical camera module. It consists of the following three components:

(a) **Conceptual model of camera.**

(b) **Simplified model of the retro reflection effect in a camera module.**

Figure 2.2: **Conceptualization of reflections in a normal camera**

- A spherical or circular **lens** which will refract light into the camera module.
- A **filter**, also known as Bayer filter, which is used to filter and guide the light that will pass to the image sensor. It can be used to filter out undesired wavelengths such as IR and UV since they are not of interest for capturing images. Also, it will properly align color wavelengths to the image sensor.
- An **imaging sensor** which will converts light into an electrical signal which can be processed and digitized. This is usually implemented with CCD or CMOS sensors.

The reason why optical reflections can be used to detect hidden cameras is because of the so-called *retro reflection* effect. Retro reflection, also known as the cat-eye effect, occurs when an incoming light beam is reflected back to the beam source, or in other words: the angle of reflection of the incident light equals its angle of incidence. The energy of the reflected beam is of a higher magnitude than that of diffuse reflections [14], and the reflections are therefore clearly visible. This effect can also be seen in cameras as is shown in Figure 2.2b. When a light beam hits the lens' surface, it is refracted and propagated inside the camera module and will reflect in two possible ways: either it will reflect on the camera's filter as is the case of wave **A**, or it will reflect on the image sensor as is the case for wave **B**. In either cases, the waves will again refract in the lens and reflect back directly to the beam source.

The conditions for detecting these retro reflections are however limited by the following two factors [14]:

1. *Distance*: The intensity of the reflected light beam decreases as the beam source's distance to the camera increases.
2. *Field-of-View*: Because of the camera's housing the reflections for typical mini cameras are only visible inside a limited FoV, approximately a 20 degrees cone from the camera's center. The reflected intensity is the strongest at the center of this cone (e.g. when the beam source is in parallel with the camera), and decreases while moving outside the cone. Because of the limited translucent screen region that hides the USC, this effect is also expected for the USC.

Time-of-Flight Sensors

Commercial hidden camera detectors predominantly rely on optical-based detection methods [23]. These devices deploy bright LEDs, which induce retro reflections from concealed cameras. Users manually inspect suspicious objects and, by visually discerning reflections or employing an integrated viewfinder within the device, ascertain whether the reflections originate from a hidden camera. This method, as previously highlighted, encounters several impediments, the principal one being the challenge in distinguishing reflections from hidden cameras and those from other reflective objects.

Recent advancements have demonstrated that ToF sensors, special infrared cameras incorporated in most modern smartphones for distance measurement, can effectively detect retro reflections [15]. Figure 2.3 presents an overview of the components constituting a ToF sensor. It includes an emitter, typically an array of LEDs or laser diodes, which projects IR light onto the scene, and a camera lens that captures the reflected infrared light.

The light is modulated with a square wave (other wave shapes are also utilized), and the pixel matrix is composed of an array of capacitors connected to the modulation component generating the square wave. This arrangement ensures that the resulting charge in each capacitor directly corresponds to the phase difference between the emitted and received square wave.

Distance estimations can be determined using the phase difference between the illuminated light and the captured reflections, as shown in Figure 2.4. The distance d can be computed using the speed of light (c), and the time-delay Δt due to the phase shift: $d = \frac{c \times \Delta t}{2}$.

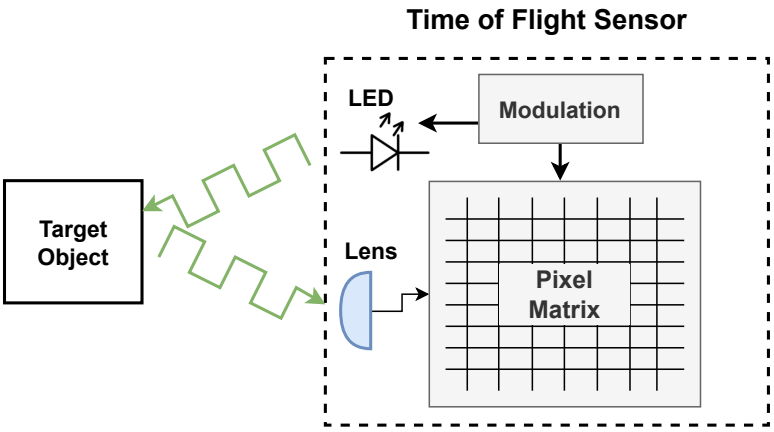


Figure 2.3: **Overview of components in ToF sensor.**

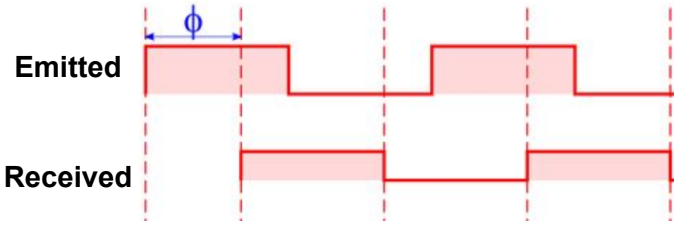
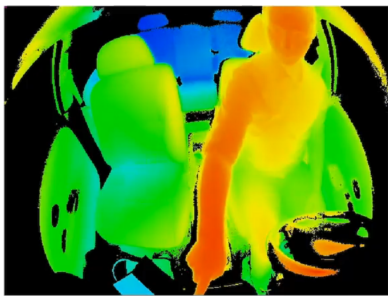


Figure 2.4: **Phase difference between emitted light and received light.**

Using the information from the pixel array, two images can be generated: (1) a depth image, shown in Figure 2.5a, in which each pixel represents the distance; and (2) an amplitude image, shown in Figure 2.5b, in which each pixel represents the strength of the received reflection. The amplitude image can be exploited to detect retro reflections from the camera because the strength of the received reflections should be very high compared to other pixels.



(a) **Depth image of ToF sensor**

(b) **Amplitude image of ToF sensor**

Figure 2.5: **Different output of ToF sensor.**

Chapter 3

Feasibility and Challenges

This chapter will discuss the feasibility and challenges for detecting USCs using detection methods based upon optical reflection. First, the results of a feasibility study on the detectability of the USC using optical reflections are discussed. Based on the results of this feasibility study, challenges that arise when detecting USCs will be thoroughly discussed in the subsequent sections.

3.1 Feasibility Study

In Figure 3.1 an illustration is given of the four major components of a USC that light needs to traverse before it will reach the Image Signal Processing Unit (ISPU) such that an image can be constructed. Optical reflection based detection method rely on the retro-reflective effect of the camera. This effect is observed when light is reflected by either the filter or the imaging sensor of the camera (see Section 2.2.2). Because the model of the USC is similar to normal cameras, except for the presence of a screen, the same optical detection principles can be applied if the light emitted by detector device can traverse the OLED screen.

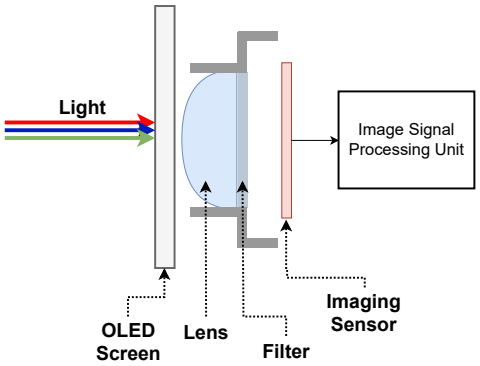
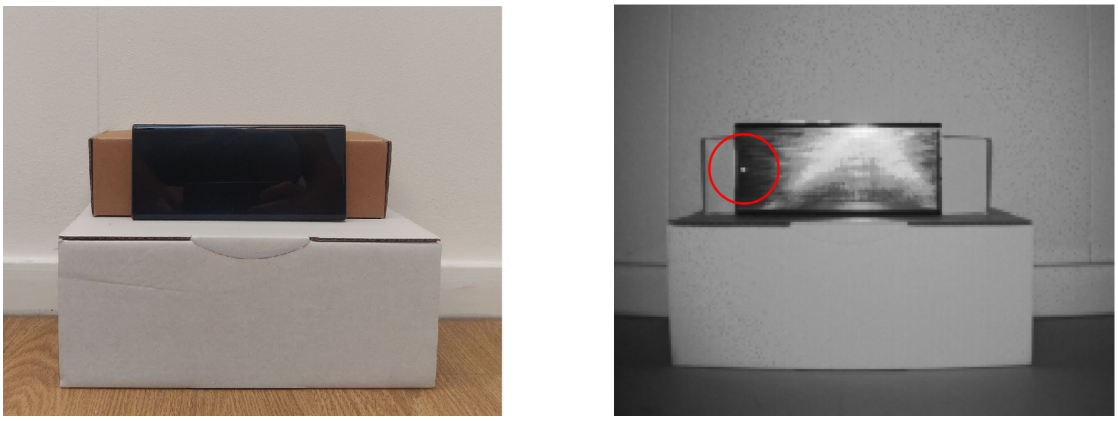


Figure 3.1: **Before light reaches the ISPU, it traverses the translucent OLED screen, camera lens, filter and imaging sensor.**

In order to verify whether the OLED screen does not block the light emitted by the ToF sensor, or weakens the reflections too much, a feasibility study has been done on one of the most recent full-screen smartphones, the ZTE AXON 40 Ultra [28]. In Figure 3.2a an RGB

image of this smartphone can be seen while Figure 3.2b shows the amplitude image of the same smartphone captured by a ToF sensor. The images clearly demonstrates that even though the USC is invisible in the RGB image, its retro reflections are clearly visible in the ToF sensor's amplitude image. To verify whether the retro reflections were actually caused by the camera, experiments were conducted in which the USC was removed from the smartphone. In these experiments, the retro reflections were not observed. From this it can be concluded that the USC still causes retro reflections even though it's installed behind the screen. There are however challenges noted during the feasibility study that will be addressed in next section.



(a) **RGB image** (b) **Amplitude image of ToF sensor**

Figure 3.2: **Figure 3.2a shows an RGB image of a smartphone with a USC (ZTE AXON 40). Figure 3.2b shows the amplitude image of the ToF sensor with the retro reflections of the USC clearly visible (annotated with red circle).**

3.2 Challenges

Although the feasibility study shows that the retro-reflective effect can still be observed for USCs, there are also several challenges found. While the screen still allows the ToF sensor to cause retro reflections, it also causes difficulty in detecting these retro reflections. The challenges caused by the screen can be divided into two main categories, that will be examined in detail in the following sections.

1. *Reflectivity of screen*: the reflectivity of the screen causes spurious reflections and interferes with the reflections of the USC.

2. *Hidden nature of USC*: the by default hidden nature of the USC increases the search space and complexity for the detection methodology.

3.2.1 Reflectivity of screen

Optical based detection methods rely on the principle of detecting retro reflections from a hidden camera. The retro reflections cause high-intensity reflections which can be distinguished from other reflections. In the case of USCs, there is a translucent OLED screen placed in front of the USC. OLED screens are reflective surfaces and show complex scattering patterns due to the

periodic pixel structures [29]. Because optical detector methods operate by emitting light and observing the reflections, the screen will also reflect the emitted light by the detection device. In Figure 3.3 this is effect is demonstrated, using a ToF sensor as detection device. The effect of these reflections on the detection of the USC causes two types of challenges:

1. *Interference with USC Reflections*: the large reflection component **A** shown in Figure 3.3 is the strongest component of the screen's reflections. When the USC is placed behind the location of this component, it will be impossible to detect its reflections because of the interfering screen's reflections.

2. *Noisy Reflections*: another challenge caused by the screen is the introduction of small noisy reflections such as **B** shown in Figure 3.3 of which the intensity profile is similar to the reflections caused by camera's. It's hard to distinguish these reflections from reflections caused by the USC.

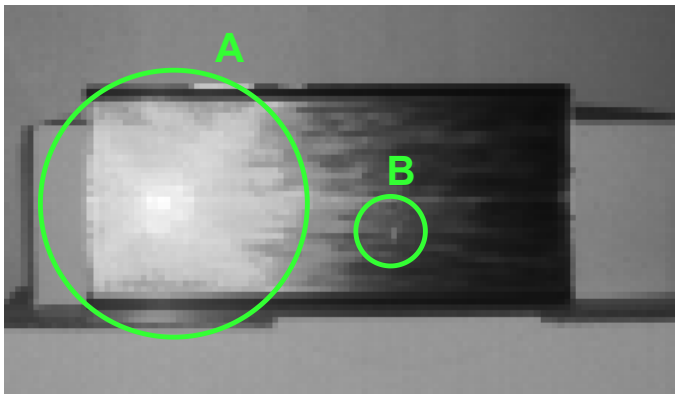
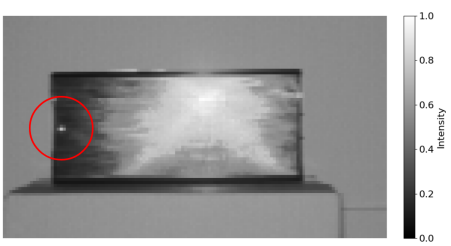


Figure 3.3: **Reflections caused by the screen. The large center component A interferes with the USC's reflections, while small components such as B introduce noisy reflections.**

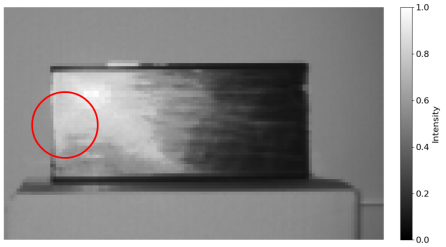
Effect on Field-of-View

In Figure 3.4 the effect on the detectable FoV is shown and in Figure 3.5 this effect is schematically shown. First, in Figure 3.5a the situation is given for a normal camera, where the FoV of the camera's reflections can be captured by the ToF sensor within a cone of angle θ_N (see Section 2.2.2). When a screen is placed before the camera, the specular reflections of the screen interfere with the reflections of the USC. The intensity of the specular reflections is the strongest when the angle of incidence is 0°, and the intensity decreases exponentially when the angle of incidence increases [29]. There is a certain cone, centered around the 0° angle of incidence, where the interference of the specular reflections becomes dominant and reflections originating from the USC become indistinguishable. This is illustrated in Figure 3.5b, where the angle of this cone of interference which blocks the normal FoV θ_N, is given by θ_B.

The USC is only visible inside on of the two side lobes of the original cone of angle θ_N that is not blocked interference of the screen's reflections. The angle of each of these two lobes, $\theta_{usc-side}$, in which the USC should be visible and detectable is given by equation 3.1. The challenge in finding the visible part of the USC is therefore dependent on two factors:



(a) **The USC is outside θ_B and inside θ_N**



(b) **The USC is inside θ_B**

Figure 3.4: **Example showing how specular interference makes the USC undetectable.**

(a) (b)

Figure 3.5: **Effect of reflections from screen on the detectable FoV of the USC compared to situation with normal camera.**

1. θ_N: this depends on how well the reflections of the USC can reach the ToF sensor and depends on the USC's deployment model.
2. θ_B: this depends on factors such as the reflectivity function of the OLED screen, and patterns inside the light beams emitted by the ToF sensor that cause different reflections in the screen.

