The Interdisciplinary Center, Herzliya

Efi Arazi School of Computer Science

# Using a Pair of Cameras for Low SNR Change Detection on Reflective Surfaces

M.Sc. Dissertation

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science (M.Sc.) Research Track in Computer Science

Submitted by Oded Hanson

Under the supervision of Dr. Yael Moses

October 2012

# Acknowledgments

I would like to express my appreciation to my supervisor, Dr. Yael Moses, for her guidance and support during the entire course of my M.Sc. studies.

I would also like to thank my wife, without whose support I would never have had the means to complete this project.

# Abstract

*Shiny floors, glasses and mirrors in indoor environments and wet surfaces in outdoor environments reflect objects that are not necessarily in the region of interest in the scene. Although such reflective surfaces are ubiquitous, they were hardly ever explicitly considered in previous studies. In this thesis we address the problem of change detection on such reflective surfaces, when foreground objects are relatively small in the scene, and their color is not guaranteed to differ significantly from the background. Such objects are said to have low signal-to-noise ratio (SNR). Our goal is to recover the directly visible changes and ignore the reflected ones. We make two contributions towards meeting this goal. Our first contribution is a theoretical analysis of change detection at the pixel level in the presence of reflective surfaces. The analysis is performed for both a single camera and a pair of wide baseline cameras in the context of low SNR. Our second contribution is a change detection algorithm that allows reflections from the ground to be removed, using a pair of cameras. By combining sensitive change detection at the pixel level with the two-camera algorithm, we are able to detect objects in a low SNR scenario while ignoring irrelevant reflections. Successful results of our algorithm on challenging scenes are presented.*

# Table of Contents

# List of Figures

# List of Algorithms

# Chapter 1

# Introduction

The problem of detecting changes with respect to a learned background has been substantially dealt with in recent literature. Reviews of the most common methods are presented in [Piccardi, 2004, Radke et al., 2005, Cristani et al., 2010]. This thesis addresses a new variant of the change detection problem where objects are located on a possibly reflective surface and the signal may be similar to the background.

Reflective surfaces such as shiny floors in an indoor environment and wet surfaces outdoors may be present in the scene. Although such reflective surfaces are ubiquitous, they were hardly ever explicitly considered in previous change detection studies. In fact, without any additional prior information regarding the scene or the object to be detected, it would be impossible from a single view to differentiate between a true object and a reflection of an object located outside of the region of interest (e.g., an object reflected from a perfect mirror). The problem becomes even more challenging when significant change between the possible foreground and background color values is not guaranteed, in particular when the objects are very small, that is, have a low signal-to-noise ratio.

We present a theoretical analysis of the challenges imposed by low SNR and reflective surfaces. In addition, we present a method for detecting foreground objects located on the ground surface while ignoring their reflections. Our method overcomes the challenges imposed by low SNR and reflective surfaces by using a of pair of cameras with overlapping fields of view in a wide-baseline configuration. In addition, we assume that the ground surface is known and may be reflective. For simplicity, we

1

consider a planar ground surface.

Most change detection algorithms use a pre-processing stage for determining the probability of each pixel to belong to the background (e.g., [Piccardi, 2004]) and then use post-processing to detect foreground objects. The core of the preprocessing stage is building a model of the background using a series of one or more images. Given a novel image, we classify each pixel (or region) as foreground by comparing it to the background model. The main differences between the various methods is how the background is learned and represented in order to be robust to gradual illumination changes, sudden illumination changes, dynamic background, camouflage, shadows, etc. A naive post-processing stage consists of thresholding the background probability of each pixel. However, meaningful foreground objects in images usually consist of multiple pixels connected as a single "connected component" or blob. Indeed, most change detection algorithms use this fact to further process the map of foreground probability using the methods of hysteresis thresholding, gap filling, discarding small blobs which constitute noise, etc.

In this thesis we address both the pre-processing stage, where the probability of each pixel to belong to the background is determined, as well as the post-processing where the objects are detected from the computed pixel probabilities.

**Reflections:** Figure 1.1 depicts the different types of background which might be viewed by a single camera focusing on a reflective surface. The blue arrow points to an image pixel which is the projection of the floor. The green arrow points to an image pixel which is the projection of a point on the floor reflecting the roof of a house which is outside the region of interest. Both the floor and the house are static objects and their image projections can be treated as *permanent background*. The reflection of a permanent background can be learned and treated using existing methods. However, as can be seen in the red arrow, a bird which is not located inside the region of interest, and is not a foreground object, is reflected by the surface and viewed by the camera. We call such background objects – which existing methods will erroneously classify as foreground – *transient background*. Such background might also include reflections of foreground objects, as can be seen in Figure 1.2 **(a)** and **(b)**. In these figures, a person is viewed by two cameras. A reflection of the person on the floor is also viewed by both cameras. However, since they are located in different positions, each camera views a different reflection of the

object.

Reflected objects are not considered as foreground, and hence, should not be detected as such. In addition, ignoring the reflections of foreground objects is necessary for obtaining their shape and exact position on the ground. This is similar to omitting the shadows of an object from a foreground region. Although reflections of objects within the scene may be thought of as similar to shadows, there are inherent differences between the two. Reflections retain edge and color information of the reflected object, whereas shadows do not. In addition, shadows can be regarded as flat objects on the ground surface, while reflected objects can be regarded as real objects in the scene that are behind (or below) the reflective surface. An illustrative example of this would be to look at the bannister in Figure 1.2 **(b)**. It is reflected on the floor and is a mirror image of the true object. The task of detecting and removing shadows was previously addressed and solutions suggested by means of color properties using a single camera (e.g., [Cucchiara et al., 2001]) or by detecting objects on the ground plane using a pair of cameras (e.g., [Lanza et al., 2007]). However, none of these methods are directly applicable for handling reflections.

**Pixel classification:** A preprocessing step of many change detection algorithms would be to classify each pixel as background or foreground, using the expected distribution of the grey-level (or color) of the background. The background distribution can be learned from the data (e.g., [Stauffer and Grimson, 1999]). However, the foreground distribution is often unknown and may overlap that of the background. Hence, the foreground classification method should be carefully chosen to ensure detection of low SNR objects.

Most existing change detection methods ignore low SNR since they often consider quite large objects (e.g., people or cars) that are assumed to be different than the background (e.g., for tracking). Probably for this reason, the choice of the threshold used to classify a pixel was usually not a problem in these methods. We select our threshold using a significance test that is guaranteed to provide the best possible detection performance subject to a predefined, acceptable false alarm rate.

A theoretical analysis is presented for pixels corresponding to a point on a reflective surface. We formally show that for an acceptable false alarm rate, the detection rate from a single camera is very small. The intuition for this result is that the background may contain reflected objects outside the
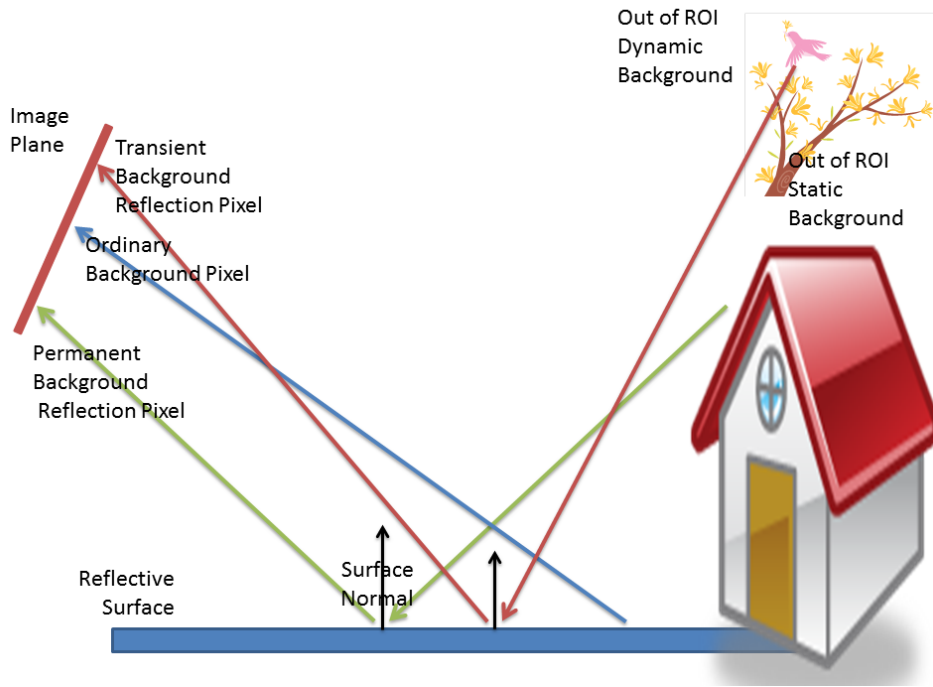
Figure 1.1: Possible background objects in an image of a reflective surface.

region of interest, which have the same distribution as a foreground object. However, since the reflected scene depends on the viewing direction, a pair of corresponding pixels taken from a pair of wide-baseline cameras can be used to reduce the ambiguity between foreground objects and reflected ones. We show how corresponding pixels in the two images can be used to significantly improve the ROC curve of each.

**Our Method:** The key idea of our method is to detect changes on the ground plane which are not due to reflections. To do so, a homography transformation is applied to align the ground plane pixels of the two images. Objects on the surface are those detected as foreground in both images while objects above the surface are detected in only one of the images. This idea was previously used for removing shadows that are detected as changes on the ground plane (e.g., [Lanza et al., 2007]). However, we note that reflected objects that can be interpreted as lying below the surface will also be detected in only one of the images. Moreover, removing shadows would also remove small objects on the surface.

Using this observation, our method detects only the foreground object and ignores its reflection.

The detected foreground objects on the ground plane are used as *seeds* for further processing. The foreground regions detected by a single camera are those connected to these seeds and located above them. Regions connected to the seeds but located below them are ignored. These are the reflections of the foreground regions.

The plane alignment is also used for classifying foreground pixels on the ground when the SNR may be low. The chances that uncorrelated noise from each camera will generate false detections in corresponding pixels are very low. Moreover, the different viewing angles of each camera increase the chances that one of the cameras will have a better view of the object, allowing the use of lower thresholds.

**Applications:**

One possible application of low SNR change detection is in the problem of foreign object debris on runway surfaces (FOD), as explained in the work of [Qunyu et al., 2009]. FOD was found to be the source of the crash of the Concorde at Charles de Gaulle airport back in 2000. A metal strip that fell off a DC-10 just few minutes before caused a tire to rupture, and the ensuing explosion caused the aircraft to crash, killing all 113 people on board. In such situations there is no prior knowledge of the object to be detected: any object which is not part of the runway surface should be detected. These objects can be very tiny screws and bolts, and they might even be pieces of detached asphalt chunks exhibiting very similar color to the background. The runway might be wet after rain and reflections of objects outside the region of interest might pose further difficulties on a detection system. Other examples could be camouflaged objects which vary only slightly from the background statistics. In all these examples, the challenge is made more difficult by the need to consider a reflective surface.

**Thesis Contribution:**

In this thesis we have made the following contributions:

- We explicitly deal with reflective surfaces for change detection, which was not done before. Dealing with such surfaces is necessary since reflective surfaces are present in many indoor as well as outdoor scenes.