As one can imagine, computing these factors is very challenging, simply because the variety of deployment scenarios is just too broad and to complex to accurately model. It will therefore be a challenge to find the visible lobes of angle 3.1 for the USC without prior knowledge of the location of the USC.

$$\theta_{usc-side} = \frac{\theta_N - \theta_B}{2} \tag{3.1}$$

3.2.2 Hidden nature of USC

As shown in Chapter 2, hidden camera detection methods aim to identify objects that contain hidden cameras or provide the exact location of the hidden camera. In order to do so, usability is an important part of the system design of these detection methods. The proposed solution should be less complex than a trial-and-error search process that users could do by themselves, but also should be more accurate and result in low false positives than the judgement of the user. With this in mind, the by default hidden nature of USCs, imposes challenges on designing a clear detection methodology:

- If a USC is not detectable by the human eye, the whole screen potentially contains a USC. Other optical reflection detection methods use the pre knowledge of a suspicious hidden camera location. While the first full screen smartphone such as the ZTE AXON 20 (2020) have USCs of which the location can still be distinguished by the human eye, the latest full-screen smartphone such as the ZTEA AXON 40 (2022) has proved that it's impossible for human eyes to detect the USC location without any prior knowledge. Therefore, it should be assumed that the whole screen can possibly contain a USC.

- There are lots of different screen dimensions and different USC optical parameters. While the validation in this work is scoped to detecting USCs in smartphones, the detection principles should also work for other types of screens. In Figure 3.6, several OLED screens to scale are shown that all contain technology that enables hiding a malicious USC. It can be easily imagined that if detecting a USC for a smartphone is challenging, detecting the USC in a 55-inch display is impossible without proper methodology.

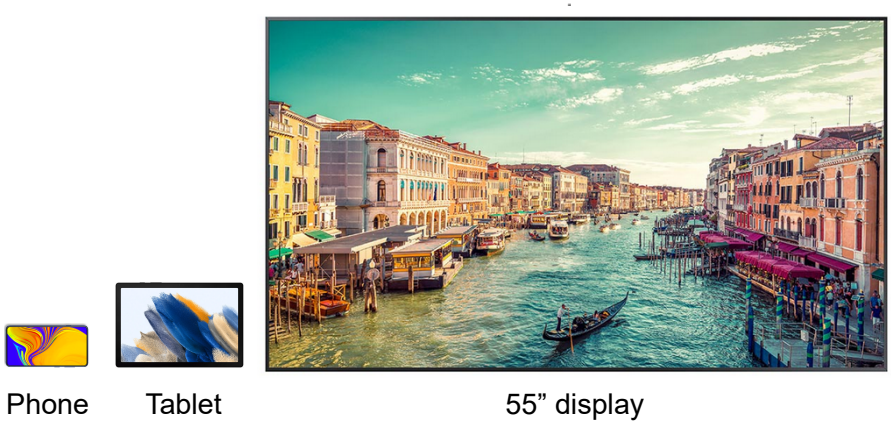


Figure 3.6: **Comparison of screens with different dimensionality, shown to scale, that could potentially hide a USC.**

Chapter 4

Detection Model

In this chapter, a detection model will be proposed that aims to solve the challenges that have been explained in the feasibility study of Chapter 3 when using optical reflections to detect USCs. First, the core principle behind the detection model will be explained, and second a methodology will be given that will integrate this core principle into a detection strategy.

4.1 Detection Principle

As stated in Section 3.2, one of the main challenges that makes detecting the presence of a USC difficult, is the reflectivity of the screen. When using a detection device such as ToF sensor, the reflectivity of the screen firstly introduces noisy reflections and secondly interferes with the retro reflectivity of the USC which limits its FoV. In order to combat these challenges, a novel detection principle is proposed that will rely on the difference in the reflective nature between the USC and the screen. In Figure 4.1 this principle is demonstrated. Let T be a trajectory of length L, on which an object will move from the start point *T(0)* to the end point *T(L)*. A ToF sensor will now travel this trajectory T and will move parallel to the screen containing the USC (for now, it will be assumed that while moving on this trajectory, the distance between the ToF and screen will be fixed). As explained in section 3.2, this ToF sensor will cause reflections in the screen and in the USC. The difference is that the USC reflections are retro reflections, while the screen reflections are specular reflections e.g. mirrored reflections. This means that as the ToF sensor moves along T, the position of retro reflections, relative to the screen, will stay the same, while the position of the mirrored reflections, relative to the screen, will change.

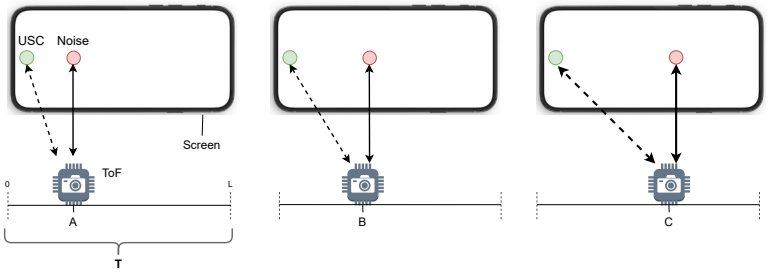


Figure 4.1: **While moving the ToF sensor along *T*, the position of USC reflections relative to the screen is static, while reflections from the screen itself are dynamic.**

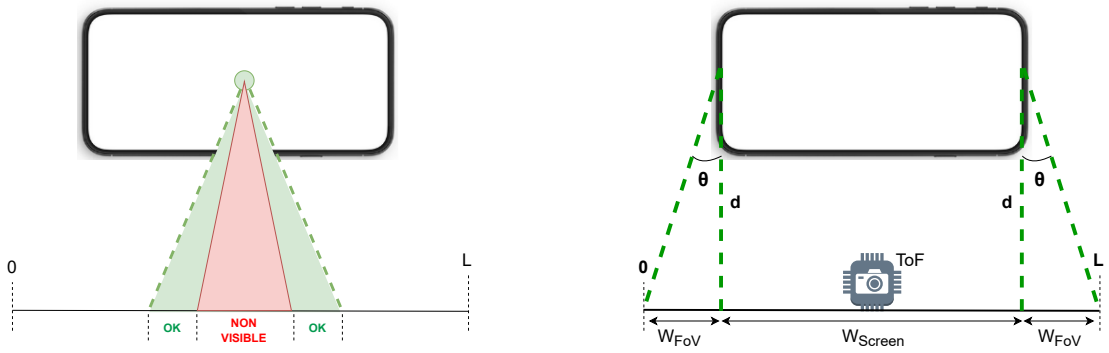
Let A, B and C be points on this trajectory which are in ascending order in terms of their distance on the trajectory, shown in Figure 4.1. Assume now that at these points, the reflections of the USC will be in the FoV of the ToF sensor. At *T(A)* the ToF sensor will cause USC reflections, shown with the **green** circle, and noisy screen reflections, shown with the **red** circle. When the ToF sensor moves along the trajectory to *T(B)* and to *T(C)*, the screen reflections will move accordingly to the movement of the ToF sensor, because they are mirrored reflections, while the USC reflection will be at the same location as it was at *T(A)*, because it is a retro reflection. Using this detection principle, it is possible to counter the challenges:

- While the ToF moves along T, reflections that continuously stay inside the same location of the screen must be caused by a retro-reflective object, which strongly suggests the presence of a camera e.g. the USC. Also, while moving along the trajectory, the ToF sensor will fully capture the FoV of the USC as shown in Figure 4.2a, allowing the visible parts of the FoV to be captured. Therefore, there will be a section on the trajectory (marked with **OK** in the Figure) for which the ToF sensor will be able to see all possible reflections from the USC. This section will be referred to as the USC's *visible section*.

- Because one of the challenges is that it's not known where the USC is located, the detection method needs to scan the whole screen. The left and right edges of the screen are the boundaries that determine the horizontal range in which USC is located. The length of the trajectory should therefore at minimum such that it allows the ToF sensor to scan the FoV of these edges. This is shown in Figure 4.2b, where θ refers to $\theta_{usc-side}$ (see Section 3.2.1). The minimum trajectory length L_{min} is given by equation 4.1:

$$\begin{aligned} L_{min} &= W_{Screen} + 2 \cdot W_{FoV} \\ &= W_{Screen} + 2 \cdot d \cdot \tan\theta \end{aligned} \tag{4.1}$$

  where W_{Screen} denotes the screen's width and W_{FoV} denotes the width of the side margins that are needed to scan the whole screen.



(a) (b)

Figure 4.2: **While moving on this trajectory, the ToF will be able to capture the visible parts of the USC's FoV as shown in *(a)*. The trajectory length, *L*, should be chosen in such a way that at minimum it is able to scan the whole screen for retro reflections as shown in *(b)***

.

4.2 Methodology

4.2.1 Reflection Grid

As explained in the previous section, the core principle behind the detection method is that while moving the ToF sensor along the trajectory, USC reflections will stay inside the same location of the screen. In order to detect this behavior, a *reflection grid* is proposed which will track the position of reflections. In Figure 4.3 this grid is shown, where the contours of the screen are used as contours for the grid. Because almost all screens have a rectangular shape, the grid will also be rectangular, although the principle can also be applied to different shapes. The grid will be an $m \times n$ matrix, called M_{RG}, where m is the amount of rows and n is the amount of columns. It will contain $m \cdot n$ boxes, and the notation for each box is $box_{i,j}$ where i is the row and j is the column of the box.

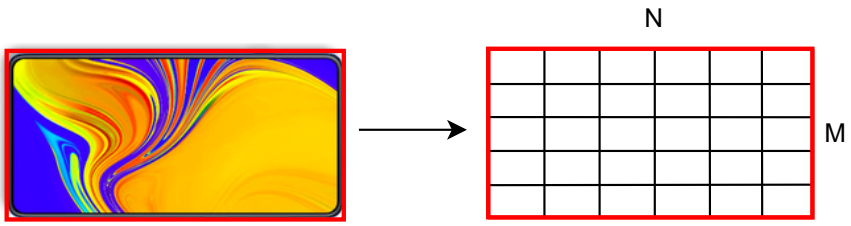


Figure 4.3: **Mapping of screen to the *reflection grid***

For each frame that the ToF sensor will capture while moving along the trajectory, a reflection grid of the screen will be detected and all reflections found within the boundaries of this reflection grid, will be stored inside the grid's corresponding. After detecting and storing these reflections, M_{RG} is described by equation 4.2. In Figure 4.4 this is demonstrated. Note that in this manner, the search space for reflections is automatically limited to the screen, which is the goal of this research, because it is focused on detecting cameras under the screen. As it becomes clear from this example, reflections that are in the same location in the screen will be mapped to the same box in the grid ($M_{RG}(2,0)$ in this example).

$$M_{RG}(i,j) = \begin{cases} 1, & \text{if reflections found in } box_{i,j} \\ 0, & \text{otherwise} \end{cases} \tag{4.2}$$

The size of the resulting boxes in the reflection grid, is one on the hyperparameters of the detection model. A large box size can allow margins in the location errors for reflections from the USC but also increases the undesired possibility that reflections from the screen are located in the same box.

4.2.2 Confidence Score

The reflection grid will now be used to construct a *confidence score*, an estimation about how likely the chance is that the scanned screen will contain a USC. The basis for this confidence score will be two observations that were made in Section 4.1:

- The behavior that defines the presence of a USC, is that its reflections stay inside the same location in the screen because of the retro-reflective effect. These reflections should be found inside the same box in the *reflection grid*.

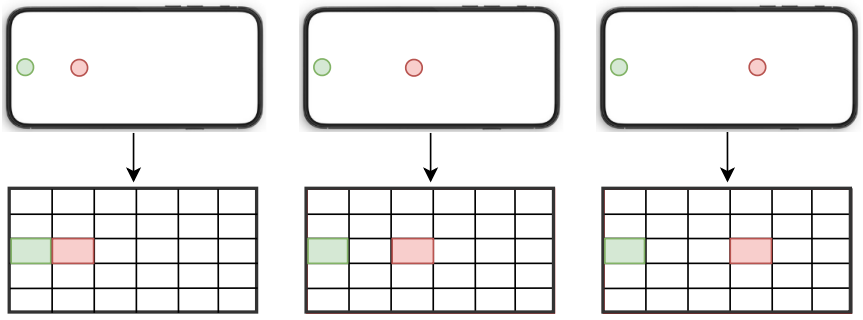


Figure 4.4: **Example of how reflections are mapped on the reflection grid.**

- The trajectory T, with a carefully chosen length L, will always contain a *visible section* where the ToF sensor will be able to fully capture the USC's reflections because the sensor will be inside the USC's FoV.

Combining these two observations, it can be derived that for the set of captured frames inside the USC's *visible section*, the corresponding reflection grids should have one box for which the amount of found reflections should be high, because this would correspond with the behavior of a retro-reflective object. In Figure 4.5 this concept is shown: the visible section consists in this example of three frames (indicated with n), and there is one box, $box_{2,0}$, that contains a reflection in every frame.

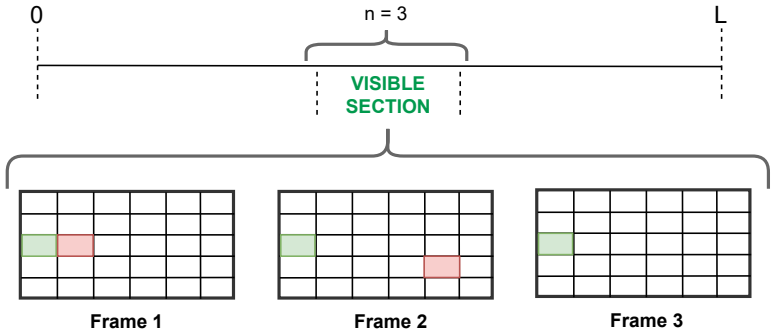


Figure 4.5: **Example showing a possible set of reflection grids in the visible section.**

The set of consecutive frames that corresponds to the visible section, is on beforehand not known, because the location of the USC is unknown. Therefore, while capturing frames on the trajectory, every n consecutive frames should be taken into account for detection, where S ideally is the number of frames in which the visible section is section. To take this into account, a *sliding window* approach will be used to turn the recorded frames into windows of S consecutive frames. The concept of the **confidence score** can now be introduced. This confidence score will be calculated for every window. A window consists of S frames for which every frame has a reflection grid. These reflection grids will be combined into a new grid, which is called the **confidence grid** M_{CG}, where each box value will be the average of all the box value from all reflection grids inside the window. For this confidence grid, the confidence of each box in the grid will be determined by equation 4.3.

$$M_{CG}(i,j) = \frac{\sum_{n=1}^{S} M_{RG}(i,j,n)}{S}$$
$$\text{where } M_{RG}(i,j,n) \text{ maps to } M_{RG}(i,j) \text{ of frame } n \tag{4.3}$$

Ideally, if every reflection can be detected, there should be a box in the confidence grid which has a confidence level of 1.0, the highest possible confidence. In Figure 4.6 the resulting confidence grid M_{CG} is shown, for the example shown in Figure 4.5.

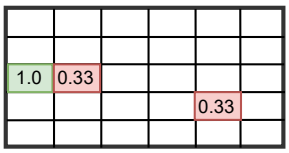


Figure 4.6: **The resulting confidence grid, for the example shown in Figure 4.5.**

4.2.3 Ideal Window Size

As explained in the previous section, a sliding window of size S will divide the video stream into windows and for each window, a confidence grid M_{CG} will be computed. The value chosen for S is important, because it determines the amount of frames that will contribute to the confidence score. Ideally this value should match with the number of frames that the ToF sensor will capture while being in the *visible section*, because this means that the window would exactly capture all reflections of the USC. If this ideal value could be found then there will be a window for which its confidence grid contains 1.0-confidence box, assuming that all reflections are accurately detected and localized. This section will provide some theoretical reasoning on how the ideal value of the window size S can be computed, and provides some practical recommendations for setting S.

Recall from Section 3.2.1 that the FoV of the USC is partially blocked by the interference of the screen's reflections. Because of this interference, reflections are only visible in one of the two side lobes of the FoV. Ideally, the window size should therefore match with the frames captured in one of these side lobes. In Figure 4.7 the situation is shown for one of the lobs with the following assumptions:

- The ToF sensor moves on the trajectory with a constant velocity of v_{sensor} and the target distance d is constant during the trajectory. The frame rate of the sensor is constant and given by FPS_{sensor}.
- The normal FoV in this lob (e.g. without the presence of a screen) has an angle of θ_N and ranges from x_c and x_n on the trajectory.
- The blocked FoV (due to the screen) is centered around the 0° angle of incidence, and in this lob reduces the USC's FoV by an angle of θ_B. On the trajectory this is between x_c and x_b.

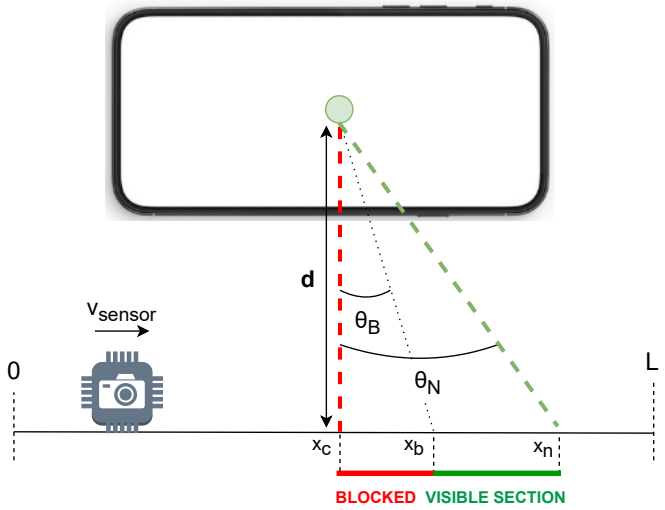


Figure 4.7: **Mathematical model used to derive the ideal window size that corresponds to the visible section**

The frames that correspond to the *visible section* are all frames captured while the ToF sensor moves between x_b and x_n. It is assumed that the ToF sensor is moving at a constant speed during these frames. Using the model from Figure 4.7 the ideal window size S_{ideal} can therefore be derived using equation 4.4:

$$\begin{aligned} S_{ideal} &= \text{frames captured between } x_b \text{ and } x_n \\ &= FPS_{sensor} \cdot (t(x_n) - t(x_b)) \\ &= FPS_{sensor} \cdot \frac{x_n - x_b}{v_{sensor}} \\ &= FPS_{sensor} \cdot \frac{(x_n - x_c) - (x_b - x_c)}{v_{sensor}} \\ &= FPS_{sensor} \cdot \frac{d \cdot \tan\theta_N - d \cdot \tan\theta_B}{v_{sensor}} \\ &= \frac{FPS_{sensor} \cdot d}{v_{sensor}} \cdot (\tan\theta_N - \tan\theta_B) \end{aligned} \tag{4.4}$$

where $t(x_n)$ and $t(x_b)$ are respectively the time instances when the ToF sensor is at x_n and x_b. It can be seen that S_{ideal} depends on a deterministic part: $\frac{FPS_{sensor} \cdot d}{v_{sensor}}$ and a non-deterministic part: $(\tan\theta_N - \tan\theta_B)$. As already discussed in Section 3.2.1, it is challenging to precisely determine θ_N and θ_B. Therefore, a more practical approach is to let these values be based on empirical analysis and research:

- The angle of the normal FoV θ_N, for cameras that are similar in dimensionality to the USC, has been the subject of several works in the field of optical detection (see section 2.2.2). In the case of mini-cameras the maximum FoV is 20° [14] and the work of LAPD confirms this observation.
- The angle of the blocking FoV θ_B can be estimated based upon its reflection profile of that

is visible in the screen. In Figure 4.8 this is shown. Because θ_B relates to the angle of the blocking FoV for a single lobe, the FoV of the reflection profile is twice as large as θ_B.

While the window size can be manually set, the theoretical model of S_{ideal} should be used as a guidance for practical implementations of the detection model. When doing so, it is however important to keep in mind that the theoretical model assumes that the ToF sensor moves with constant velocity on the trajectory, while in practice it cannot be assumed that the velocity is constant.

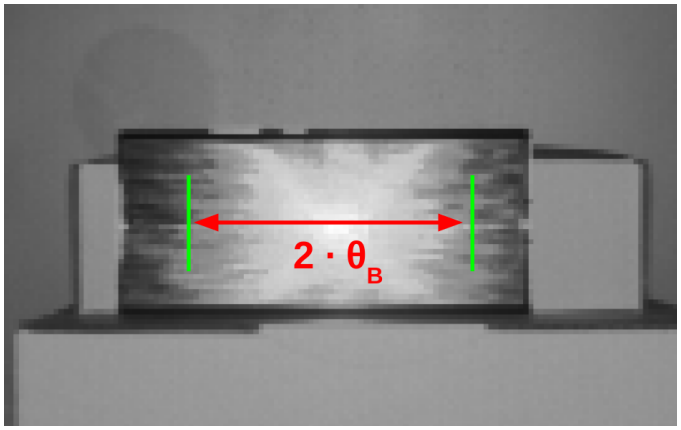


Figure 4.8: θ_B **can be estimated using the width of its reflection profile**

Chapter 5

System Architecture

This chapter will provide an architectural overview and system design of a detection system based upon the principles of the detection model which has been proposed in Chapter 4. The core components and their requirements will be determined that together will make up an architecture that can be used to detect USCs. Furthermore, an implementation of the architecture will be given which will be used to perform an evaluation on the effectiveness of the detection system, as will be shown in Chapter 7.

5.1 Overview

In Figure 5.1 an overview of the system architecture can be seen. The system consists of three modules:

1. **Screen Scanning Module**: The first module is responsible for scanning the screen and aggregating video data from the ToF sensor during this scan process. In this architecture, the scanning process needs to be finished before processing and predicting can be done.
2. **Reflection Grid Detection**: Every frame from the video stream of the first module will be forwarded to the second module, which will extract a reflection grid from this frame. This will be a two-stage process, the first stage being *screen detection* and the second stage being *reflection detection*.
3. **Decision Module**: The third module will predict based upon the methodology explained in Chapter 4, if the screen contains a USC, using the video stream and the computed reflection grids. It will do this using a sliding window approach. The output will be binary, where a **0** corresponds to *No USC* and a **1** to *USC detected*.

5.2 Screen Scanning Module

Image streaming will be done using a ToF sensor. The ToF sensor will record data with a constant frame rate while traversing the trajectory. In Figure 5.2 an outline of the measurement trajectory is shown. The distance d between the ToF sensor and the target screen will be fixed while moving along the trajectory. Because this distance will impact the strength of the retro reflections [14], as well as the reflectivity of the screen, it will be part of the preliminary evaluation of the detection model to determine the operating range of target distances for the

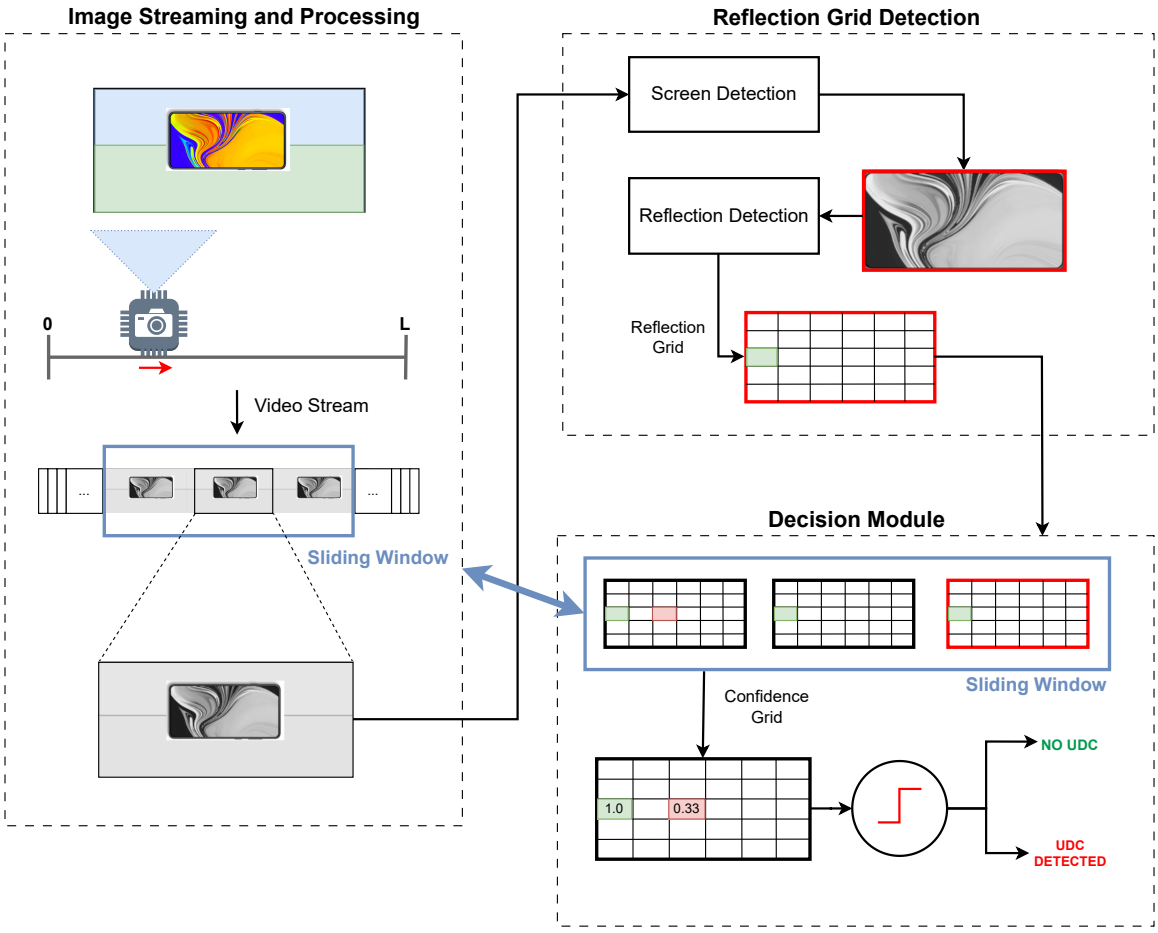


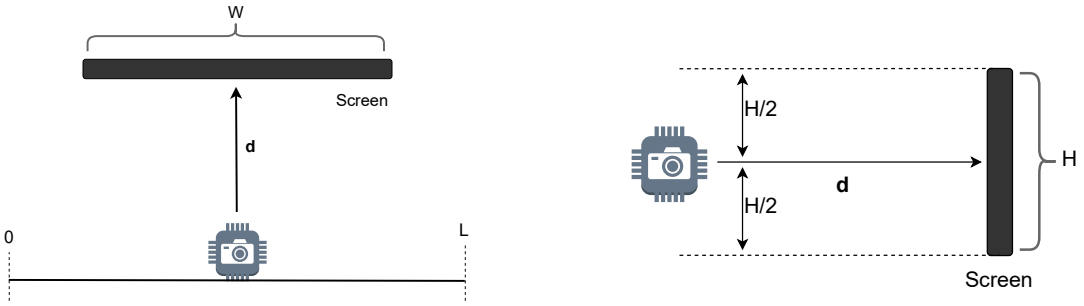
Figure 5.1: **A complete overview of the proposed system architecture used for USC detection.**

detection system. As explained in Chapter 4, the minimum trajectory length L is dependent on the target distance and the screen width, so this needs to be determined before starting the scanning process. During the scanning process, the ToF sensor's height will be fixed at the vertical center of the screen, shown in Figure 5.2b.

5.3 Reflection Grid Detection

A critical part of the effectiveness of the detection system will be the accuracy of the reflection grid detection module. This module will be responsible for computing the reflection grid for each frame and will operate using the following steps:

1. From the amplitude image captured by the ToF sensor, the mask of the screen will be detected. As stated in the previous chapter, this research will assume that the screen will have a rectangular shape. Therefore, detecting the four edges of the screen will give enough information to compute the mask of the screen.



(a) **Top view of the scan process** (b) **Side view of the scan process**

Figure 5.2: **Outline of the distances assumed in the scan process**

2. Within the boundaries of the mask of the screen, reflections will be detected.

3. Based upon the mask of the screen, an $m \times n$ grid will be generated. The chosen dimensions for the grid, will be a hyperparameter of the detection system.

4. The detected reflections will be placed inside the grid. After placing the reflections in the grid, every box inside the grid will have a binary state, where **0** means no reflections found inside box, and **1** means reflections found inside the box.

Because the principles behind this method already have been explained in chapter 4, the following sections will give a detailed explanation on the implementation of the screen detection and reflection detection parts.

5.3.1 Screen Detection

The problem of screen detection can be put in the category of Semantic Image Segmentation (SIS). This is a well-known computer vision problem where the task is to identify the class of each pixel in an image. Due to major improvements in the field of Deep Neural Networks (DNN), the field of SIS has experienced huge breakthroughs over the past decade and should therefore be categorized into two type of methods: traditional or classical methods (e.g. before DNN) and DNN-based methods [30], [31]. Traditional methods usually rely on image features such as edges, colors or histograms. They can be used in a supervised manner, such as thresholding, which is widely used in medical applications, or they can be used unsupervised, such as k-means clustering. With the rise of DNNs, several limitations that are fundamentally present in traditional SIS methods are addressed. Being able to learn complex patterns in images, DNN methods achieve higher accuracy and can better deal with novelty [32], [33].

There are several reasons why a DNN based method will be used for screen detection. First, performance will be important factor in the effectiveness of the detection system. Performance in terms of screen detection should be considered by the following two quantitative metrics:

- *Detection rate*: the percentage of frames for which a screen can be detected. If a screen can not be detected, the reflection grid cannot be constructed, which will impact the reliability and accuracy of the confidence grid and the prediction made in the decision module (see Section 5.4).

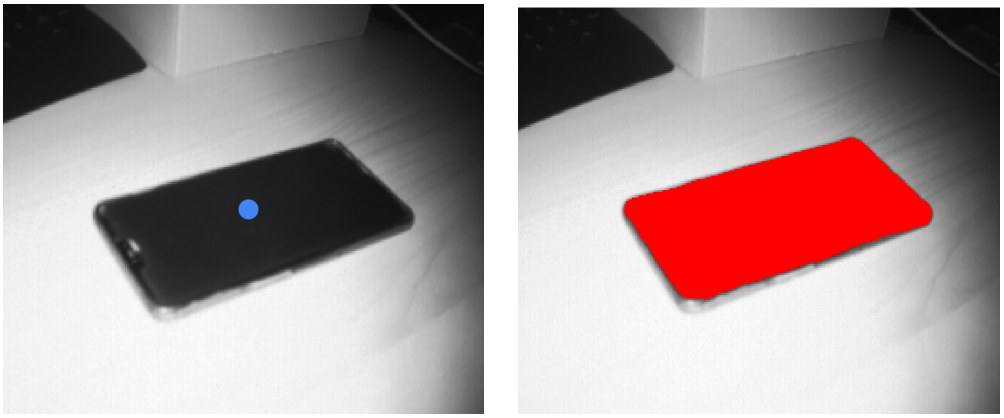
- *Prediction accuracy*: the accuracy of the mask and boundaries of the predicted screen. As the reflection grid will be based on an $m \times n$ matrix, the detected screen needs to be

mapped to the dimensions of this grid, such that reflections can be localized inside the grid. Therefore, errors in the accuracy of the screen mask will result in deviations in the localization of reflections.