- We provide a statistical background model which explicitly models the properties of reflective
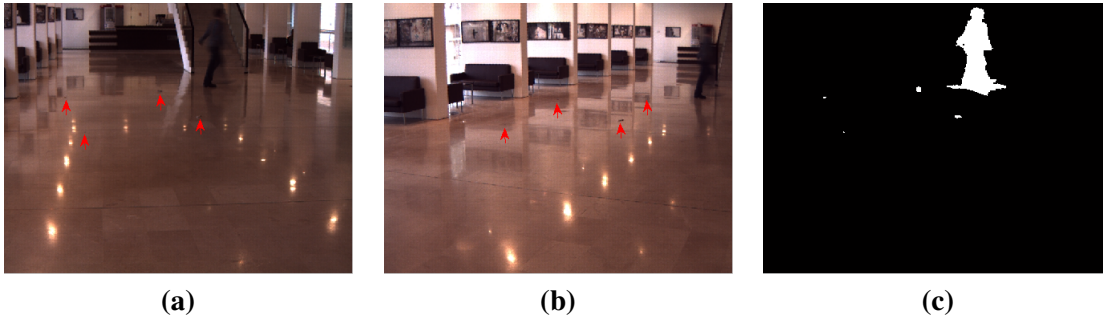
Figure 1.2: **(a)** An example of a scene containing 4 very small objects (with red arrows pointing to their locations) and a large foreground object (a person) walking in the scene. The person's self reflection can be seen on the floor too. **(b)** The same scene as **(a)** as seen by the second camera. **(c)** The final detection result of our method. All 4 objects were detected including the person. However, the person's self reflection has been eliminated.

backgrounds and theoretically analyze its performance. Such a background model is necessary for any change detection method that computes changes at the pixel level as its first step.

- We provide a statistical model of a reflective background as seen by two cameras. Such a background model is necessary for a change detection algorithm which uses two cameras.

- We develop a change detection method which has high detection performance while eliminating false detections due to reflections and noise.

- We deal with the detection of small and low SNR objects which is suitable for applications that do not have prior knowledge of the object to be detected. This is an important challenge for various applications such as FOD detection systems. Again, despite its important value, this challenge received only limited attention in previous work.

- Our algorithm is adaptive and automatically sets a pixel-wise threshold according to a predefined false-alarm rate parameter. This allows to achieve the best detection performance (high sensitivity) subject to an acceptable false alarm rate which is configured by the user.

- We provide a challenging data set for testing scenes with reflective surfaces and small objects that are similar to the background.

The rest of the thesis is organized as follows: In Chapter 3 we model the reflective background

model in a single camera as a mixture of the permanent and transient background models. The weight of the mixture depends on the rate at which transient background changes occur. We will show that the higher the prior probability for a transient change in the background, the poorer the classification of foreground objects will be.

In chapter 4 we provide a novel reflective background model based on two views of the reflective scene. We will assume a dense correspondence between pixels viewing the same scene points in both cameras. Since the reflected object viewed by a camera depends on the angle between the camera and the object being reflected, it is obvious that a wider baseline between the two views decreases the chance that two corresponding background pixels will both show a reflection of the same transient background object. The wider the baseline is, the slimmer that chance becomes. Moreover, since the probability of viewing different transient background objects in each view is independent, combining the information from two cameras will substantially reduce the false detection rate. Since true foreground objects will be seen by both cameras, this method will maintain high probability of detection.

In chapter 5 we present our full method for detecting objects on reflective ground planes using two views. Not only does our method handle transient background very well, but it also removes the reflected areas of foreground objects.

In chapter 6 we present the results of our method in different scenarios. In order to demonstrate the power of our method, we focus on detecting very small objects with low SNR, which are typically more difficult to detect than large objects with large contrast to the background.

# Chapter 2

# Previous Work

Background subtraction is commonly used in the literature for foreground segmentation. Good reviews of these methods are presented in [Radke et al., 2005] and [Piccardi, 2004]. Brutzer *et al.* ([Brutzer et al., 2011]) provide an evaluation of these methods' performance in various conditions such as gradual or sudden illumination changes, dynamic background, camouflage, and shadows. None of these papers attempt to deal with reflective surfaces or specular reflections. This is no coincidence since, as we will show in chapter 3, it is impossible to distinguish between reflections of objects outside and within the region of interest and true foreground objects using a single monocular camera. Any attempt, without use of additional prior knowledge, will either produce many false alarms or missed detections.

In [Cristani et al., 2010] a review of multi-sensor techniques for change detection is presented. In this paper the performance of each of the considered change detection methods was also evaluated on reflective surfaces. It was concluded that using multiple cameras, 3D structure, and depth information is necessary to deal with reflective surfaces. We will show that for our method only partial depth information that allows alignment of the ground surface is sufficient.

In the past few years several methods using multiple cameras have been proposed for robust change detection. Most of these use the depth or disparity from a pair of cameras. They are usually applicable only for narrow baseline configurations ([Krumm et al., 2000, Goldlucke and Magnor, 2003]). Goldlucke and Magnor [Goldlucke and Magnor, 2003] simultaneously compute the foreground and the depth by minimizing a discrete energy functional which evaluates both properties at the same

time. Another example of using a stereo camera for computing background subtraction is presented by [Bahadori et al., 2007]. There, the background model consists of intensity, disparity, and edge information. The 3D information is also used for improving segmentation. Gordon, *et al.* [Gordon et al., 1999] use both depth and color, with the depth coming from range data. Harville *et al.* [Harville et al., 2001] suggest a method in which the background is defined by per-pixel, time-adaptive, Gaussian mixtures in the combined input space of depth and luminance invariant. Zhao *et al.* [Zhao et al., 2005] use disparity for modeling the background. The main limitation of all these methods is the dependency on computing stereo correspondence between the two views. Stereo methods typically require narrow baseline configurations, where most configurations consist of a network of wide-baseline monocular cameras with overlapping fields of view. The online computation of dense stereo matches is usually very intensive and requires specialized hardware to handle it efficiently [Lim et al., 2005]. The most important limitation of these methods is when dealing with reflective surfaces such as glossy flooring. In this case, corresponding pixels on reflective surfaces are likely to have different color or texture since different parts of the scene are reflected to each of the images. Hence, the stereo algorithm is most likely to fail.

A novel work which performs the above tasks well in realtime without the need of any special hardware assistance is [Ivanov et al., 2000]. The authors construct the disparity fields of the background offline using dense stereo methods. This creates a pixel-to-pixel transformation from one image of an empty (background) scene to another. When a foreground object is in the scene, the colors between two corresponding pixels do not match because of the disparity changes. This method does not handle non-Lambertian surfaces well and also requires color calibration between the cameras.

Lim *et al.* [Lim et al., 2005] base their work on [Ivanov et al., 2000]. This is the only work in the literature which specifically addresses change detection in specular reflective surfaces. The authers suggest placing the cameras in a vertical configuration, which is optimal for minimizing occlusion shadows and missed detections as defined in [Ivanov et al., 2000]. However, the proposed configuration has some limitations. First, it does not fit many existing configurations of a network of overlapping wide FOV cameras covering as large an area as possible with a minimal number of cameras. There would usually also be some limitation on the required height for this configuration. Second, placing the cam-

eras vertically will make them parallel to the direction of the reflection on the surface, which is always towards the camera. This will, in fact, maximize the common area where both cameras view the same reflection in corresponding pixels. In our work we will show that a horizontal configuration with as large a base-line as possible will minimize this effect. Finally, the method proposed in [Lim et al., 2005] to eliminate foreground object pixels that are due to reflections on the surface is based on the vertical configuration and will not work on a wide baseline horizontal configuration.

A method which meets the requirements of a wide baseline configuration is presented in [Akman et al., 2008]. It relies on the homography between two views of a planar surface to create a combined background model. The background model is based on a multivariate Gaussian mixture model, which is a generalization of the model initially proposed by [Stauffer and Grimson, 1999]. Then, the Mahalanobis distance between a pair of corresponding pixel values and each multivariate Gaussian is computed. If the pixel pair does not match any Gaussian, it is considered as a new foreground pixel. This method has a few shortcomings. It does not handle non-planar objects which are above the surface. Thus, this method is more suitable for cases where the cameras are placed high above the scene, where the height of the object becomes negligible. The other problem is that the classification method that uses the Mahalanobis distance does not handle transient background pixels well. In such cases, only one of the corresponding pixel values will incorporate the color change while the other pixel will match the background model. The Mahalanobis distance will still be large and thus this pixel will not match any Gaussian and will be erroneously classified as foreground.

The approach most closely related to ours is presented in [Khan and Shah, 2006]. The main motivation is tracking people in crowded scenes, but the authors employ a multi-view background subtraction method as a first step. Since their motivation was to track the people's feet on the ground plane, the authors do not mention that their method can effectively handle reflective surfaces, as we indeed show in this work. Also, they do not put any emphasis on the threshold selection since they assume that the foreground objects will have high contrast relative to the background. In our work we attempt to detect very small objects with low SNR and propose a foreground classification method that can be statistically proven to be optimal.

# Chapter 3

# Single Camera Limitations

The objective of this chapter is to formalize and analyze change detection from a single camera at the single pixel level. Computing changes at the pixel level is a common step in many existing change detection methods (e.g., [Piccardi, 2004]). The post-processing step of integrating connected components of changed pixels into foreground blobs is addressed in chapter 5.

Here we formalize and analyze the required threshold to optimize the performance of an algorithm in the presence of reflective surfaces in the scene. In particular, we consider scenes with low SNR. For such scenes, discriminating between foreground and background pixels is more challenging and this challenge clearly increases in the presence of a reflective surface. We prove the limitations of a single camera to detect changes at the pixel level when reflective surfaces are presented. In particular, we show that for an acceptable false positive detection rate, the probability of true detection is very low.

## 3.1 Foreground Classification Method

Throughout this work we will use hypothesis testing in the same manner used in [Aach and Kaup, 1995], to classify the source of each pixel as a projection of a foreground or a background scene point, which we name foreground and background pixels respectively. We will use the same notation as defined in [Duda et al., 2001] and also used in [Aach and Kaup, 1995]. We consider the pixel value $x$ to be a continuous random variable. The value of $x$ can be scalar in the case of a single channel (gray) camera, but

can also be a vector of values in the case of multi-channel images (for example RGB). The distribution of $x$ depends on the state of nature (where the states of nature can be either background or foreground) and is denoted as $P(x|B)$, indicating the conditional density function of $x$ given that the true state of nature is background (that is, the value of $x$ originates from a background pixel) or $P(x|F)$ indicating the conditional density function of $x$ given that the true state of nature is foreground.

The likelihood ratio test is used to classify the source of a given pixel value $x$ to be a background or a foreground scene point:

$$\Lambda\left(x\right) = \frac{P\left(x|B\right)}{P\left(x|F\right)} \underset{F}{\overset{B}{\gtrless}} t.$$ (3.1)

We classify the pixel as background if the likelihood ratio is above the threshold $t$, and otherwise we classify it as foreground. The threshold $t$ defines the tradeoff between false positive and false negative detections. When using this formulation, we must decide (i) how to set $P\left(x|B\right)$, (ii) how to set $P\left(x|F\right)$, (iii) how to set $t$. How to best set $P\left(x|B\right)$ was extensively studied in the literature, whereas how to best set $P\left(x|F\right)$ is application specific. Both are discussed in Sections 3.2-3.3. The problem of setting the threshold $t$, which we next describe, has received less attention in previous work since signals with low SNR were rarely considered.

According to [Aach and Kaup, 1995] we may choose the threshold $t$ such that

$$\alpha = P\left(\Lambda\left(x\right) < t \,|B\right),$$ (3.2)

where $\alpha$ is a predefined acceptable false alarm rate. This procedure is termed a significance test with the false alarm rate $\alpha$ being the significance. According to the Neyman-Pearson theorem [Neyman and Pearson, 1933] this procedure will maximize $P_d$, the probability of detecting a true change, subject to the predefined probability of false alarm $\alpha$.

We denote the set of values $x$ that are classified as foreground by $R\left(t\right)$. Formally, $R\left(t\right)$ is defined by:

$$R\left(t\right) = \left\{x \,|\Lambda\left(x\right) < t\right\}.$$ (3.3)

In order to compute the false alarm rate $\alpha$ we need to consider the set of false detections: pixels belonging to the background but classified as foreground. Formally, it is given by integrating $P(x|B)$ over $R(t)$:

$$P\left(\Lambda\left(x\right)<t\,|B\right)=\int\limits_{x\in R(t)}P\left(x|B\right)dx=\alpha \tag{3.4}$$

A closed form solution to the above integral is beneficial. In particular, it allows the detection threshold to be efficiently updated when the background model is updated. In some cases, a closed form solution to 3.4 can be obtained, as presented in the next section. In other cases, it can be solved numerically and used to generate ROC curves (e.g., [Duda et al., 2001]).

## 3.2   Nonreflective Background

We will now analyze the classification method of the simple single Gaussian case, which is used in [Wren et al., 1996] for gray level images. We analyze its theoretical performance in the presence of a non-reflective surface (the ordinary case). This method was chosen because it is easy to analyze mathematically. The same analysis can be performed for color images, mixture of Gaussians ([Stauffer and Grimson, 1999]), and other methods presented in [Piccardi, 2004], but the analysis expected to be far more rigorous. In our analysis the background distribution of each pixel, $P(x|B)$, is independently modeled as a Gaussian distribution parameterized by $\mu$ and $\sigma$, which are both estimated using a sequence of background images (see, for example, Chapter 5). Although not explicitly stated in [Wren et al., 1996], it is assumed that the foreground (without any additional prior knowledge) is uniformly distributed, and hence its value is constant. We further assume that the pixels' gray level values are valid in the range of 0-255. Thus we obtain the following distribution functions for the background and foreground respectively:

$$P\left(x\,|B;\mu,\sigma\right)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{3.5}$$

$$P(x|F) = \begin{cases} \frac{1}{255} & 0 \leq x \leq 255 \\ \\ 0 & \text{else} \end{cases}. \tag{3.6}$$

By substituting equations 3.5 and 3.6 into 3.1, the likelihood ratio can be replaced by

$$\frac{P(x|B;\mu,\sigma)}{P(x|F)} = \frac{255}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \mathop{\gtrless}_{F}^{B} t. \tag{3.7}$$

By applying a $log$ on both sides and simple algebraic manipulation we obtain

$$\left|\frac{x-\mu}{\sigma}\right| \mathop{\gtrless}_{B}^{F} -\sqrt{2\log\left(\frac{t\sqrt{2\pi}\sigma}{255}\right)} = t_1. \tag{3.8}$$

Thus, we may conclude that the region of $x$ where the likelihood ratio value is above some constant threshold $t$ is equivalent to the Mahalanobis distance ($\frac{x-\mu}{\sigma}$) being less than a constant threshold $t_1$ that is defined above. It follows that the region $R(t)$ in equation 3.3 is equivalent to

$$R(t_1) = \left\{ x \,\middle|\, \left|\frac{x-\mu}{\sigma}\right| > t_1 \right\}. \tag{3.9}$$

Then, for a given false alarm rate $\alpha$, $t_1$ is obtained (according to equation 3.4) by solving the following integral

$$\int_{x \in R(t_1)} P(x|B)\,dx = \alpha, \tag{3.10}$$

which is equivalent to solving

$$\int_{x \notin R(t_1)} P(x|B)\,dx = 1 - \alpha. \tag{3.11}$$

Substituting equations 3.9 and 3.5 into the above, we get

$$1 - \alpha \quad = \quad \int\limits_{\mu-\sigma t_1}^{\mu+\sigma t_1} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx$$

$$(3.12)$$

$$= \quad \int\limits_{-t_1}^{-t_1} \frac{1}{\sqrt{2\pi}} e^{-x^2} dx,$$

and then finally

$$1 - \alpha = F_N\left(t_1\right) - F_N\left(-t_1\right) = 2 \cdot \left(F_N\left(t_1\right) - 0.5\right), \quad (3.13)$$

where $F_N$ is the cumulative distribution function of the normal distribution. For a false alarm rate $\alpha = 0.001$, for example, we will choose, according to the normal CDF, a value of $t_1 = 3.2905$.

It turns out that the decision rule produced by our analysis is identical to that proposed in [Wren et al., 1996], where the Mahalanobis distance is compared against a threshold.

Figure 3.1 shows ROC curves depicting this classification performance for different values of $\sigma$, assuming $\mu = 128$ and the foreground pixel is uniformly distributed between 0 and 255. Not surprisingly, it can be seen that for pixels with larger $\sigma$ values (i.e., noisier pixels), the expected detection rate decreases. We added specific data cursors for the different ROC curves, indicating the values where $\alpha$ is 0.001. This will be an acceptable false alarm rate in many cases. For example, the expected number of false detections in a 1M pixel image is about 1000. Since the pixels are independent, these sporadic detections will be uniformly distributed along the image space, and the probability of connected components (see chapter 5) in the image being created due to noise will be very small.

## 3.3   Reflective Background

We now introduce a new concept to our description of a background. Up to now, all previous work has assumed that the background is static. The more advanced methods attempt to handle some form of dynamic background such as swaying trees and bushes (see [Radke et al., 2005, Piccardi, 2004]). In our environment the background consists of reflective surfaces. In that case, the intensities viewed by the camera might be some static background reflected from the surface (e.g., the photo frames and staircase
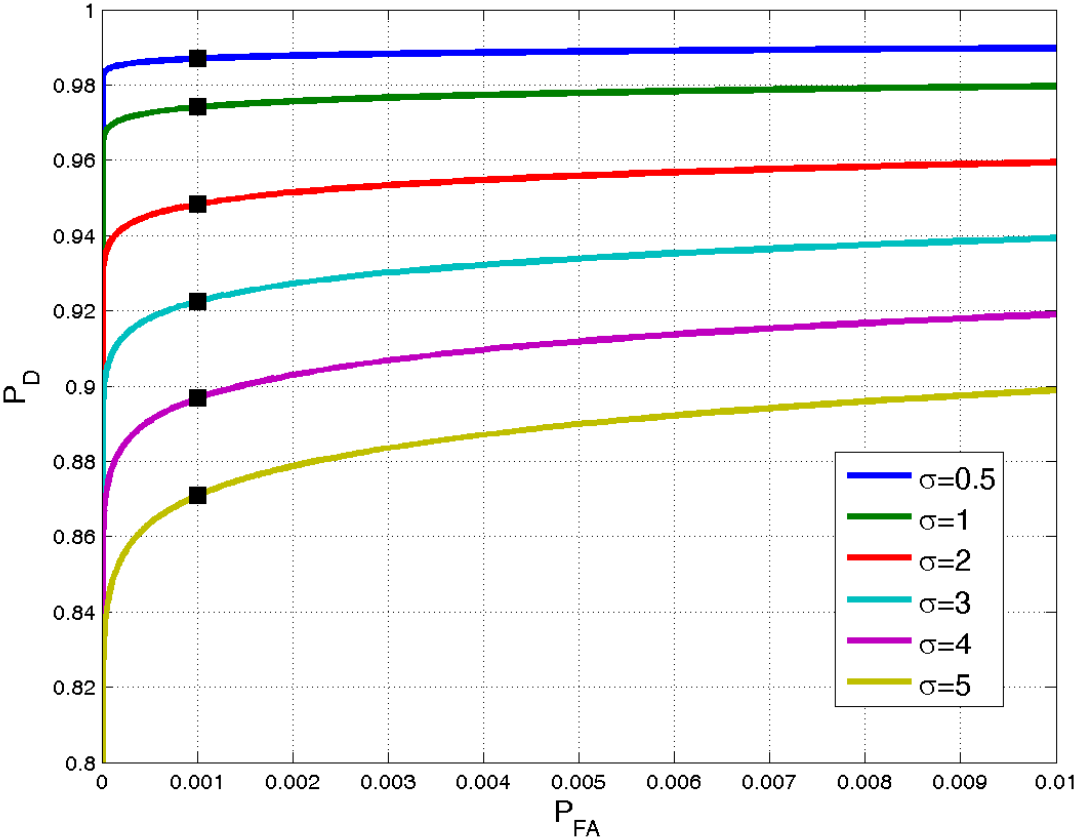
Figure 3.1: ROC curves of single Gaussian performance for different $\sigma$ values. The data cursers indicate the expected $P_D$ for each curve for $P_{FA}$ value of 0.001.

in Figures 1.2 **(a)** and **(b)**). We will call this the permanent background scene, for which the reflection can be modeled by existing methods. However, non-static objects not physically located in the area of interest might be reflected and captured by the camera. These objects, if not correctly handled by the background model, will obviously produce false alarms. We call this kind of background a *transient background*. An example of such objects are the lights in the ceiling reflected by the floor as seen in Figures 1.2 **(a)** and **(b)**. These reflections might appear or disappear when turning the light switch on and off. Another example of a transient background object can be seen in Figure 3.2. A person is walking out side the room but is seen through the glass window and is reflected on the floor.



Figure 3.2: An example of a transient background object reflected on the floor. The person is walking out side the room however, he is seen through the window and reflected on the floor. This will produce a false detection.

We next show that without additional prior knowledge of the intensity distribution of reflected objects, it is (almost) impossible to distinguish between true objects (objects which are in the area of interest) and reflections. We use the same flow as in the previous section: first the foreground and background models are defined, then the classification rule is shown to be given by the Mahalanobis distance, and finally it is shown that for an acceptable false alarm rate, the detection rate is very low.

### 3.3.1   Background and Foreground Probability

The background model is assumed to be a mixture of static and non-static distributions. At any given moment, the intensity of the pixel measured by the camera might either be a projection of the static (permanent) background or a reflection of a transient background. We denote the prior probability,

$P(B_P)$, as the probability that at any given time, the measured pixel value will be the projection of the permanent background. Then, the prior probability that a pixel value will be a projection of a transient background is given by $P(B_T) = 1 - P(B_P)$.

We model the permanent background distribution as a single Gaussian distribution with the parameters $\mu$ and $\sigma$ identical to the one in Eq. 3.5. Thus we have:

$$P\left(x \,|B_P; \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}. \tag{3.14}$$

Without any additional information about the transient background and its reflection properties, we model by a uniform distribution the transient background as in Eq. 3.6. Thus we have:

$$P\left(x|B_T\right) = \begin{cases} \frac{1}{255} & 0 \le x \le 255 \\ 0 & \text{else} \end{cases}, \tag{3.15}$$

where transient background is denoted by $B_T$. Then, using 3.14 and 3.15, we get the combined background density:

$$P\left(x \,|B; \mu, \sigma\right) = P\left(x \,|B_P; \mu, \sigma\right) \cdot P(B_P) + P\left(x|B_T\right) \cdot (1 - P(B_P)). \tag{3.16}$$

The foreground distribution is assumed to be identical to the foreground distribution of the non-reflective model and is equal to equation 3.6.

### 3.3.2 Classification Performance

We next show that the classification is equivalent to the Mahalanobis distance from a Gaussian. Hence, the limits of the integral are similar to the simple case with non-reflective surfaces. However, the function that is required to integrate is more complex. We substitute equations 3.16 and 3.6 into equation 3.1 and get

$$\Lambda\left(x\right) = \frac{P\left(x \,|B_P; \mu, \sigma\right) \cdot P(B_P) + P\left(x|B_T\right) \cdot (1 - P(B_P))}{P\left(x|F\right)} \underset{F}{\overset{B}{\gtrless}} t. \tag{3.17}$$

By simple algebraic manipulation we obtain:

$$P\left(x\,|B_P;\mu,\sigma\right)\underset{F}{\overset{B}{\gtrless}}\frac{t\cdot P\left(x|F\right)-P\left(x|B_T\right)\cdot\left(1-P(B_P)\right)}{P(B_P)}=t_1. \tag{3.18}$$

We denote by $t_1$ the right-hand side of the inequality, which is constant. By substituting with 3.14, some simple manipulation and applying log to both sides, we get, as in equation 3.8, that

$$\left|\frac{x-\mu}{\sigma}\right|\underset{B}{\overset{F}{\gtrless}}t_2, \tag{3.19}$$

and thus we have obtained a detection region similar to the one we found in equation 3.9. Then, using the same reasoning, we can obtain the threshold $t_2$ given the false alarm rate $\alpha$ by solving the following integral:

$$\int_{\mu-\sigma t_2}^{\mu+\sigma t_2} P\left(x\,|B_P;\mu,\sigma\right)\cdot P(B_P)+\frac{1-P(B_P)}{255}\,dx=1-\alpha, \tag{3.20}$$

after which we obtain

$$P(B_P)\left(F_N\left(t_2\right)-F_N\left(-t_2\right)\right)+\frac{1-P(B_P)}{255}2\sigma t_2=1-\alpha, \tag{3.21}$$

where $F_N$ is the cumulative distribution function of a standard normal distribution. Note that $\mu$ was eliminated from the equation but, unlike in the previous case, $\sigma$ remains.