Second, there is a large variation in the image data that will be captured by the ToF sensor. Screens have different dimensions, can be located or installed in a variety of locations, and will be recorded by the ToF sensor under a variety of angles or distances. DNNs are better resilient against these dynamic features of the images and also are more suitable to be updated to learn new features.

Segment Anything Model

Segment Anything Model (SAM) is a novel segmentation model developed by Meta AI Research [34] in 2023. The model is part of the Segment Anything project which goal is to build a generalized foundation model for image segmentation. It is a promptable model based upon zero-shot learning (ZSL), meaning the model can be used to predict masks of objects belonging to a class samples not observed during training. The model is trained on the largest segmentation dataset to date, containing 1 billion masks on 11 million images. The mask quality has an Intersection-over-Union (IoU) of more than 90% for 94% using a test dataset of 50k images. More interesting, experiments with human ratings show that the masks are categorized as accurate and identifiable and *"... errors are small and rare (e.g., missing a small, heavily obscured disconnected component, ...)"*. In this research, SAM is used in the implementation of the detection system, first to speed up the validation of the detection model because of its zero-shot capabilities and second because of its promising capabilities to provide screen masks for a diverse set of images. In Figure 5.3, an example is shown how a mask is generated from an ToF sensor image, using only one location prompt.



(a) (b)

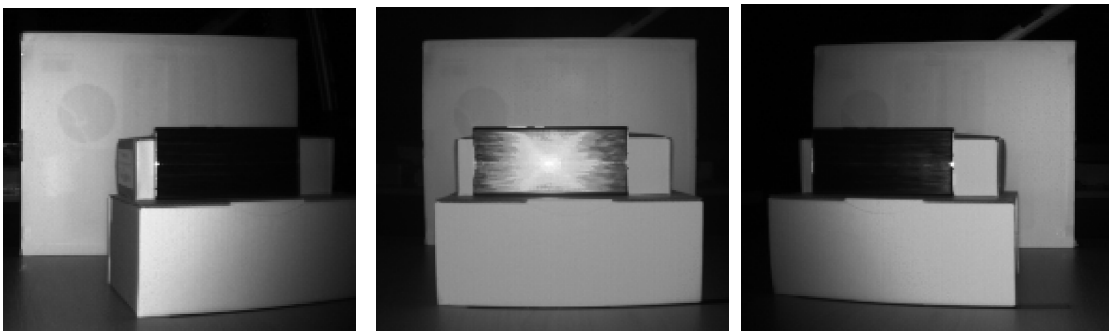
Figure 5.3: **Example of an image from the ToF sensor containing a screen *(a)* and the generated screen mask *(b)*, annotated with red using SAM. The location prompt is annotated with the blue circle shown in *(a)*.**

Because SAM needs a prompt in order to generate a mask from an image, an automatic prompting process is needed to prevent the user from manually providing prompts for every image the ToF sensor takes. This is done in the following simple manner: for the first image, ask

the user to select the location of the screen. For all subsequent images, use as location prompt the center of the detected grid from the previous image.

5.3.2 Screen to Grid Mapping

From the segmentation mask, a grid needs to be constructed. The grid is an $m \times n$ matrix, and is a based upon the screen being scanned e.g. the aspect ratio of the screen, should match with the aspect ratio of the grid. There is however a challenge in mapping the detected screen mask to this grid. Because images are taken from a variety of angles on the scanning trajectory, the shape of the detected screen mask is impacted by optical distortion and will deviate from the grid's rectangular structure. This is illustrated in Figure 5.4 where images from several angles can be seen when moving on the trajectory.



(a) (b) (c)

Figure 5.4: **Several images captured on the measurement trajectory from different angles which shows how the shape of the screen is impacted by optical distortion.**

The principle used to solve this challenge is based upon a technique called *homography decomposition* [35], [36]. This popular technique used in the field of computer-vision is used to compute a Homography Transformation Matrix, M_H, which describes the spatial relation between two images or views of the same object. The homography matrix can be calculated using four correspondence points of the object found in both images, the four edges of the screen in this case. The computation of M_H will be therefore be done using the following two steps, shown in Figure 5.5:

1. Detect the four corners of the screen mask (marked with yellow), which should correspond to the four corners of the grid rectangle.

2. Map the four corners of the screen to the four corners of the grid and compute H_M using the four corners or the screen and the four corners of the grid.

In order to compute M_H from the detected screen, the grid needs to be known. Because in this proposed architecture, processing is done after the whole video stream is recorded, the image with the least distorted screen will be used to determine the aspect ratio of the grid. This is when the center of the screen is exactly the center of the captured image e.g. when the observation angle between the ToF sensor and the center of the screen is 0° (such as in Figure 5.2a or Figure 5.4b). The width and height of the detected mask from this center image will be used to determine the dimensions of the grid.

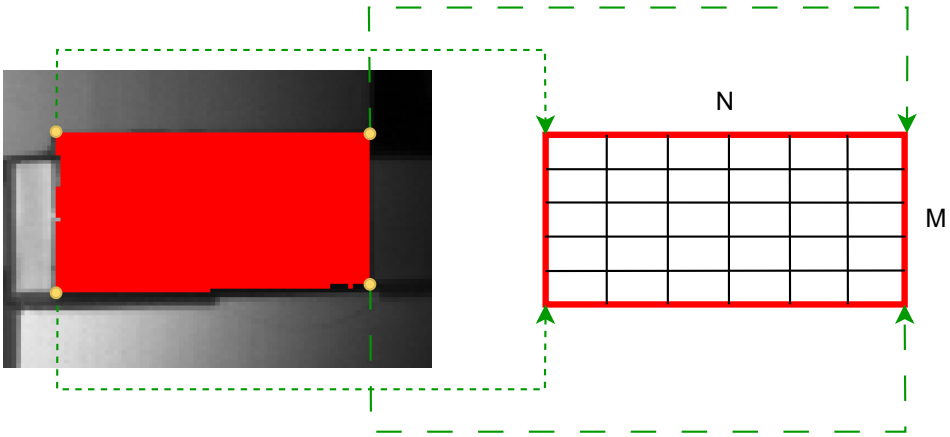


Figure 5.5: **In order to map the detected screen mask to the grid, the four corners of the mask should be detected and mapped to the grid's corners.**

5.3.3 Reflection Detection

The goal of the reflection detection module is to localize retro reflections inside the detected screen. Recall from Chapter 2 that retro reflections from a camera are high-intensity blobs. In order to detect these reflections, a combination of filters is applied to the image. In Figure 5.6 the stages used in this filtering process are shown. The following sections will describe the purpose of each filter stage.

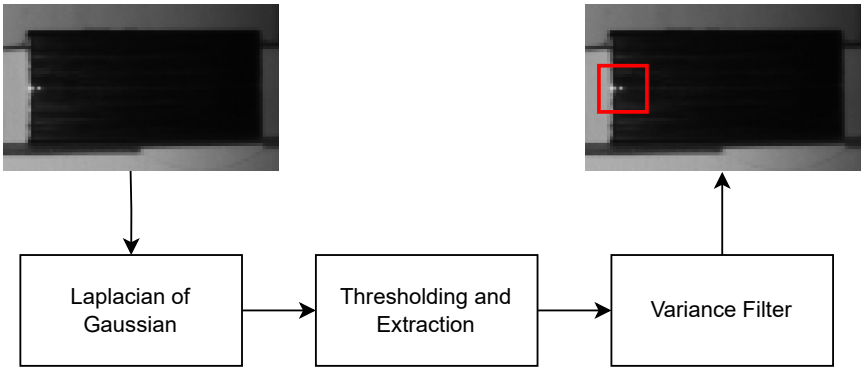


Figure 5.6: **The filter stages used to detect reflections inside the screen.**

Laplacian-of-Gaussian Filter

The first filter, the Laplacian-of-Gaussian (LoG), is used to transform the image such that high-intensity blobs will be highlighted. The LoG filter is a well known method to identify blob structures in images[37] and has many applications in the field of medical imaging [38]. The LoG

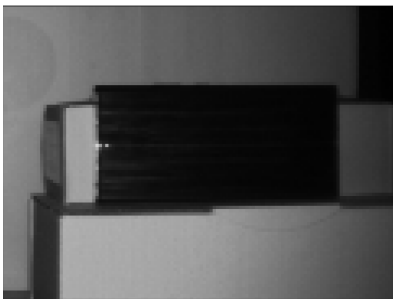
function is defined as:

$$LoG(x,y) = \nabla^2 G(x,y;\sigma) = \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}\right) G(x,y;\sigma)$$

where $G(x,y;\sigma)$ is the Gaussian function, with σ as standard deviation, and is defined as:

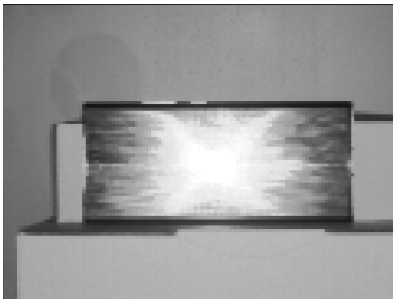
$$G(x,y;\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} * exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

The effect of the ∇^2 operator is to highlight regions of rapid intensity change and the Gaussian function is used to smoothen the image and attenuate noise. In Figure 5.7 the effect of convolving this filter with amplitude images from the ToF sensor can be seen. In Figure 5.7b the retro reflection is highlighted with a strong white blob. Furthermore, in Figure 5.7d, the attenuation effect of the LoG function on large high-intensity regions can be seen, such as the screen reflections. Because these large regions don't have the rapid change in intensity compared to small high-intensity blobs, they will not be highlighted in the convoluted image.

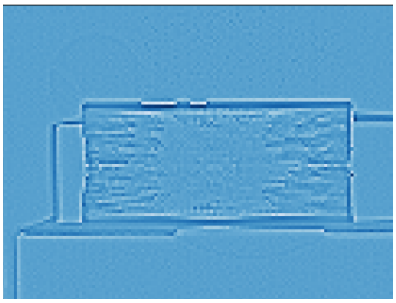


(a) **Image with retro reflection**

(b) **Image in (a) LoG filtered**



(c) **Image with screen reflections**



(d) **Image in (c) LoG filtered**

Figure 5.7: **Effect of LoG filter on screen reflections**

Thresholding and Extraction

The goal of this filter is to extract the high-intensity blobs from the LoG filtered image $I_{log}(x,y)$. First, a threshold is applied which will produce a binarized image $I_{bin}(x,y)$ in which **1**s represent

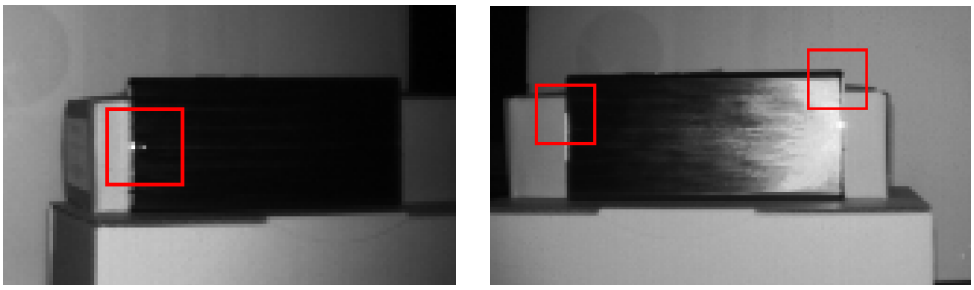
highlighted blobs and **0**s represents regions without highlighted blobs. This is done as following:

$$I_{bin}(x,y) = \begin{cases} 0 \text{ if } I_{log}(x,y) \leq T_{bin} \\ 1 \text{ if } I_{log}(x,y) > T_{bin} \end{cases}$$

The binarization threshold T_{bin} is tunable parameter that allows the filter to be sensitive to the intensity of the blob. After the thresholding, a list of high-intensity blob locations can de derived by extracting all **1s** from the binarized image $I_{bin}(x,y)$.

Variance Filter

While the high-intensity blob list derived from the previous filter, contains retro reflections, it also contains a lot of reflections that have an intensity distribution which does not correspond with intensity distributions expected from camera retro reflections (see Chapter 2). In Figure 5.8 some reflections are shown that are extracted from the image using the previous two filters. The reflection in Figure 5.8a matches with the intensity distribution of a camera retro reflections, while the reflections in Figure 5.8b certainly do not correspond with this distribution. These reflections are typically caused by non-uniformities in the screen such as its edges. The goal of the variance filter is to refine the list of detected reflections in order to decrease the amount of false positives.



(a) (b)

Figure 5.8: **Detected reflections shown in (a) match the distribution of retro reflections while detected reflections such as in (b) do no match, and cause false positives in the reflection grid, if not filtered.**

For each detected reflection, the variance filter will calculate the underlying standard deviation $\sigma_{neighbors}$ among the neighboring pixels. This variance if typically high for reflections from edges while it is low for reflections inside the screen. The variance filter operates using an odd-sized $n \times n$ kernel. The center element of this kernel corresponds to the location of the detected reflection. In Figure 5.9a a kernel of $n = 3$ is shown, where the reflection location is the centered green element. Because reflections can consume more space than 1 pixel in the ToF sensor image, the kernel allows the possibility to mark neighbors of the center element such that they will not be included in the calculation of $\sigma_{neighbors}$. In Figure 5.9b a kernel of $n = 5$ is shown, where the 8 adjacent cells of the center cell are marked to be excluded. More complex markings are possible, although simple kernels should be preferred to make the filter understandable.

After the calculation of $\sigma_{neighbors}$ the filter will determine based upon a threshold, called the variance threshold T_{var} if the reflection should be removed from the detected reflection list. Like T_{bin}, T_{var} is also a tunable parameter of the filter pipeline, and allows the filter to be less or more or less strict in filtering reflections based on their intensity distribution.

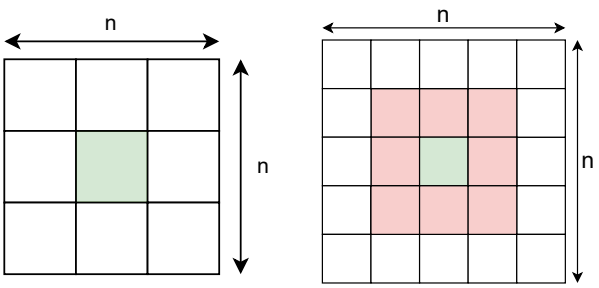


Figure 5.9: **Two examples of variance kernels used to calculate the standard deviation $\sigma_{neighbors}$ among the neighboring cells of the centered green cell. If cells are marked with red, they are not included in the calculation.**

Reflection Grid

After the reflections have been localized, the reflection grid M_{RG} of the frame can be computed. Recall from Section 5.3.2 that the transformation from the detected screen and the grid is described by the computed transformation matrix M_H. Therefore, the location of a reflection found in the screen, **p**, can be translated to a location inside the grid, **p'**, using the matrix multiplication shown in equation 5.1 [36].

$$\mathbf{p'} = M_H \mathbf{p} \tag{5.1}$$

In Figure 5.10 the mapping is visualized. Since the reflection grid is an $m \times n$ matrix, the location in the grid will be determined by the box that contains **p'**. It is possible that depending on the chosen dimensions of the reflection grid, multiple reflections are translated to the same box. The resulting matrix M_{RG} doesn't take this into account and only contains **1**s or **0**s, as described by equation 4.2.

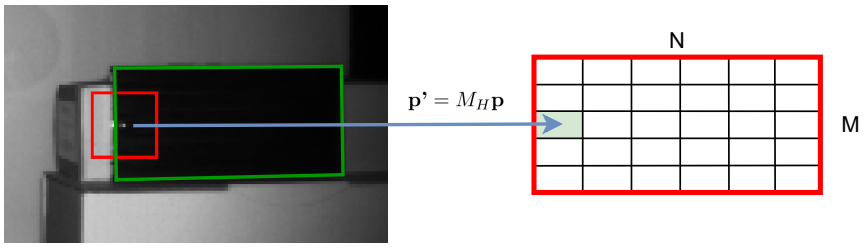


Figure 5.10: **Mapping localized reflections to the reflection grid using the transformation matrix M_H.**

When a screen cannot be detected by the screen detection module, the reflection grid can not be computed because it depends on M_H, which is based upon the detected screen. In this case, the reflection grid for the screen will be set the zero matrix $0_{m \times n}$ because the location of reflections can not be determined for this frame. This emphasizes the importance of an accurate and robust screen detection module.

5.4 Decision Module

After all frames are processed and a reflection grid for each frame is computed, the decision module will decide whether the screen contains a USC. In order to make this decision, the following four steps will be done, illustrated in Figure 5.11:

1. Using the predefined window size W_{SW}, divide the video into $N_{windows} = (N_{frames} - W_{SW})$ amount of windows where N_{frames} is the amount of frames captured on the measurement trajectory and $N_{windows}$ is the amount of resulting windows.

2. For each window, compute the confidence grid according to equation 4.3 using the computed reflection grids. This results in a list of confidence grids with length $N_{windows}$.

3. From all confidence grids, select the box with the highest confidence score. The confidence score of this box, is called the C_{found}.

4. Determine using a minimum confidence score threshold C_{min}, whether the confidence score C_{found} is high enough to be classified as USC. If $C_{found} > C_{min}$, the output of the decision module will be **1** else it will be **0**.

The value of C_{min} is important because it will determine the certainty of the detection system of its classification. In the evaluation, the value of C_{min} will be further examined.

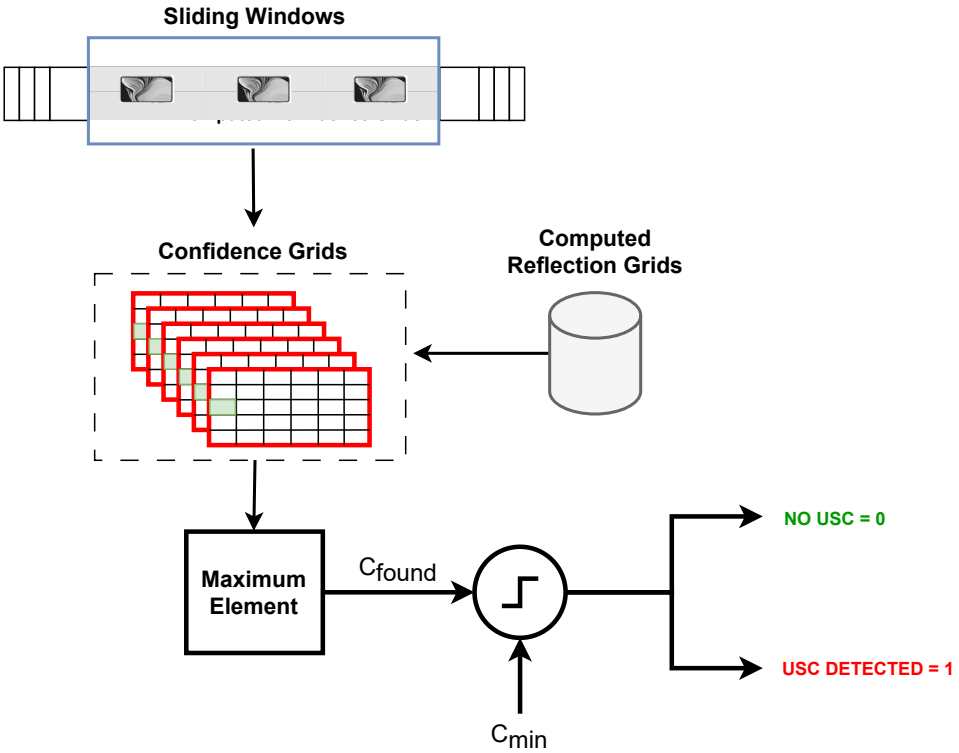


Figure 5.11: **Overview of the process of the decision module**

Chapter 6

Implementation

This chapter will give an overview of the implementation of a detection system that will be used to evaluate the detection architecture. The main goal of the implementation is to validate the detection model and principles from Chapter 4 and to assess the performance of the proposed detection architecture from Chapter 5. In Figure 6.1 an overview of the implementation can be seen, where two main components can be distinguished based on their functionality: **(1)** capturing images using a ToF sensor and **(2)** running the USC detection system using the ToF sensor data. The following sections will provide more details on the specific choices and considerations that were made during the design process of the implementation.

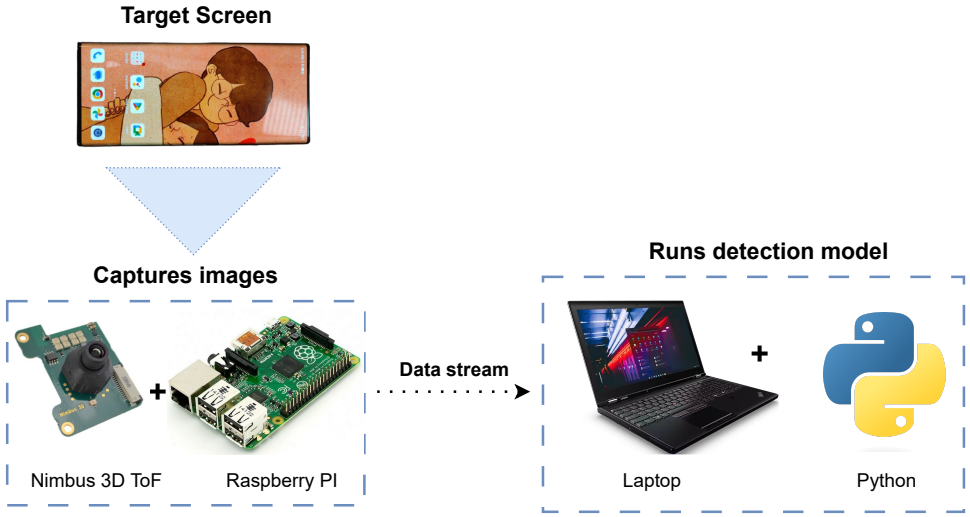


Figure 6.1: **Overview of the implementation used for evaluation of the detection system**

6.1 Target Screen

In Figure 7.8a an overview of different models of full-screen smartphones equipped with a USC can be seen that are used as target screens. In Table 6.1 an overview of the technical specifications

of the smartphones can be found. Note that the shape, size and layout of pixels in all smartphone screens is different and its specifications. A special Android APP is developed to control the displayed colors in the translucent screen region. The APP is used during the evaluation to assess the detection system's performance under varying screen content. The APP can run with Android 4.4.2 and above.

Smartphone model	**Screen Type**	**Screen Size**	**Density (PPI)**	**USC specs**
ZTE AXON 20 5G	OLED	6.92 inch	400	32 MP, f/2.0
ZTE AXON 30 5G	AMOLED	6.92 inch	400	16 MP, f/2.5
ZTE AXON 40 Ultra	AMOLED	6.8 inch	400	16 MP, f/2.0
Samsung Galaxy Z Fold3	AMOLED	7.6 inch	374	4 MP, f/1.8

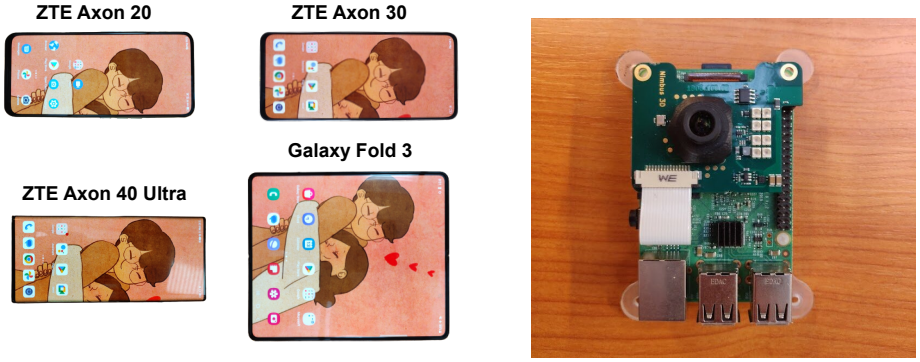
Table 6.1: **Technical specification of each used smartphone model**

6.2 Time-of-Flight Sensor

The ToF sensor is at the heart of the USC detection principle proposed in this work. From a high-level perspective it will provide a data stream from which the detection system needs to determine the presence of the USC. The ToF sensor market contains a large variety of devices, ranging from cheap and simple sensors that only provide one-dimensional distance data [39] to advanced and expensive equipment capable of performing high resolution accurate depth measurements [40]. Also, a lot of modern state-of-the art smartphones contain ToF sensors. The following considerations are taken into account before choosing the ToF sensor:

- It must be capable of capturing two-dimensional images of a moderate resolution such that reflections from 2-5 mm sized cameras are still detectable and tasks such as screen mask segmentation do not suffer large performance penalties due to low-resolution images.
- The ToF sensor should be easy to integrate into the detection system and its data interface must be convenient to use in order to increase development speed.

Based on these requirements, the Pieye Nimbus 3D ToF Camera Module [41] has been selected as feasible ToF sensor. The sensor, shown in (see Figure 6.2b), has a 352×288 resolution which is comparable to the resolution of ToF sensors found in modern smartphones. The camera's FoV is 66° × 54° (horizontal × vertical), which gives a horizontal pixel resolution of $\frac{2\cdot(d\cdot\tan 33°)}{352} \approx 3.69d \cdot 10^{-3}$ m/pixel where d is the distance in meters between the object corresponding to the pixel and the ToF sensor. For example, at $d = 0.6$, this gives a horizontal resolution of 2.21 mm per pixel. Note that this calculation did not take into account the errors introduced by the camera's lens distortion which causes radial distortion in the images [42]. The Nimbus ToF sensor is part of the Nimbus platform [43], an open-source platform which provides easy and accessible means to use the ToF sensor. The ToF sensor can be mounted on a Raspberry Pi, and using a set of preconfigured Linux kernel and modules, an interface to the Nimbus ToF sensor's is provided. It is possible to directly access the ToF sensor's data on the Raspberry Pi, however in this implementation, the *nimbus-server* interface is chosen, because the processing will not be done directly on the Raspberry Pi. The nimbus-server is a server that using web sockets provides an interface to access the ToF sensor's depth and amplitude images but also can be used to configure settings such as frame rate, exposure time and HDR settings. Important to note is that the communication with the web server on the Raspberry Pi needs to be done via Ethernet, because wireless connections cannot sustain the frame rate.



(a) **Smartphones used as target screens** (b) **The Nimbus ToF Camera**

Figure 6.2: **Implementation devices.**

6.2.1 Sensor Data

The python interface that communicates with the nimbus-server streams the sensor data of the ToF sensor. The sensor data contains the amplitude images, which is the data needed to run the detection system. Every amplitude image taken by the ToF sensor is a 352×288 matrix that contains grayscale pixel data. Every pixel in this matrix has a bit-depth of 16, which theoretically allows 65536 unique amplitude values. However, in practice the maximum values found are in the range of 5-6k which occurs when the imaging sensors of the ToF sensor saturate (which typically when observing retro reflections). Before the data is processed by the detection system, it is normalized and scaled down to a 352×288 array of 8 bit-depth. This is done to make the detection system less sensitive to the scale of the data and to make it usable for some later processing steps which require 8-bit depth data.

6.2.2 Sensor Configuration

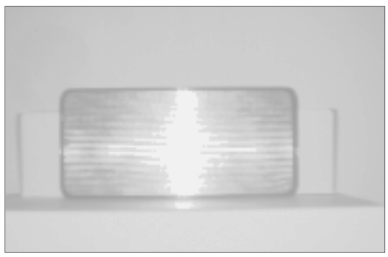
As mentioned earlier the nimbus server also provides an interface to configure several settings of the ToF sensor. The following configuration is implemented:

- *Use of HDR mode*: the ToF sensor supports the use of High Dynamic Range (HDR) technology. It captures several images of the scene using different shutter speeds and combines these images into a single image while balancing the resulting image's brightness based on the luminance distribution of the captured images. The impact of this setting on the quality of the images has been examined, from which can be concluded that the retro reflections of the USC camera are still visible with the same intensity while the intensity of reflections from other objects, such as the screen or background scenery becomes less dominant when HDR mode is turned on. When turned off (see Figure 6.3a), the reflections of the screen become more dominant and cause the screen's edges to be less visible which has a negative impact on the screen detection module. Based on these findings, HDR is turned on.

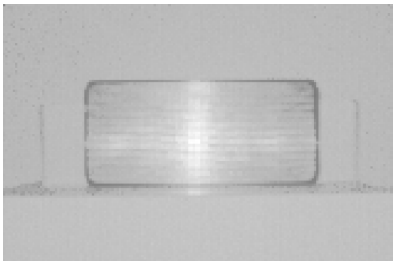
- *Exposure Time*: the exposure time controls the amount of time the image sensor cells will capture data during each frame. This value can be automatically controlled by the ToF sensor itself or can be manually adjusted. In Figure 6.3c the effect can be seen on the captured images when the exposure time is set too high. The cells become saturated and

the image quality drops. Also, the interference from the screen's reflection with the USC reflections becomes stronger. Therefore, the exposure time is set to relatively low values resulting in images such as shown in Figure 6.3d, to minimize the impact of the screen's reflections. Since the strength of the light received by the ToF sensor is distance-dependent, the exposure time is manually selected for different target distances between the ToF sensor and the target screen.

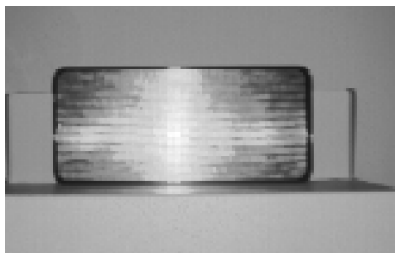
(a) **HDR mode turned off**



(b) **HDR mode turned on**



(c) **High exposure time**



(d) **Low exposure time**

Figure 6.3: **Figure depicts the output of ToF sensor with different configurations.**

6.3 Detection System

All data processing and execution of the decision system is done on a laptop. The laptop is connected via an Ethernet cable with the Raspberry Pi. A python module is used to interface with the ToF sensor [44] and the implementation of the decision module is also done using Python. This is chosen because of the powerful functionality and abstractions which Python (and its community) provides. Because the goal of this research is to investigate the detectability of USCs, the detection system will operate in an *offline* method e.g. video data will first be recorded and processing and evaluation of the video data will be done after the ToF sensor finished traversing the trajectory. However, in the evaluation of the system, the processing time will be examined in order to determine the feasibility of real-time processing and to provide insights into the time users would have to wait until the detection is finished.

Chapter 7

Evaluation

This chapter describes the evaluation of the implementation of the detection system as given in Chapter 6. To assess the performance of the detection system, some clear metrics will be needed. Recall from 5.4 that the maximum found confidence score, C_{found}, for a measurement will be the box with the highest value from all computed confidence grids. The confidence score threshold C_{min} is needed in order to determine whether the maximum found confidence score is high enough to be classified as *USC found*. This value is important because it determines the reliability and usefulness of the detection system. During the evaluation this value will be set to 0.5, which is a common value to start with. However, later on in Section 7.2.3, a recommendation is given on suitable values for C_{min}. To evaluate the decision model, the following performance metrics will be used:

- *USC Detection*: the confidence score of the predicted location is higher than C_{min} and the location matches the actual location of the USC (also called True Positive). The *USC Detection Rate* relates to the percentage of USC Detections in a set of measurements and should ideally be 100%.
- *False Positive (FP)*: the confidence score of the predicted location is higher than C_{min}, but the location does not match with the actual location of the USC. The *FP Rate* relates to the percentage of false positives in a set of measurements and should ideally be 0%.