Once $t_2$ is solved, given $\alpha$ and $\sigma$, the classification rule, similar to the simple Gaussian model, is simply the Mahalanobis distance compared to $t_2$. Unfortunately, there are no analytical means to solve 3.21. Nonetheless, the function is easy to sample and can be used to generate ROC curves. These curves, shown in figures 3.3 **(a)** and **(b)**, allow us to analyze the theoretical properties of this model.

### 3.3.3   Single Camera Limitations

Using the formulation and given the acceptable false alarm rate, $\alpha$, we next study the detection rate, $P_D$. The ROC curve depends on $\sigma$ as well as on $P(B_P)$, as can be seen from Equation 3.21. Note that $F_N()$ also depends on $\sigma$. The ROC curves were computed for different values of $0.5\leq\sigma\leq5$, for a fixed
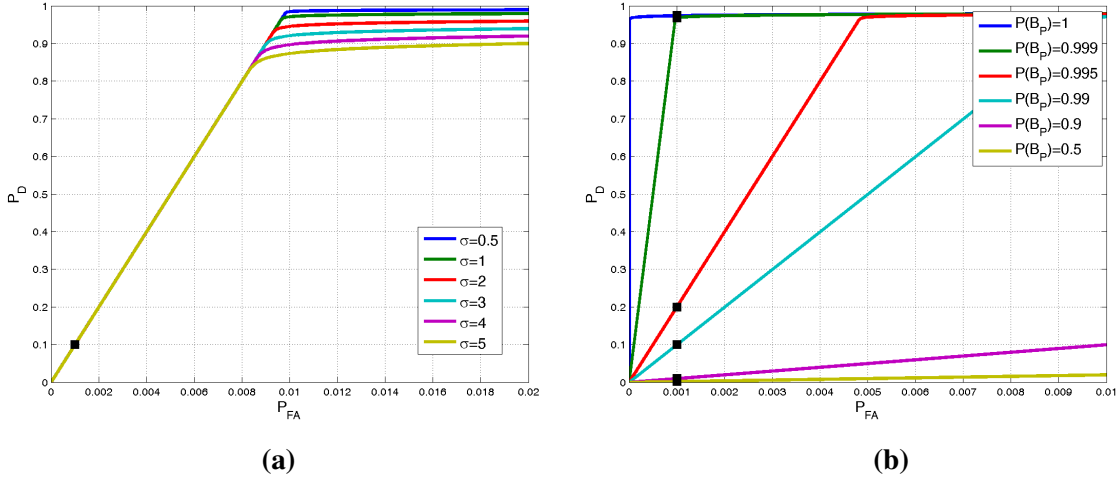
(a)                                                          (b)

Figure 3.3: **(a)** ROC curves of single Gaussian performance with reflective background for different $\sigma$ values. The curves where generated with the value $P(B_P) = 0.99$. **(b)** ROC curves of single Gaussian performance with reflective background for different $P(B_P)$ values. The curves where all generated with the value $\sigma = 1$.

$P(B_P) = 0.99$ (Figure 3.3). It can be seen that for all considered $\sigma$ values, the ROC curves converge to the same linear function for which the same $P_D$ value of 0.1 and $\alpha$ value of 0.001 is obtained (the same $\alpha$ as considered in Section 3.2). This can also be shown analytically when considering large enough values of $t_2$. For large values of $t_2$ the integral of $P(x|B_P; \mu, \sigma)$ will be approximately 1. Substituting $P(x|B_P; \mu, \sigma) = 1$ into equation 3.21 we get that:

$$P(B_P) - \frac{1 - P(B_P)}{255} 2\sigma t_2 = 1 - \alpha, \tag{3.22}$$

which can be solved for $t_2$, giving us $t_2$ as a function of $P(B_P)$ and $\alpha$:

$$t_2 = \frac{255 \left(1 - P(B_P) - \alpha\right)}{2\sigma \left(1 - P(B_P)\right)}. \tag{3.23}$$

Using $t_2$ we wish to compute the value of $P_D$, which is given by integrating $P(x|F)$ over $R(t)$, which in our case would be:

$$\int_{\mu - \sigma t_2}^{\mu + \sigma t_2} P(x|F) \, dx = \int_{\mu - \sigma t_2}^{\mu + \sigma t_2} \frac{1}{255} \, dx = 1 - P_D. \tag{3.24}$$

**20**

Solving the above integral we get:

$$P_D = \frac{255 - 2\sigma t_2}{255}. \tag{3.25}$$

Finally, substituting 3.23 into 3.24 we get the final result for $P_D$:

$$P_D = \frac{\alpha}{(1 - P(B_P))}. \tag{3.26}$$

We can see from equation 3.26 that $P_D$ is a linear function of $\alpha$ (for small enough values of $\alpha$). We can also see that this function's slope depends only on $P(B_P)$.

In figure 3.3(**b**), the ROC curves are computed using different values of $P(B_P)$; thus, it can be seen that each curve begins with a linear part for which the slope is different. This figure also depicts the drastic reduction of the $P_D$ as $P(B_P)$ decreases. Note that for $P(B_P) = 1$, there are no transient reflections. Hence, it corresponds to the case of no reflections. This can be seen by comparing it to Figure 3.1. However, even for a small probability (0.005) of transient background, for which $P(B_P) = 0.995$, the performance decreases to $P_D$ of 0.2 for $\alpha = 0.001$, which would be unacceptable in most systems. This analysis motivates the use of an additional camera.

## 3.4   Summary of Single Camera Limitations

In this chapter we showed that the detection performance induced by the simple mechanism of background subtraction decreases drastically when reflective surfaces appear in the background. This is quite obvious since the very essence of background subtraction is to identify regions that differ from the normal background and since reflections of objects look identical to true foreground objects when viewed by a camera. Thus, the only single-camera method for reducing false alarms induced by these reflections is post-processing, additional prior knowledge, and trying to eliminate changed blobs after the change takes place, as is done, for example, in [Yoon and Kweon, 2004].

In the next chapter, we will provide a framework that uses a wide baseline stereo configuration to create a dual-view change detection method that handle reflective surfaces very well without the need

for complex post-processing and additional prior knowledge.

# Chapter 4

# Pixel Level Two Camera Analysis

In chapter 3 we discussed detection from a single camera in the presence of a reflective surface. We showed that it is virtually impossible to produce a system for detection with good performance (low false alarm rate with high detection rate) without any additional prior knowledge about the object to be detected.

The objective of this chapter is to formalize and analyze the information available from two images taken from different viewpoints. We propose a method that combines the information from a pair of cameras in order to overcome the limitations of a single camera as presented in Chapter 3. In particular, we consider the detection of changes in a pair of corresponding pixels in two images. The manner in which the correspondence is obtained can vary, and a specific application which uses a homography based correspondence is discussed in Chapter 5. Again, we consider scenes with low SNR and reflective surfaces. As in the previous chapter, we will develop a hypothesis testing method to obtain the optimal decision boundary, but the background and foreground distribution functions will now be of a pair of pixel values from two images.

## 4.1 Reflective Background Using Two Images

We assume two cameras, each maintaining an independent background model. The PDF of each background is the same as described in equation 3.16. The background model for each camera is indepen-

dent since we assume that the noise arising from the permanent model is independent and there is no correlation between the color values each camera is viewing. Since the PDF is independent between the cameras, we may use univariate Gaussian distributions in our method instead of the multivariate distributions proposed in [Akman et al., 2008].

Hence, given two corresponding pixel values from each camera, $x = (x_L, x_R)$, the probability density function for $x$ given that the state of nature is background, is:

$$P\left(x \, | B; \mu_L, \sigma_L, \mu_R, \sigma_R\right) = P_L\left(x_L | B; \mu_L, \sigma_L\right) \cdot P_R\left(x_R | B; \mu_R, \sigma_R\right), \tag{4.1}$$

where $P_L\left(x_L | B; \mu_L, \sigma_L\right)$ is the conditional density function of left camera pixel $x_L$, given that it comes from the left camera view of the background. $P_R\left(x_R | B; \mu_R, \sigma_R\right)$ is the conditional density function of right camera pixel $x_R$, given that it comes from the corresponding camera view of the background.

We assume the foreground distribution in each camera to be the same as in section 3.2 equation 3.6. We also assume that the distribution in each camera is independent of the object. This is true since the views might be from very wide baseline cameras and we do not wish to impose correlation between the color and texture values in each view. Thus the foreground distribution for a pair of views would then be

$$P\left(x | F\right) = \begin{cases} \frac{1}{255^2} & 0 \leq x_L, x_R \leq 255 \\ 0 & \text{else} \end{cases}. \tag{4.2}$$

## 4.2 Classification Performance

We substitute equations 4.1 and 4.2 into the likelihood ratio function, which is valid in the range $0 \leq x_L, x_R \leq 255$:

$$\Lambda\left(x\right) = \frac{\left[P(B_P)\frac{1}{\sqrt{2\pi}\sigma_L}e^{-\frac{1}{2}\left(\frac{x_L - \mu_L}{\sigma_L}\right)^2} + \frac{1 - P(B_P)}{255}\right] \cdot \left[P(B_P)\frac{1}{\sqrt{2\pi}\sigma_R}e^{-\frac{1}{2}\left(\frac{x_R - \mu_R}{\sigma_R}\right)^2} + \frac{1 - P(B_P)}{255}\right]}{\frac{1}{255^2}} \underset{F}{\overset{B}{\gtrless}} t \tag{4.3}$$
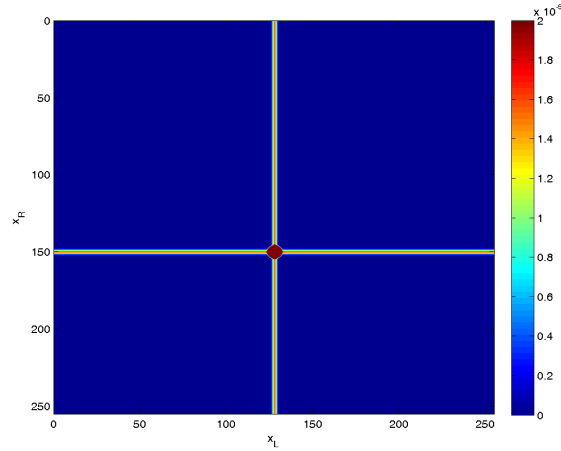
Figure 4.1: The likelihood function for two camera detection

To better understand how this likelihood function affects classification, we consider a concrete example. We use values of $\mu_L = 128$, $\sigma_L = 1$, $\mu_R = 150$, $\sigma_R = 1$ and $P(B_P) = 0.99$, as depicted in Figure 4.1. It can be seen that highest likelihood ratio is when both $x_L$ is close to $\mu_L$ and $x_R$ is close to $\mu_R$. This occurs when the values viewed in both pixels are from the permanent background. Since the prior probability for permanent background is $P(B_P) = 0.99$, then the prior probability for a reflection of a transient background object is 0.01. Thus there is still a high (but less likely) chance that $x_L$ or $x_R$ will be from a transient background object. Consequently, we can still see quite high likelihood ratio values when $x_L$ is close to $\mu_L$ but $x_R$ can be any value in the range of 0-255 and vice versa. Since the chance of viewing a reflection of two different transient background objects in both pixels is much lower – $0.01 \cdot 0.01 = 0.0001$, we can see in the figure that the likelihood ratio where both $x_L$ is far from $\mu_L$ and $x_R$ is far from $\mu_R$ is lowest.

Similar to the single camera case, setting a specific value of threshold $t$ will produce a region in the space of $x$ denoted $R(t)$, as defined in equation 3.3. Unlike the single camera case, however, the space here is two-dimensional and there is a correlation between the two dimensions. This can be seen in Figure 4.2(a), which was obtained by choosing a specific value for $t$ such that $P_{FA} = 0.001$. The region labeled in black is the region where pixel values are classified as background. The white is the region where pixel values are classified as foreground objects.

(a)                                                                (b)
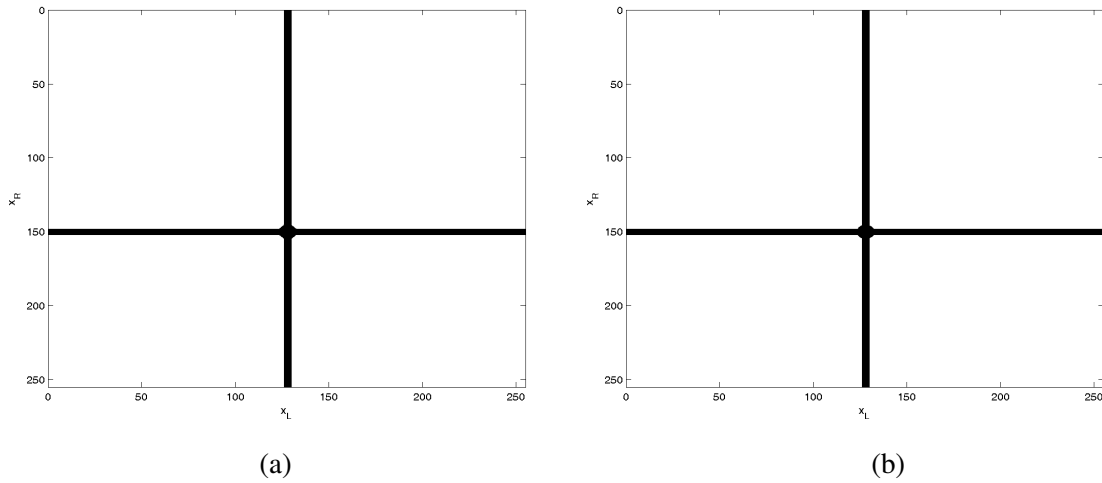
Figure 4.2: ((**a**)) The region obtained from the likelihood function of figure 4.1 after using the threshold $t = 2.1002e - 006$. This region will produce a false alarm rate $P_{FA} = 0.001$ caused by pixels with intensity close to the background distribution. ((**b**)) The matching region obtained using the simple detection method.



(a)                                                                (b)

Figure 4.3: ((**a**)) A zoom view of the region in figure 4.2((**a**)). ((**b**)) A zoom view of the region in figure 4.2((**b**)).

Figure 4.4: **(a)** ROC curves for the two Gaussian case using different $P(B_P)$ values. The data cursers indicate the expected $P_D$ for each curve for a $P_{FA}$ value of 0.001. **(b)** ROC curves for both single and two Gaussian cases with reflective background using different $P(B_P)$ values. The curves were all generated with the value $\sigma = 1$.
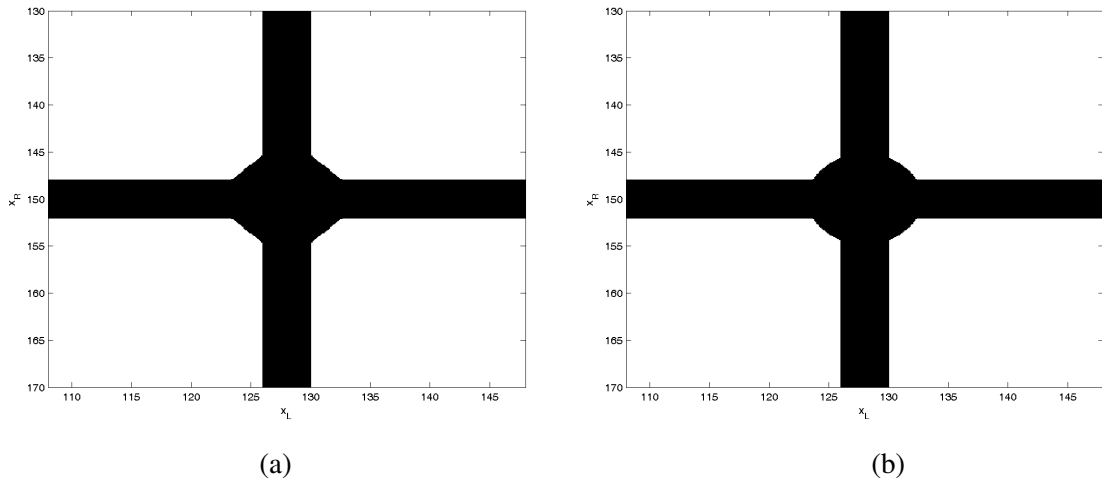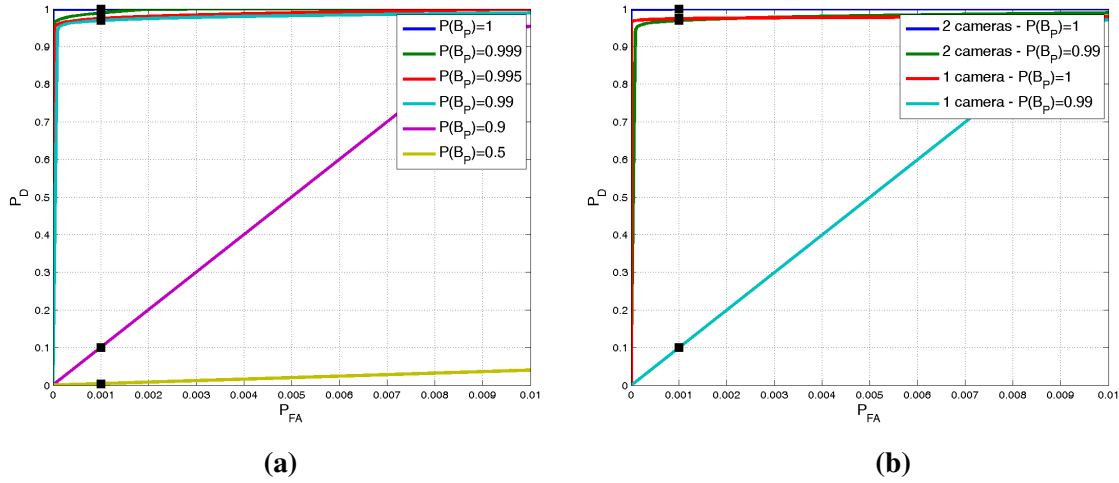
As before, an analytical solution for equation 3.4 when substituting it with equations 4.1 and 4.1 is not possible. Instead, we sample the function and generate the ROC curves depicted in figures 4.4**(a)** and **(b)**.

In figure 4.4**(a)** we depict the ROC curves computed for different values of $P(B_P)$ where the $\sigma$ values for both Gaussians are set to 1. We compare the performance of the single Gaussian case and the two Gaussian case. As can be seen, the performance for $P(B_P) \geq 0.99$ is very good. It is interesting to observe the case where $P(B_P) = 1$. In this case there are no transient background reflections and the classification method converges back to the Mahalanobis distance in the 2D case, which is basically the classification method in [Akman et al., 2008]. As can be seen in the ROC curve, the performance for $P(B_P) = 1$ is almost perfect. However, for the same parameters, the single camera performance is not as good. This demonstrates that using two cameras improves change detection for objects with low SNR even when no reflective surfaces are present. For smaller values of $P(B_P)$, performance degrades substantially for the two-camera case, even to the point of yielding useless results. This demonstrates the limitations of our method when the probability of transient background is high. We believe, in this case, that additional cameras can lead to improved performance.

### 4.2.1   Simplified Classification Method

In the previous section we developed the classification rule for the two image case. Ideally, a closed form solution to $t$ is desired, given the acceptable false alarm rate, $\alpha$. However, since such a solution is unknown, we computed $t$ numerically by sampling the likelihood ratio function and generated ROC curves that exhibited superior performance over the corresponding single image case. In this section we propose an easier-to-compute approximation of the classification region presented in Figure 4.3**(a)**. The region, resembling an ellipse, is centered at $\mu_L$ and $\mu_R$ and holds most of the density of the background density function. This component is weighted with $P(B_P) \cdot P(B_P)$, which causes it to have high density values compared to the other regions. The two rectangular regions, one centered at $\mu_L$ and the other at $\mu_R$, cover cases that occur with probability $P(B_P) \cdot (1 - P(B_P))$, where one pixel value originates from the permanent background and the other from the transient background.

Using two thresholds $t_P$ and $t_T$, we define our simplified detection region as follows. The ellipse shape is defined by:

$$R_P\left(t_P\right) = \left\{ x = (x_L, x_R) \left| \left( \left(\frac{x_L - \mu_L}{\sigma_L}\right)^2 + \left(\frac{x_R - \mu_R}{\sigma_R}\right)^2 \right) > t_P \right. \right\}. \tag{4.4}$$

The two rectangles are defined by:

$$R_T^L\left(t_T\right) = \left\{ x = (x_L, x_R) \left| \left| \frac{x_L - \mu_L}{\sigma_L} \right| > t_T \right. \right\} \tag{4.5}$$

$$R_T^R\left(t_T\right) = \left\{ x = (x_L, x_R) \left| \left| \frac{x_R - \mu_R}{\sigma_R} \right| > t_T \right. \right\}. \tag{4.6}$$

Finally, R is defined using two thresholds rather than one:

$$R\left(t_P, t_T\right) = R_P\left(t_P\right) \cap R_T^L\left(t_T\right) \cap R_T^R\left(t_T\right). \tag{4.7}$$

In order to set thresholds $t_P$ and $t_T$, we note the following:

- If both $x_L$ and $x_R$ originate from the permanent background, then they each should have a normal distribution. Thus, the expression $\left( \left(\frac{x_L - \mu_L}{\sigma_L}\right)^2 + \left(\frac{x_R - \mu_R}{\sigma_R}\right)^2 \right)$ is distributed $\chi^2$ with 2 degrees

of freedom. The $\chi^2$ CDF then defines the probability $\left( \left( \frac{x_L - \mu_L}{\sigma_L} \right)^2 + \left( \frac{x_R - \mu_R}{\sigma_R} \right)^2 \right) < t_P$.

- The expression $\left| \frac{x - \mu}{\sigma} \right|$ is normally distributed $N(0, 1)$, and thus the standard normal CDF defines the probability that $\left| \frac{x - \mu}{\sigma} \right| < t_T$.

We define $\alpha_P$ to be the probability that $\left( \left( \frac{x_L - \mu_L}{\sigma_L} \right)^2 + \left( \frac{x_R - \mu_R}{\sigma_R} \right)^2 \right) < t_P$, and we define $\alpha_T$ to be the probability that $\left| \frac{x - \mu}{\sigma} \right| < t_T$. Then, the problem of selecting $t_P$ and $t_T$ becomes one of selecting an appropriate probability of false alarm $\alpha_P$ and $\alpha_T$ according to the $\chi^2$ CDF and normal CDF, respectively.

As in Chapter 3, instead of integrating over the region classified as foreground, $R(t_P, t_T)$, we integrate over the region classified as background, that is, its compliment $\overline{R(t_P, t_T)}$. As in chapter 3, we would like the integration of $P(x|B)$ over the region $\overline{R(t_P, t_T)}$ to produce the probability of a true negative, which should be $1 - \alpha$.