The evaluation will be two-fold: the preliminary evaluation and the robustness evaluation.

1. **Preliminary evaluation**: The detection system will be tested on its feasibility, and the optimization of some hyperparameters of the model will be done in order to determine their influence on the detection accuracy. First, an individual evaluation will be done on the screen detection module detection accuracy using the two defined metrics in Section 5.3.1. Since the accuracy of the screen detection module is important for the overall performance of the system, this will be an important part of the evaluation. Second, a preliminary evaluation will be done using a predefined test setup where test factors such as the USC model and environmental parameters such as ambient light or deployment scenario are fixed. The influence of the target distance will be examined for this specific implementation of the detection system and the effect of some hyperparameters such as the variance threshold T_{var} are discussed.
2. **Robustness evaluation**: Using the results of the preliminary evaluation, a robustness evaluation will be done on the overall robustness and performance of the detection system.

The model will be tested on several state-of-the-art USCs and a variety of deployment scenarios will be considered in order to verify the model's robustness. For the robustness evaluation, each measurement setup will be evaluated on detection accuracy using confusion matrices. An overall performance of the model is given based upon all measurements that have been done and a discussion on the influence of factors such as screen activity or screen protectors on the performance of the model is done.

7.1 Preliminary evaluation

7.1.1 Setup and procedure

For the preliminary evaluation, a predefined test setup is used in which the feasibility of the detection model is verified. In Figure 7.1 the test setup is shown that is used to perform the preliminary evaluation. The setup is placed on an office desk located in a bright office room. Because the target distance d is fixed for each measurement, a marking is placed on the table at a fixed distance d. The minimum trajectory length L_{min} is computed on beforehand using equation 4.1, where θ is set to 10°. For the target smartphone that is used in the preliminary experiments, the ZTE AXON 40 Ultra is chosen because at the moment of this work, it is among the smartphones containing the best hidden USC. In Table 7.1 an overview of the default parameters that were used during the measurements in the preliminary evaluation can be found. During the tests, several target distances will be used in order to see their influence on the USC's detectability. For each variation in the setup, e.g. different target distances, the measurement will be repeated **ten** times such that the influence of noise and measurement variances such as movement speed is minimized. Half of the ten measurements are done moving the ToF sensor from left to right on the trajectory at a nearly constant velocity, and the other half is done vice versa in order to make the results not depending on the trajectory direction.

Recall from equation 4.4 that the ideal window size depends on the trajectory speed of the ToF sensor. Because it is hard to move with constant speed repeatedly for each measurement, the trajectory speed will be calculated for each measurement such in order to compute the ideal window size. The average trajectory speed of the sensor can be approximated using the sensor's horizontal trajectory speed $v_{sensor,x}$ because the height of the ToF sensor and the target object will be fixed. $v_{sensor,x}$ can be estimated by tracking the movement of the center of the detected target screen. In the captured images, the center moves across the FoV of the image, a certain amount of pixels ($\Delta pixels$), and the amount of frames this takes ($\Delta frames$) can be used to determine the time of this movement. The computation is given in equation 7.1.

$$v_{sensor,x} = \frac{\Delta x}{\Delta t} = FPS_{sensor} \cdot \frac{\Delta x}{\Delta frames} = FPS_{sensor} \cdot \frac{\Delta pixels \cdot \alpha}{\Delta frames} \tag{7.1}$$

where α is the horizontal pixel resolution, which is $3.69d \cdot 10^{-3}$ (see Section 6.2).

θ_B	5°
θ_N	10°
FPS_{sensor}	18
T_{var}	20
Grid box size	5 pixels

Table 7.1: **Default parameters used during the preliminary evaluation.**

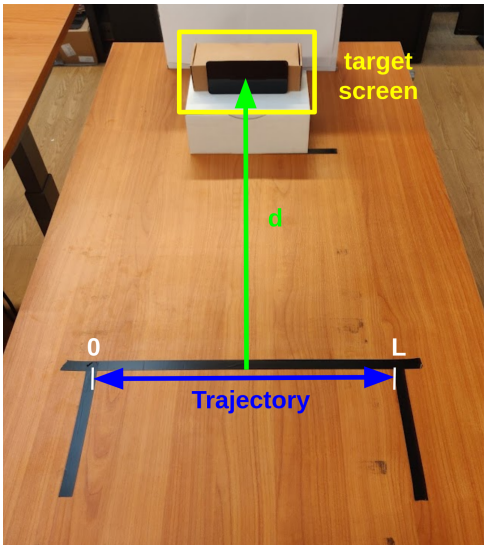


Figure 7.1: **The test setup *table*, used for the preliminary evaluation.**

7.1.2 Screen Detection Accuracy

As mentioned earlier, within the architecture of the USC detection system, the accuracy of the used screen detection model is crucial to the reliability and accuracy of the model's performance. In the implementation and evaluation of this work, SAM is used to perform screen detection. Although SAM is already thoroughly evaluated in the analysis and experiments section of the work[34], it is still important to consider its performance on the dataset produced by the ToF sensor used in this work. The screen detection accuracy is evaluated on its prediction accuracy. **Prediction accuracy**: the accuracy of the predicted screen compared to the ground truth of the screen. For this metric, the Intersection over Union (IoU) is used, which is defined as: $IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$. The Area of Overlap is computed using the intersection of the pixels of the predicted screen and the pixels of the ground truth screen, while the Area of Union is calculated using the sum of pixels of the predicted screen and the ground truth screen (see Figure 7.2 for an illustration of these areas).

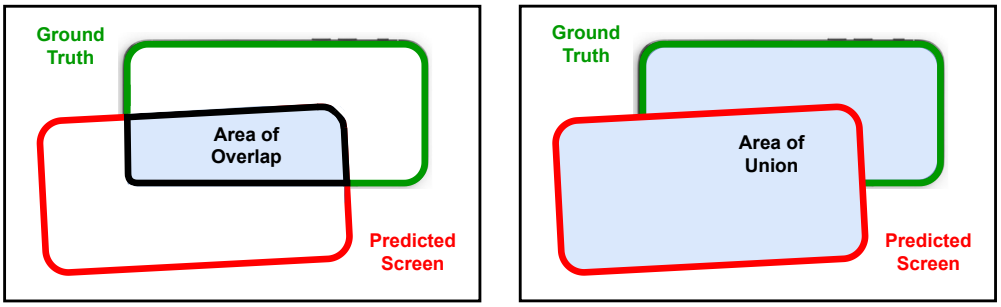


Figure 7.2: **The Intersection of Union (IoU) is used to assess the performance of the screen detection module.**

A dataset of 100 images captured by the Nimbus 3D ToF sensor has been created. The images

in the dataset have been captured with the following four varying factors: (1) screens with or without USCs, (2) ambient conditions, (3) deployment environments and objects surrounding the screen, and (4) observing angle and distance between ToF sensor and screen. For this dataset, the average IoU is 0.92 which is a very solid performance

7.1.3 Impact of Target Distance

The performance of the decision model is critically evaluated across various target distances. The target distance significantly influences several components of the detection system, including (1) the strength and profile of the screen's reflections, (2) the precision of the screen detection module, and (3) the length of the trajectory, which consequently affects the measurement time. The evaluated target distances span from 30 to 90 cm, chosen to reflect realistic distances that users of the detection system might encounter.

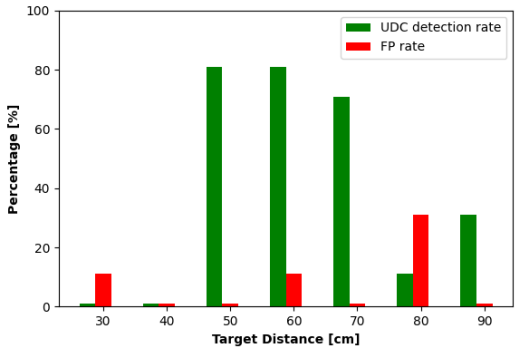
In Figure 7.3, we present the results of this evaluation. Figure 7.3a shows the USC detection and FP rates, while Figure 7.3b presents box plots for the maximum computed confidence scores. Based on these results, we draw the following conclusions regarding the impact of the target distance:

- The distances in the range of 50 to 70 cm perform wells in terms of USC detection rate (roughly 77% for 30 samples) and FP rate (only one false positive is marked for $d = 60$ cm, which is caused by misalignment of the reflections). The maximum confidence scores for these distances are also in the upper range (means>0.6).
- The detection system struggles to perform well for the *short-ranged* distances, 30 and 40 cm. The maximum confidence scores are too low (0.36 [30 cm] and 0.17 [40 cm] on average), to provide a reliable detection result and further investigation shows that the detected reflections are mostly caused by noise of the screen reflections. Note that this however results in only one false positive ($d = 30$ cm), meaning that the detection principle e.g. the sliding windowing method eliminates most false positives caused by spurious reflections from the screen.
- For long distances (80 and 90 cm) the detection system has difficulty to detect all reflections from the USC. The inverse-square law effect of the distance causes the reflections to be not strong enough to pass the reflection filter. Some further investigation showed that the confidence scores increase T_{var} is set to a less strict value, but this also increases the FP rate.

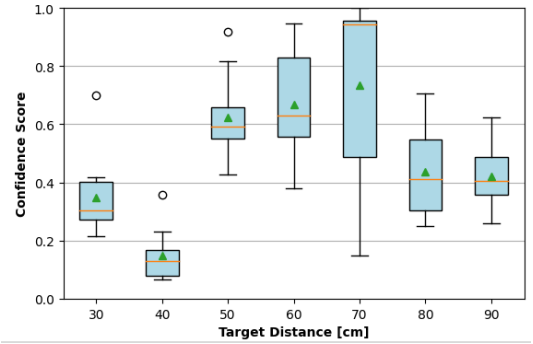
7.1.4 Ablation Study

Variance Threshold

Recall from Section 5.3.3 that while the variance filter provides means to control the aggressiveness of the filter to remove reflections for which the intensity distribution do not resemble the distribution of USC retro reflections. The variance threshold T_{var} is used for this purpose, where low values of the threshold will cause the filter to be very strict while large values will make the filter more relaxed. Because this is an important hyperparameter of the detection system, its influence of the performance is examined. In Figure 7.3 the results are shown for varying the variance threshold T_{var} from 10 to 80 using steps of 2. In Figure 7.4a the USC detection rate and FP rate is shown while in Figure 7.4b the mean values of the maximum confidence scores for all 10 measurements at $d = 60$ cm are shown. From these results, the following conclusions are drawn on the value of T_{var}:



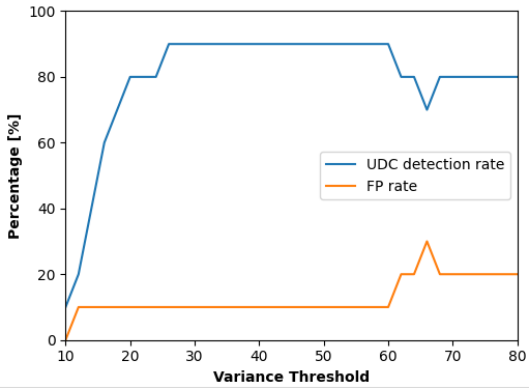
(a) **USC detection and FP rate**



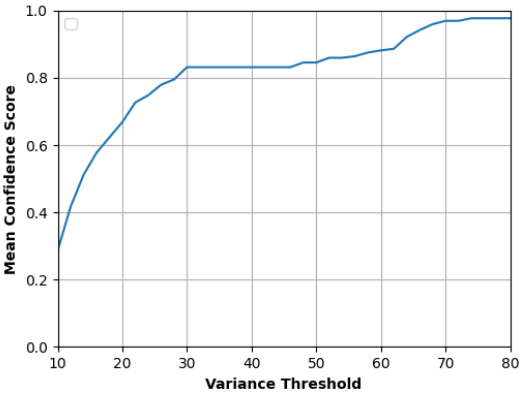
(b) **Maximum confidence score.**

Figure 7.3: **Varying target distances and the corresponding results using $C_{min} = 0.5$.**

- For $T_{var} < 20$, the USC detection rate starts to drop because the filter becomes to strict. The effect of the filter becomes stable around $T_{var} = 30$, where the maximum USC detection rate and lowest FP rate can be found.
- Increasing T_{var} also increases the mean confidence score as can be seen in Figure 7.4b. This makes sense because it allows more reflections to be detected. It is however no guarantee for a performance improvement because after $T_{var} = 30$, the USC detection rate does not increase while the FP rate increases (even at the expense of true positives). It is therefore important to find the cutoff point (such as $T_{var} = 30$) in order to prevent this effect.



(a) **USC detection and FP rate**



(b) **Mean of maximum confidence scores**

Figure 7.4: **Varying T_{var} and the corresponding results for $d = 60$ cm using $C_{min} = 0.5$.**

Grid Box Size

The grid box size is the amount of pixels of the width and height of boxes in the reflection grid. In the ideal case the boxes should perfectly encapsulate reflections such that all reflections caused by the USC are in the same box while other reflections are in other boxes. Unfortunately, because the screen detection module is not 100% accurate, misalignment occurs in the detected screen which causes small deviations in the location of reflections. This negatively impacts the performance of the detection system. Therefore, the grid box size should allow for such errors. In Figure 7.5 the effect of several grid box sizes is shown. It is clear that a small box size (2 or 3 pixels) confirms the hypothesis that the performance will suffer from this.

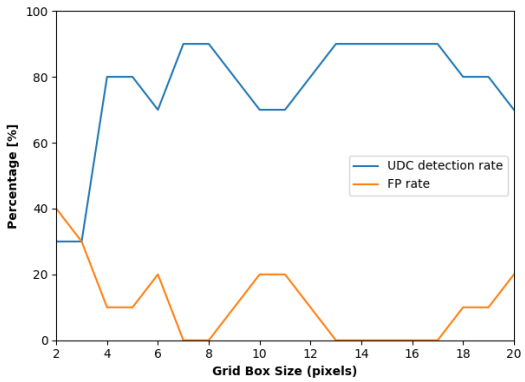


Figure 7.5: **Varying grid box sizes and the corresponding results for $d = 60$ cm using $C_{min} = 0.5$.**

7.2 Robustness Evaluation

7.2.1 Setup and Dataset

For the robustness evaluation of the proposed detection model, a variety of deployment scenarios and measurement conditions are taken into account that could impact the performance of the detection system. First, two deployment setups are used during the evaluation, chosen to represent two common deployment scenarios for screens:

- The first setup is a *table* scenery which is chosen to resemble the deployment scenario of a screen or monitor on a desk or table.
- The second setup is a *wall* scenery setup which is chosen to resemble the deployment scenario of screen in front of a wall.