Substituting equations 4.1 and 4.7 into the above integral, we get the following equation:

$$
\begin{aligned}
\int\limits_{x \notin R(t_P, t_T)} P\left( x \,|B; \mu_L, \sigma_L, \mu_R, \sigma_R \right) dx &= \int\limits_{x \notin R(t_P, t_T)} P_L\left( x_L \,|B; \mu_L, \sigma_L \right) \cdot P_R\left( x_R |B; \mu_R, \sigma_R \right) dx \\
&= P(B_P)^2 \cdot P_{PP} + P(B_P)(1 - P(B_P)) \cdot P_{PT} \\
&\quad + P(B_P)(1 - P(B_P)) \cdot P_{TP} + (1 - P(B_P))^2 \cdot P_{TT} \\
&= 1 - \alpha.
\end{aligned}
$$

$$(4.8)$$

The expression in equation 4.8 is composed of a weighted sum of 4 integrals. The first integral, $P_{PP}$ is defined by:

$$
P_{PP} = \int\limits_{x \in \overline{R(t_P, t_T)}} \left[ \frac{1}{\sqrt{2\pi}\sigma_L} e^{-\frac{1}{2}\left( \frac{x_L - \mu_L}{\sigma_L} \right)^2} \cdot \frac{1}{\sqrt{2\pi}\sigma_R} e^{-\frac{1}{2}\left( \frac{x_R - \mu_R}{\sigma_R} \right)^2} \right] dx, \tag{4.9}
$$

where weight $P(B_P)^2$ models the probability that both pixels originate from the permanent back-

ground. The second two integrals, $P_{PT}$ and $P_{TP}$ are defined by:

$$P_{PT} = \int\limits_{x \in \overline{R(t_P, t_T)}} \left[ \frac{1}{255\sqrt{2\pi}\sigma_L} e^{-\frac{1}{2}\left(\frac{x_L - \mu_L}{\sigma_L}\right)^2} \right] dx \tag{4.10}$$

$$P_{TP} = \int\limits_{x \in \overline{R(t_P, t_T)}} \left[ \frac{1}{255\sqrt{2\pi}\sigma_R} e^{-\frac{1}{2}\left(\frac{x_R - \mu_R}{\sigma_R}\right)^2} \right] dx, \tag{4.11}$$

where weight $P(B_P)(1 - P(B_P))$ models the probability that one pixel originates from the permanent background and the other from the transient background. The last integral $P_{TT}$, is defined by:

$$P_{TT} = \int\limits_{x \in \overline{R(t_P, t_T)}} \frac{1}{255^2} \, dx, \tag{4.12}$$

where weight $(1 - P(B_P))^2$ models the probability that the two pixels originate from the transient background.

The larger the area of $R(t_P, t_T)$ is, the higher the detection rate is. Hence, we wish the area of the region that is classified as background $\overline{R(t_P, t_T)}$ to be as small as possible while having the integral in equation 4.8 satisfy the acceptable false alarm rate $\alpha$. We next describe how these two objectives are met by approximating the thresholds. As stated earlier, equation 4.8 is composed of a weighted sum of 4 integrals: $P_{PP}$, $P_{PT}$, $P_{TP}$, and $P_{TT}$. The first component, $P_{PP}$, which has the largest weight, $P(B_P)^2$, is also very dense and an integral of a relatively small region will produce very high probability. Selecting the threshold $t_P$ according to the CDF of the $\chi^2$ distribution such that the probability will be 0.99999 will produce a fixed threshold $t_P = 23.0259$. We do not choose 1 because this will produce an infinite region. Note that we have only considered the region $R_P$ defined in equation 4.4. Integrating over the entire region $\overline{R(t_P, t_T)}$ should produce an even higher value $P_{PP}$ closer to 1. We will, in this approximation, assume that $P_{PP} = 1$.

Since we assume that the region $\overline{R(t_P, t_T)}$ will be as small as possible, we expect that $(1 - P(B_P))^2 \cdot P_{TT}$ will be very small. Thus, we neglect it and set it as 0. We also assume that the region $\overline{R(t_P, t_T)}$ is symmetric and that $P_{PT} = P_{TP}$. Thus, plugging $P_{PP} = 1$ and $(1 - P(B_P))^2 \cdot P_{TT} = 0$ and $P_{PT} = P_{TP}$ into 4.8, we get that:
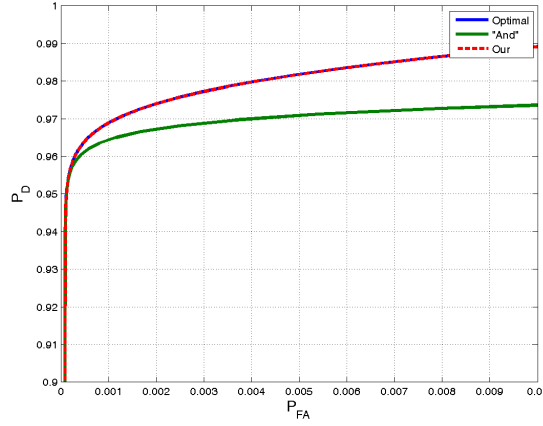
Figure 4.5: ROC curve comparing the performance of our method against the optimal and naive methods. The blue shows the optimal likelihood ratio based method. The green is a naive method where we simply perform a logical AND operation between the results of independent detections in each image. The red is our proposed method.

$$P(B_P)^2 + 2P(B_P)(1 - P(B_P)) \cdot P_{PT} = 1 - \alpha, \tag{4.13}$$

and solve for $P_{PT}$:

$$P_{PT} = \frac{1 - \alpha - P(B_P)^2}{2P(B_P)(1 - P(B_P))}. \tag{4.14}$$

Using the normal CDF, we may obtain the threshold $t_T$, which satisfies $P_{PT}$.

Figure 4.2(**b**) shows the detection region of the simplified classification method. It can be seen that this region is very similar in shape to that of the region in Figure 4.2(**a**). In Figure 4.3 (**a**) and (**b**) the zoomed view of same regions are presented. Figure 4.5 shows ROC curves created for each of the following three methods:

- Using the optimal detection region as defined by equation 4.3.

- Using the naive method (as in [Lanza et al., 2007]) for which a logical AND operation is performed between the results of a change detection performed on each image separately.

- Our simplified detection method.

**31**

As can be seen, the performance of our simplified method is almost identical to that of the optimal one, which is better than the naive method.

## 4.3   Summary of Detection From Two Cameras

In this chapter we showed how using images of the same scene obtained from two cameras from different viewpoints substantially improves the change detection performance compared to the single view case, when reflective surfaces appear in the background. We analyzed the detection performance at pixel level, assuming a pixel-to-pixel correspondence between the two images. We further showed how to set the threshold. The proposed detection is applicable only to those corresponding pixels which consists of foreground objects.

# Chapter 5

# Detection of Foreground Objects From a Pair of Images

In this section we present our method for detecting objects on a possibly reflective planar surface. Thus, the bottom part of the objects are located directly on the surface but other parts might be above the ground and can also be reflected from the surface, as can be seen in Figure 5.1 **(a)** and **(b)**. We assume that two cameras monitor an overlapping region. We also assume the availability of a homography transformation that correlates each pixel of the ground plane in one image to the corresponding pixel in the other.

Our method combines the detection of objects using a single image change detection algorithm with the detection of objects using the more robust image pair change detection algorithm. The latter algorithm only detects foreground objects located on the ground plane. Combining the two methods allows us to detect low SNR objects, while discarding reflections and noise but still preserving the detection of foreground objects above the ground plane. Our method consists of four steps:

- A single image change detection algorithm. The goal of this algorithm is to detect changes due to large objects, not necessarily located on the ground plane.

- An image pair change detection algorithm. The goal of this algorithm is to detect changes on the ground plane, including changes due to objects with low SNR that are not due to reflections.

- A connected component labeling process combining the results of the above two steps in order to remove reflections of objects that are not located on the ground, as well as detect small objects with low SNR on the ground surface.

- A post-processing step to remove reflections of large objects from the detected blob.

We next describe each of these steps.

## 5.1    A Single Image Change Detection Algorithm

In general, our method can use any existing single image change detection algorithm that produces a binary image of the changed pixels. Here we will use the simple single Gaussian method described in section 3.2. The background at the pixel level is defined independently for each camera. The background model of each pixel is updated when a new frame is captured in order to deal with slow background changes, using a method similar to that used in [Wren et al., 1996, Stauffer and Grimson, 1999]. That is, $\mu$ and $\sigma$ at time $t$ are updated using the intensity value, $x$, at time $t$, and the previous values, $\mu_{t-1}$ and $\sigma_{t-1}$ :

$$\mu_t \quad = \quad \alpha x_t + (1 - \alpha) \, \mu_{t-1}$$

$$(5.1)$$

$$\sigma_t^2 \quad = \quad \alpha(x_t - \mu_t)^2 + (1 - \alpha) \, \sigma_{t-1}^2.$$

The Mahalanobis distance for each pixel from the background model is computed and a threshold is applied. The threshold is chosen according to section 3.2 such that an expected false alarm rate will be met, using the standard normal CDF. The output is a binary change image.

The result of this part of the algorithm is a binary map with the foreground blobs, as can be seen, for example, in Figure 5.1 **(c)** and **(d)**. It is important to note that any other change detection algorithm could be used up to this point. The new challenge we address in this thesis is the detection of foreground objects while discarding reflected ones using our image pair change detection method.

## 5.2    An Image Pair Change Detection Algorithm

In 4.2.1 we defined a method for detecting pixels which originate from foreground objects using two views. This method is applicable only if a correspondence is available. To compute a correspondence, a homography transformation [Faugeras, 1993] is used to align the projection of the ground plane in the two images. The homography can be computed offline using corresponding points in both background images. Applying the homography transformation to one of the images aligns each pixel of the ground plane to its corresponding pixel in the other image. Other pixels are aligned to arbitrary pixels in the other image. The two pixels in the arbitrary pair cannot both be expected to be foreground objects, nor can they both be expected to be reflected objects.

We use the method presented in the previous section to define and learn the background (Eq. 5.1). The Mahalanobis distance of the foreground image from the background image is transformed to the coordinates of the reference image. The method defined in section 4.2.1 (Eq. 4.7) is used to classify the pixels in the reference image to foreground and background using the corresponding pixel defined by the alignment. The foreground pixels on the ground will be aligned and hence detected as foreground. Since our method deals with very small objects as well, we correct for small misalignments of the transformation by applying a 3x3 Max filter to the transformed Mahalanobis distance image. This allows for misalignments of up to 1 pixel.

Pixels that are projections of objects above the ground or that belong to reflected objects are not aligned by the homography. Hence, they will not be detected as foreground. However, false positive foreground detections may occur when reflections of different objects are aligned. This has low probability when a wide baseline setup of the cameras is used, and the foreground objects are sparse. However, this may indeed be a limitation of our method. To mitigate this limitation, we apply an opening morphological operation on the binary foreground image using a 1x2 structuring element. This will eliminate tiny blobs which are most likely due to noise. Figure 5.1(**f**) shows the result of the foreground detection.

## 5.3 Combining Image Pair and Single Foreground Images

According to the analysis in chapter 4, we expect the foreground map produced in the previous section, using a pair of images, to perform much better than the method which uses only a single image. However, since the latter method only detects parts of the object that are on the ground plane, the foreground image will not reflect the true shape and size of the foreground object's blob. On the other hand, the single image method is expected to produce a foreground image which does reflect the true shape and size of the blob, but is expected to result in more false detections. It might also miss very small objects or objects with low SNR. We combine the foreground images by performing a logical OR operation between the pixels of each image followed by a morphological closing operation using a 3x3 structured element. This should close small holes and smooth the binary result. A labeling algorithm is then run on the binary output image, producing a list of connected component blobs. These blobs are then filtered out if their size is too small (5 pixels) or if they do not consist of at least one pixel that belongs to the foreground produced by the image pair algorithm. This process ensures that only objects detected by the more powerful image pair algorithm remain. It also ensures that small blobs that are due to noise are removed as well.

## 5.4 Self Reflection Removal Process

In the process defined in the previous section, blobs are removed if they do not overlap with the foreground detections produced by the image pair detection process. As a result, blobs originating from reflections of objects that are not located on the ground plane are removed. The remaining blobs may consist of both the foreground objects and their self reflections from the ground plane, in particular when the object is not flat (see the person in Figure 5.1). To remove these self reflections, we use the following observation, assuming we have a mapping between corresponding foreground blobs in each image. After the homography transformation that aligns the ground plane of the two images is applied, the overlapping area between the blobs in both images should always be located on the ground plane (See [Khan and Shah, 2006] Propositions 1 and 2). Hence, the portion of the blobs above the intersecting area is part of the foreground object while those below the intersecting area are self reflections.

Thus, a simple way to take advantage of this fact would be to remove all blob pixels which are located underneath the intersecting area.

Ideally, there should be a one-to-one mapping between corresponding foreground blobs in each image view. However, due to noise and other false detections, it is possible that a blob in one image might have two or more blobs in the second image that overlap it. In this case, the foreground object mapping is not uniquely defined. The larger the blob of the foreground object in the image is, the more likely it is to occur (because more pixels might overlap). Thus, for small objects we can assume that only a single blob in the corresponding image will overlap, whereas for large objects, there might be more than just one. However, out of all the corresponding blobs in the second image, we can assume that there will be a single, large, corresponding blob (this is the true foreground object blob) and the rest of the corresponding blobs (which are due to noise) will be smaller.

This leads us to the following greedy matching algorithm:

1. Perform the detection algorithm presented in the previous sections on both images and produce foreground blobs.

2. Sort all the blobs in the first image by size in descending order (larger blobs come first).

3. For each blob in the sorted list in the first image, select the largest overlapping blob from the second image as the corresponding foreground blob.

4. Discard unmatched blobs from both first and second images.

Figure 5.1(**f**) shows the final result after removing self reflections. It can be seen that the person's self reflections are removed, but shadows are not.

Algorithms 5.1 and 5.2 provide pseudo-code implementation of the ideas discussed in the previous sections.
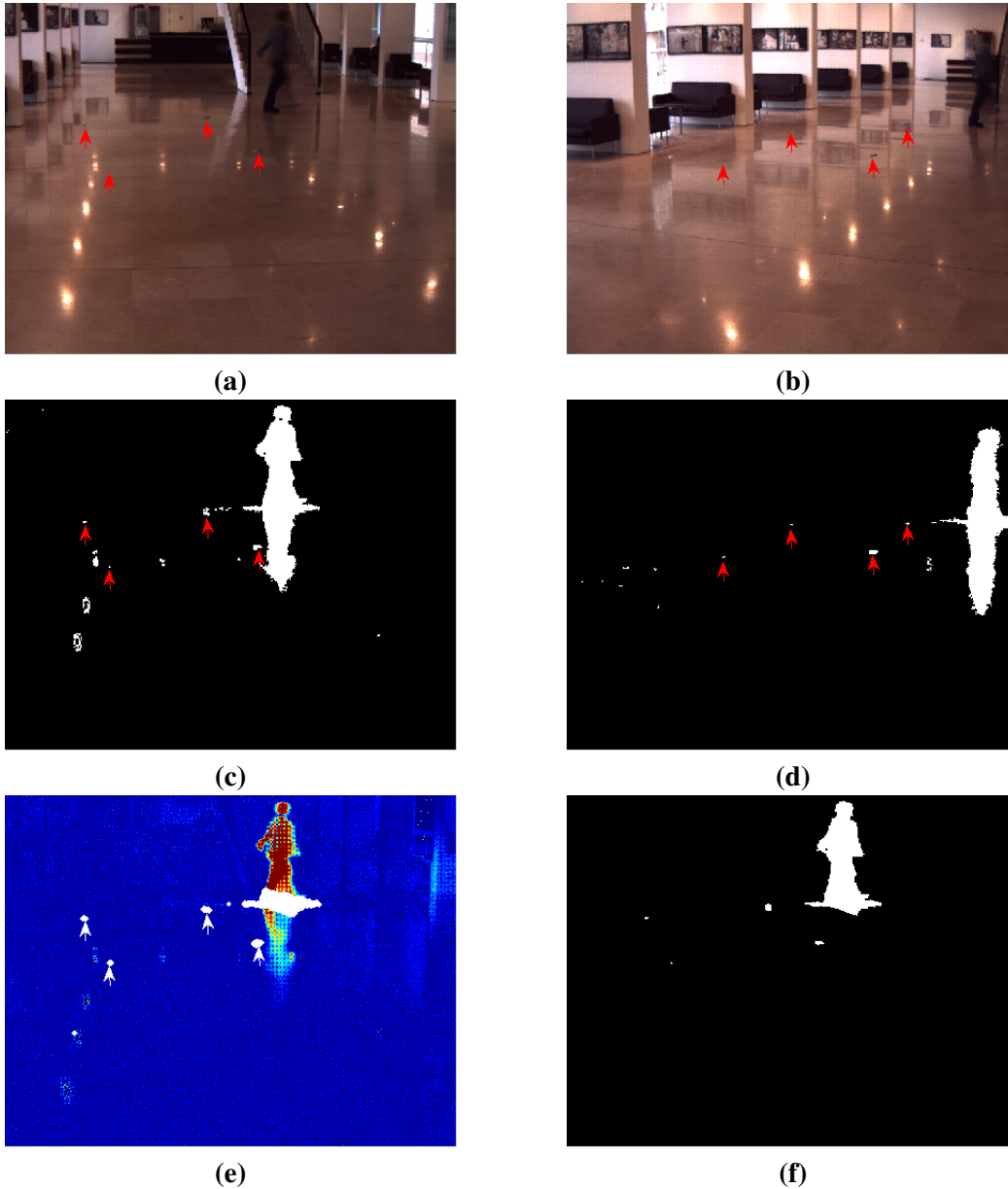
(a)

(b)

(c)

(d)

(e)

(f)

Figure 5.1: **(a)** An image containing 4 very small objects (with red arrows pointing to their location) and a large foreground object (a person) walking in the scene. The person's self reflection can be seen on the floor too. **(b)** The same scene as **(a)** as seen by the second camera. **(c)** The result of a (simple) single image change detection algorithm performed on **(a)**. **(d)** The result of a (simple) single image change detection algorithm performed on **(b)**. **(e)** The normalized difference image between **(a)** and its background model, superimposed with foreground pixels detected by the two image detection algorithms. **(f)** The final result of our method.

**Algorithm 5.1** Algorithm for detecting objects using two cameras.

$L \leftarrow$ ***TwoCameraChangeDetection***$(I_L, I_R)$

$H$ - a 3x3 projective transformation matrix between the left image and the right image.

$\mu_L$ - The left background mean image.

$\sigma_L$ - The left background standard deviation image.

$\mu_R$ - The right background mean image.

$\sigma_R$ - The right background standard deviation image.

1:  $D_L \leftarrow \left| \frac{I_L - \mu_L}{\sigma_L} \right|$

2:  $D_R \leftarrow \left| \frac{I_R - \mu_R}{\sigma_R} \right|$

3:  $L_L \leftarrow$ *ChangeDetection*$(D_L, D_R, H)$

4:  $L_R \leftarrow$ *ChangeDetection*$(D_R, D_L, H^-1)$

5:  $L'_R \leftarrow$ *NearestNeighborInterpolation*$(L_R, H)$

6:  $L \leftarrow \emptyset$

7:  $L_L \leftarrow$ *Sort by descending size* $(L_L)$

8:  **for all** $l \in L_L$ **do**

9:      $M \leftarrow$ *Dilation* $(l)$

10:     $S \leftarrow \{l \,|\, l \in L'_R \text{and} l \cap M \neq \emptyset\}$

11:     **if** $S \neq \emptyset$ **then**

12:         $l' \leftarrow$ largest label in $S$.

13:         $m \leftarrow l' \cap l$

14:         $m \leftarrow$ *RemoveBottom*$(l, m)$

15:         $L \leftarrow L \cup \{m\}$

16:     **end if**

17: **end for**

---

**Algorithm 5.2** Algorithm for a single image change detection.

---

$L \leftarrow \textbf{\textit{ChangeDetection}}(D_{REF}, D_{SEC}, H)$

$s$ - The minimal allowed blob size (in pixels).

 1: $D_{SEC} \leftarrow \textit{Interpolate}\ (D_{SEC}, H)$

 2: $D_{SEC} \leftarrow \textit{MaxFilter3x3}\ (D_{SEC})$

 3: $S \leftarrow \left(\left(D_{REF}^2 + D_{SEC}^2\right) > t_P\right) \& \left(D_{REF} > t_T\right) \& \left(D_{SEC} > t_T\right)$

 4: $S \leftarrow \textit{MorphOpenning}(S)$

 5: $C \leftarrow \textit{MorphClosing}(S \| D_{REF} > t_T)$

 6: $L_{temp} \leftarrow \textit{Labeling}(C)$

 7: $L \leftarrow \emptyset$

 8: **for all** $l \in L_{temp}$ **do**

 9:   **if** $|l| > s$ **then**

10:     **if** there exist any overlapping pixels between $l$ and $S$ **then**

11:       $L \leftarrow L \cup \{l\}$

12:     **end if**

13:   **end if**

14: **end for**

15: **return** $L$

---

# Chapter 6

# Experiments

To demonstrate the effectiveness of our method, we applied our detection algorithm to video sequences of different scenes containing reflective surfaces. The detection algorithm was implemented in Matlab.

In all tests we use 2 USB cameras (IDS uEye UI-1545LE-C) connected to a laptop with an Intel$^{(R)}$ Core$^{(TM)}$ i5-2520M CPU running at 2.5GHz. The camera's raw pixel output was used as the source of the change detection algorithm. Our camera's CCD uses a Bayer matrix of elements to produce the color images that are presented to the user. For change detection, it is better to use the raw output of the camera before demosaicing is performed. The process of demosaicing increase the dependencies between pixels and is non-linear in contrast to our assumption that pixel noise is independent and Gaussian. It also allows us to perform the entire process on a single plane instead of the 3 RGB planes. All though the algorithm is performed on the raw single channel images, in this Thesis we present the resulting images after demosaicing so they may be displayed in color.

In all experiments, the algorithm from chapter 5 is run on a sequence of images taken from two static cameras. The homography transformation between the two cameras was obtained by manually finding corresponding points in the two images of each scene and then applying the normalized direct linear transformation algorithm given by Hartley and Zisserman ([Hartley et al., 2003]).

We tested our method on 5 sets in 2 different environments, all of which are indoors with shiny marble flooring. In contrast to other conditions such as illumination changes and dynamic background, which are usually more challenging in outdoor conditions, indoor marble floors are the most challeng-

ing background in the case of reflections since they act as almost perfect mirrors. In the first two experiments, we tried to detect small objects, with some color similarity to the background, that were placed on the shiny marble floor. These experiments demonstrate the effectiveness of our method in dealing with low SNR and reflective surfaces. In the other 3 experiments, we placed both large and small objects within the region of interest and discarded any reflections from foreground and background objects that could be seen on the floor within the region of interest. These experiments demonstrate the effectiveness of our method in dealing with low SNR as well as large objects. Finally, we tested the performance of the method presented in [Hofmann et al., 2012] on our data sets and compared to our method.

## 6.1 Experiment 1

The following images were taken in a challenging environment where the background consists of a marble floor. An outdoor scene with moving trees, which is not in the region of interest, is reflected by the floor. Figure 6.2 **(a)** shows the pair of background images. It can be seen that each image is reflecting a different scene; hence, dense stereo matching based algorithms (e.g. [Krumm et al., 2000, Goldlucke and Magnor, 2003, Ivanov et al., 2000, Lim et al., 2005]) are not applicable.



Figure 6.1: The objects in experiment 1 (3 small coins and a small key).

In order to test our detection algorithm on objects with low SNR, we placed 4 very small objects with varying contrast relative to the floor in the overlapping fields of view of the cameras. Close-up images of these objects, which were not available to the algorithm, are presented in Figure 6.1. These objects are also seen in figure 6.2 **(a)**, where they are placed on the surface and viewed by the two cameras. To help the viewer, the object locations are marked with small red arrows. Figure 6.3 **(a)** shows the same images, but zoomed into the region of interest. The algorithm from section 5.1 was applied to these images and figure 6.2 **(b)** shows the difference between the current images with the placed objects and the background, normalized by sigma values for each pixel, that is, the value $\left|\frac{x-\mu}{\sigma}\right|$. As expected, it is very hard, when using a single image, to distinguish between the true changes due to the objects on the floor, and changes due to reflections. Figure 6.3 **(b)** shows the same difference images but zoomed into the region of interest. Figure 6.3 **(c)** shows these images superimposed with the foreground seed pixels obtained using equation 4.7 (see section 5.2). As can be seen, the "seeds" are located at the bottom of the foreground objects, where they connect to the ground plane.