A notable difference between the two setups is amount of emitted light by the ToF sensor that is reflected by the background scenery which is high for the *wall* setup and low for the *table* setup. The overview of different setups is shown in 7.6 and 7.7. To verify whether the detection system is impacted by measurement conditions, the following categories can be found in the dataset that is used for the robustness evaluation:

To verify whether the detection system is impacted by measurement conditions, a variety of conditions is included in the dataset used during the robustness evaluation. The following

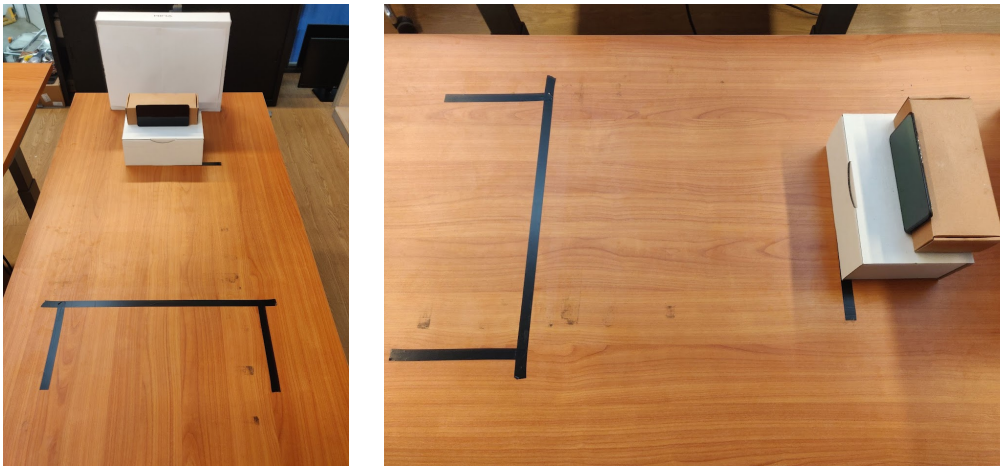


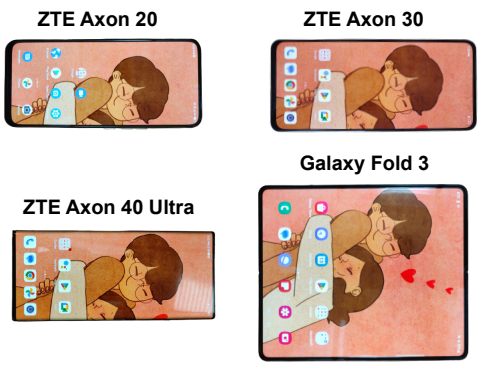
Figure 7.6: **Table Scenery Setup**

Figure 7.7: **Wall Scenery Setup**

section provides more detail on the results for each condition. An overview of the variations per condition be seen in Table 7.2 and in Figure 7.8. All four measurement conditions have been tested individually e.g. all other conditions are kept static while testing a certain condition. Just like the preliminary evaluation, each measurement setup is repeated 10 times to minimize the influence of variance in the results. Together, all measurements that have been done result in a dataset that contains a total number of *200* videos.

7.2.2 Overall Performance

Based upon the results from the preliminary evaluation, the detection system will be evaluated using three specific values for T_{var}: a low value of 20 which makes the model defensive in selecting reflections, a moderate value of 30 which should give balanced results between accuracy and FP rate and a high value of 40 which should favor a high USC detection accuracy over a low FP rate. Also, the impact of two grid sizes is investigated. Other parameters such as θ_N and θ_B are set to the same default values as used in preliminary evaluation. In Table 7.3 the results can be seen using a confidence threshold of $C_{min} = 0.5$. The best USC detection rate is 71.5% while having a FP rate of 21.5%. The results clearly show that increasing T_{var} also increases the USC



(a) **Different full-screen smartphones**

(b) **Varying ambient light conditions**

(c) **Different screen protectors**

(d) **Varying screen content**

Figure 7.8: **Overview of the variety of conditions used in the measurements**

detection rate but with the cost of increasing the FP rate. Interestingly a higher grid box size slightly improves both the USC detection rate and FP rate. This might be caused by some USC reflections now having the highest confidence score instead of the false positive.

The following sections will explore the results of the performance evaluation of the detection system under several conditions. The performance of the detection system on these conditions is closely examined. Unless stated otherwise, the measurements results and metrics shown are obtained using a T_{var} of 40 and grid box size of 8. These values are chosen for two reasons: (1) the overall performance shows the highest detection accuracy using these values and (2) the FP rate is also high, so it allows a further investigation into the origin of false positives.

Impact of ambient light conditions: the robustness of the detection system is evaluated under different ambient light conditions. Three different ambient light conditions are considered, shown in Figure 7.8b: (1) Bright artificial office lights; (2): Natural daylight; and (3) Night with minimal indoor illumination. These ambient light conditions are chosen to represent common scenarios in which people interact with screens. The impact of ambient light is presented in Figure 7.9a. It can be observed that independent of the ambient light, the USC can still be detected. This is expected because the ToF sensor operates using infrared light and should therefore be resilient to changes in the ambience of visual light. Further investigation showed that the underlying differences in detection rate and FP rate between the conditions are not caused by a difference in ambient light conditions but are the result of the variance in measurements.

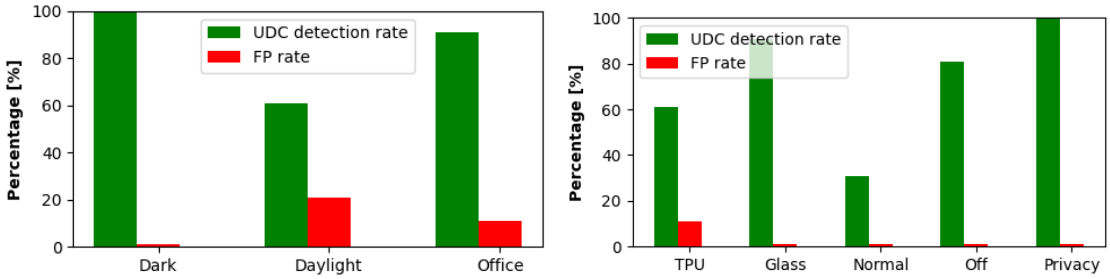
Full-screen smartphones	ZTE Axon 20, ZTE Axon 30, Samsung Galaxy Z Fold 3, ZTE Axon 40
Ambient Conditions	Bright Office Lights, Natural Daylight, Dark
Screen Content	Dynamic (video), Static (red, blue, green)
Screen Protectors	Glass, Plastic, Privacy, Normal (factory default), Off (removed)

Table 7.2: **Overview of the variations in the measurement conditions, used to construct the dataset for robustness evaluation.**

Hyperparameters	**USC Detection Rate (%)**	**FP rate (%)**
$T_{var} = 20$, Grid Box Size $= 5$	17.5	2.5
$T_{var} = 20$, Grid Box Size $= 8$	18.5	2.0
$T_{var} = 30$, Grid Box Size $= 5$	46.0	13.0
$T_{var} = 30$, Grid Box Size $= 8$	57.0	10.5
$T_{var} = 40$, Grid Box Size $= 5$	68.0	23.5
$T_{var} = 40$, Grid Box Size $= 8$	71.5	21.5

Table 7.3: **Results for several runs of the whole dataset using $C_{min} = 0.5$.**

Impact of screen protectors: the robustness of the detection system is furthermore evaluated when different screen protectors are used. Five different screen protector configurations conditions are used, of which shown are shown in Figure 7.8c: (1) Glass protector; (2): Plastic (TPU) protector; (3) Privacy protector, which makes the screen invisible outside a particular FoV; (4) Normal (e.g. factory default) protector which contains a cut-out hole in front of the USC; and (5) Off, in which the default screen protector is removed from the screen. The impact of these screen protectors on the detection rate is shown in Figure 7.9b. It can be observed that independent of the screen protector, the USC can still be detected. The screen protectors do not seem to hide the reflections of the USC from the ToF sensor, and also do not cause screen reflection does have interference with the USC. Just like the impact of the ambient light conditions, further investigation showed that the underlying differences are not caused by the screen protectors.



(a) **Results for different ambient conditions** (b) **Results for screen protector types**

Figure 7.9

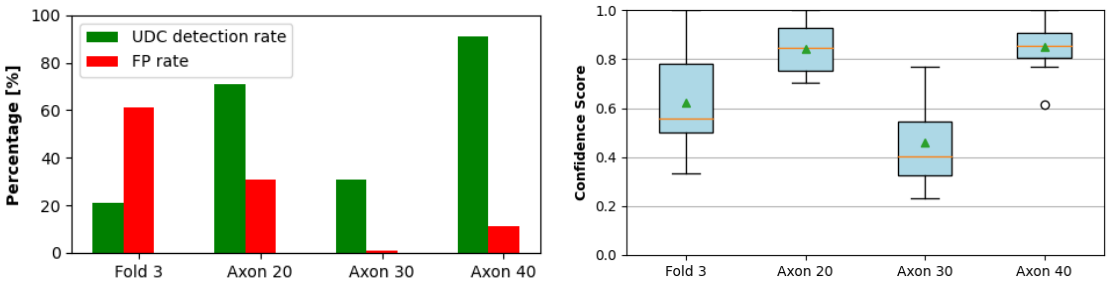
Impact of Screen Diversity

To assess the performance of the detection system on different screens on different full-screen smartphones, four different smartphones were selected and used to verify the performance (see Figure 6.2a): (1) ZTE Axon 20 (the first full-screen smartphone with USC); (2) ZTE Axon 30; (3) Samsung Galaxy Z Fold 3; and (4) ZTE Axon 40 Ultra. The results are shown in Figure 7.10, from which the following conclusions are drawn:

- The detection system shows a strong performance for the ZTE AXON 40 and a moderate performance for the ZTE AXON 20. From the confidence scores shown in 7.10b it can be seen that the detection system achieves high confidence scores which means that almost all the reflections from the USC can be captured and detected.

- The Samsung Galaxy Z Fold 3 causes a large amount of false positives. A further investigation shows that the screen detection module has problems detecting the mask of the screen. This is shown in Figure 7.11a, in which can be seen that the screen detection module identifies the large specular reflection from the ToF sensor as the screen's contours. The reason this happens for this particular smartphone, is that the shape of the specular reflection is different for the odd-sized Galaxy Fold screen. SAM, responsible for screen detection identifies this as an object of interest. Because the location prompt is set to the center of the screen, and because only one prompt is used, SAM will prefer this object over the screen. This could be fixed by increasing the amount of location prompts and aligning them with the screen's contour instead of only using the center location.

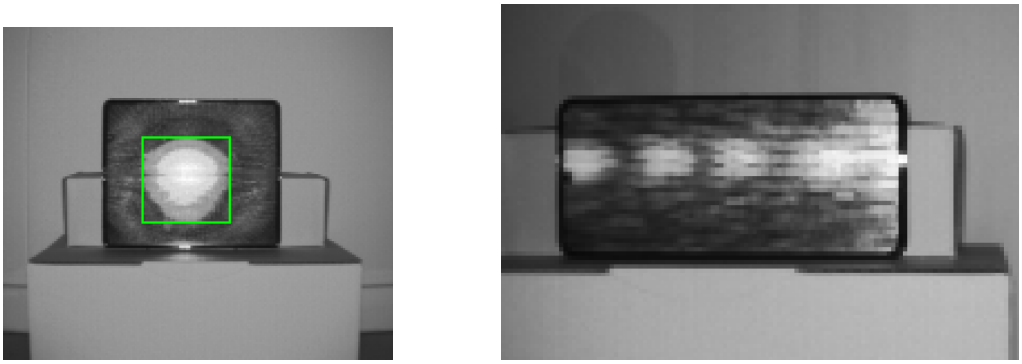
- Although the ZTE Axon 30 performs well in terms of a zero FP rate, its USC detection rate is rather low. From Figure 7.10b it can be seen that the confidence scores are rather low, having a mean of 0.46, which indicates that a lot of reflections from the USC are not detected. Further investigation shows that the specular reflections from this smartphone have a strong and wide horizontal component that interferes more than usual with the USC reflections, as can be seen in Figure 7.11b. For such screens, it can be possible to set the angle of θ_B to a larger value. Results from a separate run of the detection system for this smartphone where θ_B is increased from 5° to 7° shows that the USC detection rate increases to 90% while the mean confidence score increases to 0.68 and a zero FP rate is maintained.



(a) **Results for different full-screen smartphones**

(b) **Confidence scores of full-screen smartphones**

Figure 7.10

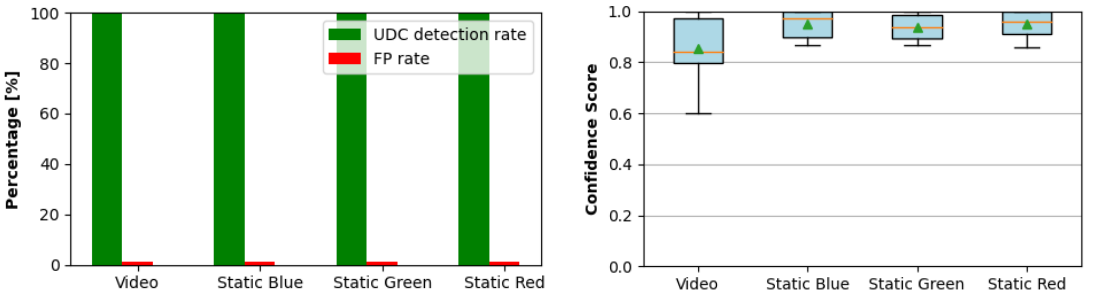


(a) **Screen detection errors for the Fold 3** (b) **Screen reflections for the ZTE Axon 30**

Figure 7.11

Impact of Screen Content

One of the important research challenges of this work is to find out if the USC can still be detected while the screen is showing content. To evaluate the impact of screen content, four different screen content scenarios are evaluated, which are shown in Figure 7.8d: (1) Dynamic screen content such as a video; Static screen content where (2) Blue; (3) Green; and (4) Red colors are displayed on screen's translucent region where the USC is hidden. To maximize the impact of the screen content, the brightness of the screen is set to 100% in order to maximize the possible interference. The results of the measurements can be seen in Figure 7.12a with corresponding confidence scores in Figure 7.12b, showing an excellent performance of the detection system. From this it can be concluded that the ToF sensor can still detect retro reflections behind the translucent region of the smartphone while content is displayed.



(a) **Results for variety of screen content** (b) **Confidence scores for varying screen content**

Figure 7.12

7.2.3 Confidence Score Threshold

For now all measurements were conducted using a confidence threshold C_{min} of 0.5. It is however important to provide insights in how the confidence score affects the operation mode of the detection system, such that the model could be deployed for different applications[45]:

- A low confidence score can be desired if a USC high detection rate is desired. This however also increases the FP rate. For example,
- A high confidence score can be desired if the reliability of the prediction made by the model is preferred. For example, when deployed in an industrial or automated fashion, this can guarantee the trustworthiness of the detection results.

In Figure 7.13 the impact of varying C_{min} from 0.0 to 1.0 can be seen using $T_{var} = 40$ and a grid box size of 8. It is important to remember that because the confidence threshold C_{min} is a threshold on the maximum found confidence score C_{found}, it is still possible to detect the USC while C_{min} is set to a low value. Because all samples in the dataset are from screens with a USC, it is therefore expected that the sum of the USC detection rate and the FP rate equals 100% when C_{min} is set to zero. The results show that the decline of the USC detection rate (around $C_{min} = 0.3$) sets in earlier than the decline of the FP rate (around $C_{min} = 0.7$) which suggests that the detection system struggles between these two confidence thresholds to eliminate false positives and is therefore not a suitable operating range. Around $C_{min} = 0.8$, the USC detection rate starts to fall with increased pace, which suggest that a balanced operating mode for the detection system should be found here.

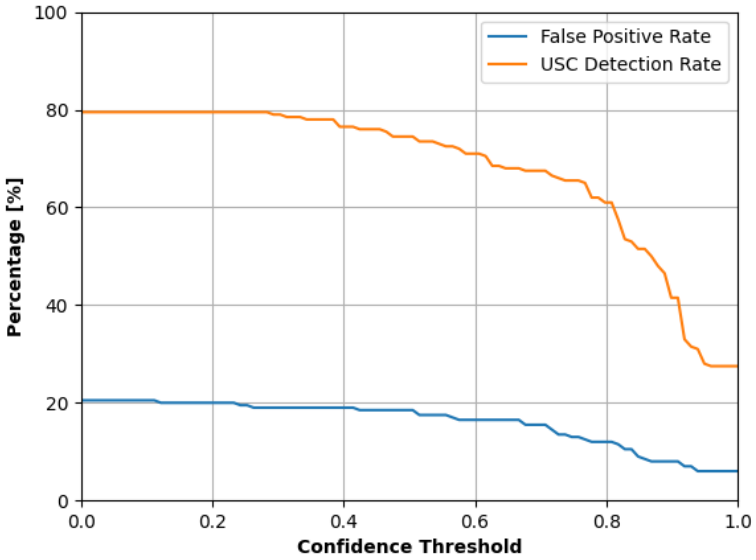


Figure 7.13: **Impact of the confidence threshold on the performance of the detection system.**

Chapter 8

Discussion and Future Work

The following sections provide some insights into ideas and thoughts that came up during this work that would be interesting to research or implement in future work.

8.1 Larger screen sizes

The decision model and its implementation in this thesis were specifically designed and evaluated for use on smartphone screens equipped with a USC, primarily because these are currently the only widely available screens integrating this technology. However, it is anticipated that the market segment for USCs will broaden to include other types of screens, such as tablets and large monitors. While the fundamental detection principle is agnostic to screen size, the methodologies and implementation strategies proposed in this work will require adaptations to accommodate devices larger than smartphones. A key challenge may arise when the screen size exceeds the FoV of the ToF sensor. A potential solution entails expanding the range of the ToF sensor from a one-dimensional horizontal sweep to a comprehensive three-dimensional scan. By broadening the FoV of the ToF sensor, not only can we accommodate larger screen sizes, but also enhance the scope and precision of the scanning process.

8.2 ToF placement

The proposed detection architecture and implementation still have some challenges before they can be turned into a market-ready product that can be used by people to scan screens on the presence of a USC. The methodology relies on some assumptions (constant measurement speed, constant ToF sensor height) that cannot be made when people would use it themselves. On the other hand, the possibility of deploying this solution in an automated fashion such as integration with cleaning robots, and drones would still provide a stable predictable measurement trajectory in which these challenges can be easily solved.

8.3 Online processing

To reduce the complexity during the implementation phase, an offline processing method was chosen such that recording and processing could be done independently of each other. However, an offline processing method has several disadvantages such as the lack of possibility to provide feedback to the user on how to perform the measurement. Also, online processing provides the

possibility to stop the detection process early if the detection system found a high-confidence USC reflection. A key challenge in shifting towards an online processing detection method lies in achieving (near) real-time screen mask detection, particularly in resource-constrained environments such as smartphones or embedded devices. The pursuit of a balanced approach that combines accuracy in screen detection with speed becomes imperative. At present, the Segment Anything Model (SAM) is used, offering high accuracy in screen detection but at the cost of extended processing times. Future work could consider more efficient segmentation networks such as the efficient segmentation tracking network [46], or FastSAM [47]. Alternatively, a dedicated screen detection network, potentially built upon the U-Net architecture [48] with less resource requirement, could be developed, trading off the generality of larger networks for increased screen mask generation speed. Such enhancements would pave the way towards a more interactive and efficient online processing method for USC detection.

8.4 Improved reflection detection

The reflection detection method built in this work can still be improved in order to reduce the FP rate. Especially reflections from screen edges can negatively impact the FP rate. A possibility would be to train a neural network on the reflection profile of USC reflections [15].

Chapter 9

Conclusion

In this work, the feasibility of detecting USCs using the phenomena of retro reflections has been researched. A small study was conducted to find out whether this detection principle can still be used to detect USCs. While this study showed that the retro reflections from the USC still can be observed, several challenges were also noted that obscure the detectability of the USC. Mainly these challenges come forth from the specular reflections that are caused by the screen that hides the camera. These reflections can look similar to reflections from a camera and also interfere with the retro reflections of the USC and greatly limit its detectability.

To overcome these challenges, a detection principle is proposed that aims to detect and capture the behavior of USCs retro reflections while being resilient to the specular reflections caused by the screen. The detection principle is based on the fundamental difference between retro reflections and specular reflections. While moving the ToF sensor in a sideward motion, the observed reflections from a USC maintain their location in the screen while specular reflections will move across the screen accordingly to the ToF sensor's motion. The principles are integrated into a system architecture that combines computer vision, image processing and machine learning techniques. An implementation of the detection system is built in order to evaluate the detection model. The combination of a Nimbus ToF sensor and Raspberry Pi is used to capture and stream amplitude images and the detection architecture is implemented in Python and run on a PC that is connected to the ToF sensor.

A preliminary evaluation was performed on the detection system in order to validate the detection principles that the system is built upon. In this evaluation, the screen detection module's performance is investigated because its accuracy is crucial to the final performance of the detection system. With an IoU score of 0.92 the module was deemed perfectly acceptable. To validate the feasibility of the detection system, a static test setup was used. In this setup, the ZTE AXON 40 Ultra is used while several target distances were tested in order to investigate its impact on the performance. For target distances within the range of 50-70 cm, the detection system achieves good results with a detection accuracy of roughly 77% for 30 measurements with only one false positive. For smaller target distances the amount of noise becomes dominant while for larger distances the reflections become less visible. Also, an ablation study is done to tune the hyperparameters of the detection system.

Finally, a robustness evaluation is performed on the detection system in which a variety of scenarios and conditions are tested. In two test setups, four different smartphones with a USC, different ambient conditions and a variety of screen protectors are tested. Also, the impact of the activity of the target screen on the detection system's performance is investigated. Using these different configurations, a dataset of 200 videos is created. Using the results of the preliminary

evaluation, the impact of several hyperparameters was investigated on the performance of the whole dataset. On the whole dataset, the detection system achieves a detection accuracy of 71.5% while the FP rate is 21.5%. A slightly more defensive configuration of the detection system achieves a USC detection rate of 57.0% while the FP rate drops to 10.5%. An in-depth investigation on the condition categories included in the dataset showed that the detection system is able to detect the USC independently of the screen activity, ambient conditions or screen protectors that were included in the dataset, from it can be concluded that the detection system is robust to a variety of conditions. The model's performance is most negatively impacted by some different screens, especially when the smartphone screen contains reflective edges.

Bibliography

[1] Statista Research Department. Number of spycam related crimes in South Korea from 2011 to 2021. https://www.statista.com/statistics/1133121/south-korea-number-of-spycam-crimes/, 2022.

[2] Courtney Linder. There Might Be Secret Surveillance Equipment in Your Vacation Rental. https://www.popularmechanics.com/technology/security/a32983077/vacation-rental-surveillance-spying/, 2020.

[3] Human Rights Watch (HRW). My Life is Not Your Porn. https://www.hrw.org/report/2021/06/16/my-life-not-your-porn/digital-sex-crimes-south-korea, 2021.

[4] Sehoon Lim, Luming Liang, Yatao Zhong, Neil Emerton, Tim Large, and Steven Bathiche. 18-3: Free viewpoint teleconferencing using cameras behind screen. *SID Symposium Digest of Technical Papers*, 52:218–221, 05 2021.

[5] Neil Emerton, David Ren, and Tim Large. 28-1: Image capture through tft arrays. *SID Symposium Digest of Technical Papers*, 51:402–405, 08 2020.

[6] Hanting Ye, Jie Xiong, and Qing Wang. When vlc meets under-screen camera. In *Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and Services*, MobiSys '23, page 343–355, New York, NY, USA, 2023. Association for Computing Machinery.

[7] Yuqian Zhou, David Ren, Neil Emerton, Sehoon Lim, and Timothy A. Large. Image restoration for under-display camera. *CoRR*, abs/2003.04857, 2020.

[8] Ruicheng Feng, Chongyi Li, Huaijin Chen, Shuai Li, Chen Change Loy, and Jinwei Gu. Removing diffraction image artifacts in under-display camera via dynamic skip connection network, 2021.

[9] Marcos V. Conde, Florin Vasluianu, Sabari Nathan, and Radu Timofte. Real-time under-display cameras image restoration and hdr on mobile devices, 2022.

[10] Sam Byford. The world's first under-display selfie camera isn't very good. https://www.theverge.com/2020/12/21/22191459/zte-axon-20-5g-under-display-camera-hands-on/, 2020.

[11] Simranjeet Singh, Rajneesh Sharma, and Alan F. Smeaton. Using gans to synthesise minimum training data for deepfake generation, 2020.

[12] Dinhnguyen Dao, Muhammad Salman, and Youngtae Noh. Deepdespy: A deep learning-based wireless spy camera detection system. *IEEE Access*, 9:145486–145497, 2021.

[13] Zhiyuan Yu, Zhuohang Li, Yuanhaur Chang, Skylar Fong, Jian Liu, and Ning Zhang. Heatdecam: Detecting hidden spy cameras via thermal emissions. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, CCS '22, page 3107–3120, New York, NY, USA, 2022. Association for Computing Machinery.

[14] Chun Liu, Changming Zhao, Haiyang Zhang, Zilong Zhang, Shuyuan Gao, and Yunshi Wang. Analysis of mini-camera's cat-eye retro-reflection for characterization of diffraction rings and arrayed spots. *IEEE Photonics Journal*, 11(4):1–12, 2019.

[15] Sriram Sami, Sean Rui Xiang Tan, Bangjie Sun, and Jun Han. Lapd: Hidden spy camera detection using smartphone time-of-flight sensors. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, SenSys '21, page 288–301, New York, NY, USA, 2021. Association for Computing Machinery.

[16] ZTE unveils world's first under-display camera smartphone Axon 20 5G. `https://www.zte.com.cn/global/about/news/20200901e2.html`, 2023.

[17] Under Display Camera Phones List — 2023 Edition. `https://www.thephonetalks.com/under-display-camera-phones-list/`, 2023.

[18] Sehoon Lim, Yuqian Zhou, Neil Emerton, Tim Large, and Steven Bathiche. 74-1: Image restoration for display-integrated camera. *SID Symposium Digest of Technical Papers*, 51(1):1102–1105, 2020.

[19] Zhibin Wang, Yilu Chang, Qi Wang, Yingjie Zhang, Jacky Qiu, and Michael Helander. 55-1: Invited paper: Self-assembled cathode patterning in amoled for under-display camera. *SID Symposium Digest of Technical Papers*, 51:811–814, 08 2020.

[20] Anqi Yang and Aswin C. Sankaranarayanan. Designing display pixel layouts for under-panel cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(7):2245–2256, 2021.

[21] Hanting Ye and Qing Wang. Spiderweb: Enabling through-screen visible light communication. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, SenSys '21, page 316–328, New York, NY, USA, 2021. Association for Computing Machinery.

[22] Can a thermal imaging camera detect hidden cameras? `https://www.quora.com/Can-a-thermal-imaging-camera-detect-hidden-cameras`, 2022.

[23] Adam Juniper. Best hidden camera detector in 2023: easily find bugs, trackers and spy cams. `https://www.digitalcameraworld.com/buying-guides/best-hidden-camera-detector`, 2023.

[24] How To Detect Hidden Cameras in 5 Simple Steps (And What To Do After). `https://alfred.camera/blog/how-to-detect-hidden-cameras/`, 2023.

[25] Cem Çatalbaş, Mustafa Nil, Yavuz Ozturk, Metin Nil, and Mehmet Akgül. Passive thermal management of an oled tv display. *OPTOELECTRONICS AND ADVANCED MATERIALS-RAPID COMMUNICATIONS*, 12:401–406, 08 2018.

[26] Junho Park, Seung Lee, Jongmin Kim, Dongpil Park, Woo Choi, and Wonbin Hong. An optically invisible antenna-on-display (aod) concept for millimeter-wave 5g cellular devices. *IEEE Transactions on Antennas and Propagation*, PP:1–1, 02 2019.

[27] Myoungsun Kim, Dongseop Lee, Yerim Oh, Jae-Yeong Lee, Bumhyun Kim, Junho Park, Dongpil Park, and Wonbin Hong. Antenna-on-display concept on an extremely thin substrate for sub-6 ghz wireless applications. *IEEE Transactions on Antennas and Propagation*, 70(7):5929–5934, 2022.

[28] ZTE Axon 40 Ultra. `https://ztedevices.com/en-us/axon-40-ultra/`, 2022.

[29] Pierre Boher, Thierry LEROUX, Thibault Bignon, and Véronique Collomb-Patton. Multispectral brdf measurements on anisotropic samples: application to metallic surfaces and oled displays. 02 2016.

[30] Xiaolong Liu, Zhidong Deng, and Yuhan Yang. Recent progress in semantic image segmentation. *Artificial Intelligence Review*, 52(2):1089–1106, jun 2018.

[31] Gabriela Csurka, Riccardo Volpi, and Boris Chidlovskii. Semantic image segmentation: Two decades of research, 2023.

[32] Kohei Arai and Supriya Kapoor, editors. *Advances in Computer Vision*. Springer International Publishing, 2020.

[33] Mithun Kar. A review on progress in semantic image segmentation and its application to medical images. *SN Computer Science*, 2, 09 2021.

[34] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[35] Ezio Malis and Manuel Vargas. Deeper understanding of the homography decomposition for vision-based control. 01 2007.

[36] Matthew Harker and Paul O'Leary. Computation of homographies. *British Machine Vision Conference*, Jan 2005.

[37] Hui Kong, Hatice Cinar Akakin, and Sanjay E. Sarma. A generalized laplacian of gaussian filter for blob detection and its applications. *IEEE Transactions on Cybernetics*, 43(6):1719–1733, 2013.

[38] S N Kumar, A. Fred, Ajay Kumar Haridhas, and S. Varghese. Medical image edge detection using gauss gradient operator. *Journal of Pharmaceutical Sciences and Research*, 9:695–704, 01 2017.

[39] Mouser Electronics. ams OSRAM TMF8701-1BM. `https://nl.mouser.com/ProductDetail/ams-OSRAM/TMF8701-1BM?qs=vLWxofP3U2xF2JnCZtHFhQ%3D%3D`, 2023.

[40] AliExpress. Pmd CamBoard Pico Flexx2 Monstar TOF Depth Camera with SDK Technical Support Spot. `https://www.aliexpress.com/item/1005003094782648.html`, 2023.

[41] Elektorstore. Pieye Nimbus 3D ToF Camera Module for Raspberry Pi. `https://www.elektor.com/pieye-nimbus-3d-tof-camera-module-for-raspberry-pi`, 2023.

[42] Zhongwei Tang, Rafael Grompone von Gioi, Pascal Monasse, and Jean-Michel Morel. A precision analysis of camera distortion models. *IEEE Transactions on Image Processing*, 26(6):2694–2704, 2017.

[43] Markus Proeller. Nimbus documentation. `https://nimbus-docs.readthedocs.io/en/latest/`, 2018.

[44] Markus Proeller. nimbus-python 0.0.6. `https://pypi.org/project/nimbus-python//`, 2020.

[45] Simon Wenkel, Khaled Alhazmi, Tanel Liiv, Saud Alrshoud, and Martin Simon. Confidence score: The forgotten dimension of object detection performance evaluation. *Sensors*, 21(13), 2021.

[46] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. *arXiv:2306.05422*, 2023.

[47] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything. *arXiv preprint arXiv:2306.12156*, 2023.

[48] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.