Figure 6.2 **(c)** shows the results of the change detection process applied to each image separately without using the seed information. As can be seen, many other blobs were detected in addition to the blobs created by our 4 objects. It was impossible to set a threshold such that these four blobs would be eliminated without missing one of the 4 true objects. Figure 6.2 **(d)** shows the result of our method. After the seed information was applied, all the false blobs were eliminated but the true objects remained.

In the previous figures, we demonstrated our performance on a set of objects placed on the floor. In figure 6.4 we show the distribution of the normalized difference values (the value $\left|\frac{x-\mu}{\sigma}\right|$) in an image taken without any objects. Thus, the values used consist of only noise. For each possible normalized difference, we plot the percentage of the pixels in the image that have a greater difference value. We also plot the percentage of corresponding pixels whose values in both images are greater. Basically this provides us the false detection rate per threshold. As expected, using a single image would require a much higher threshold to achieve the same false alarm rate.
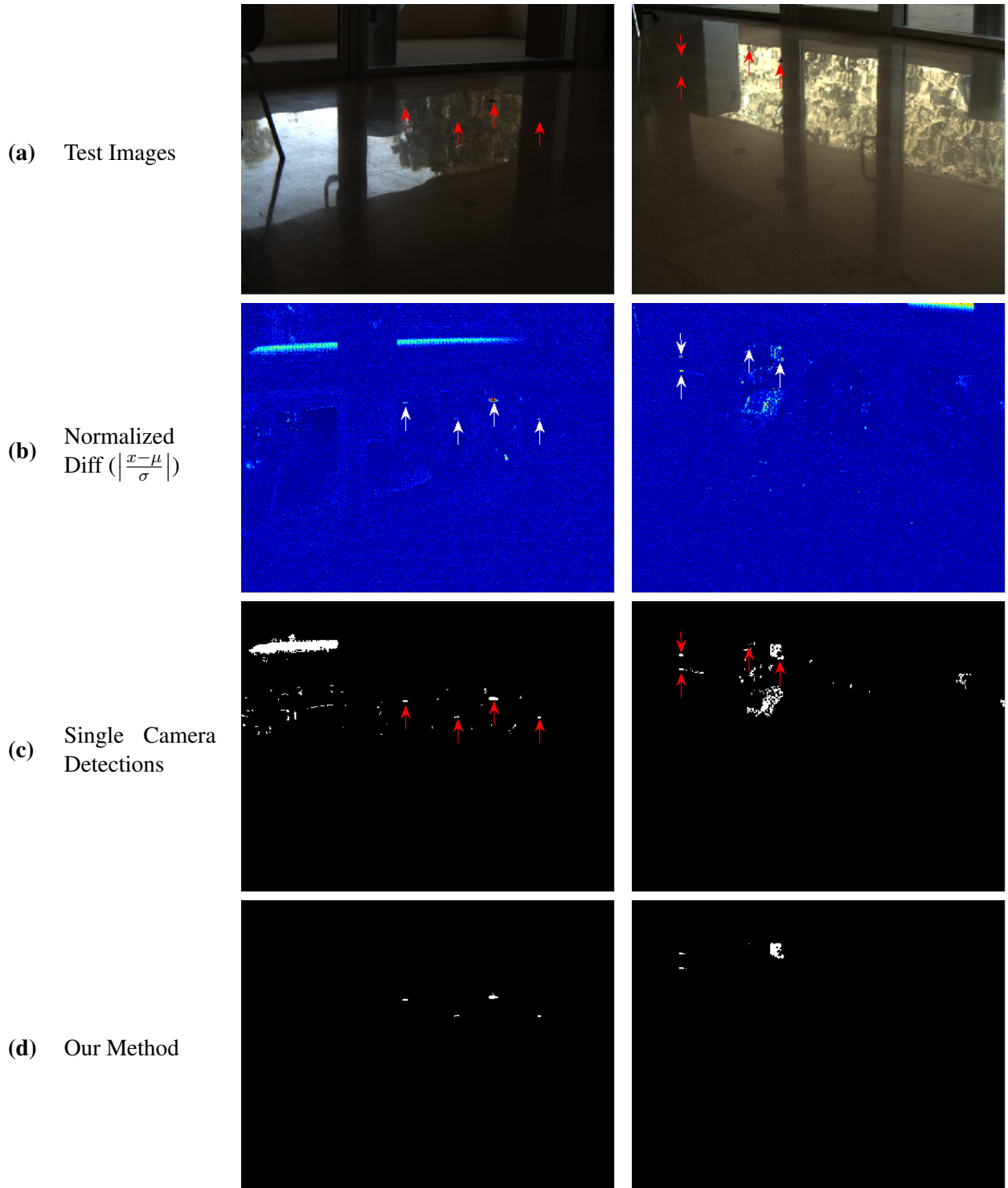
Figure 6.2: Results of experiment 1. **(a)** Test images from two cameras. **(b)** Images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Results of the change detection method obtained from each image independently. **(d)** Results of our method.

**(a)** Zoomed Test Images

**(b)** Zoomed Normalized Diff ($\left|\frac{x-\mu}{\sigma}\right|$)
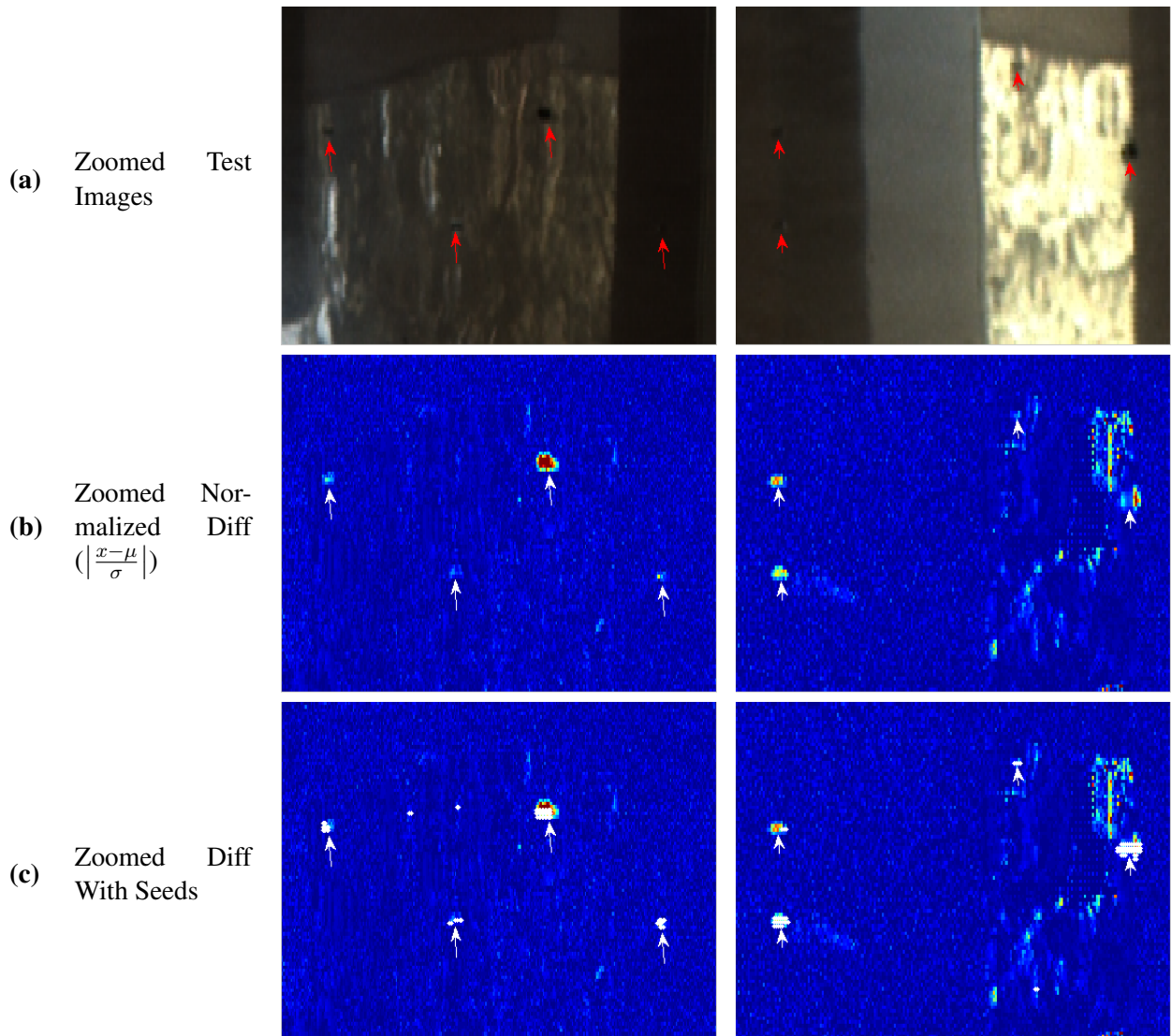
**(c)** Zoomed Diff With Seeds



Figure 6.3: Zoomed results of experiment 1. **(a)** Zoomed images of figure 6.2**(a)**. **(b)** Zoomed images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Difference images superimposed with pixels that were identified as foreground using the image pair change detection method.
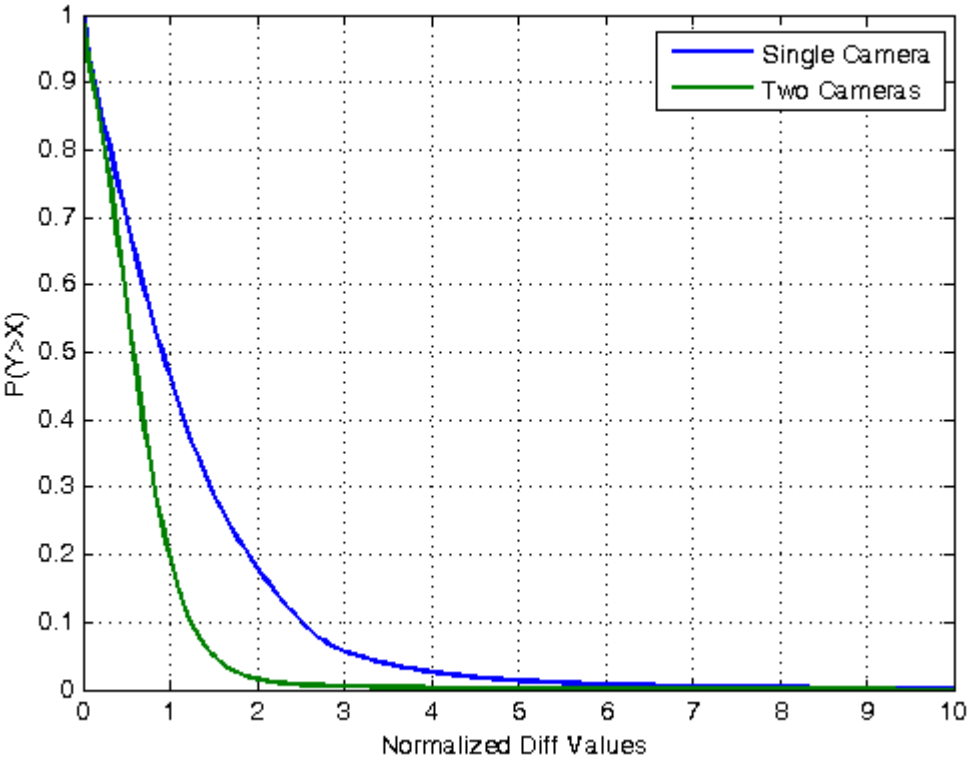
Figure 6.4: The distribution for single and two cameras from experiment 1.

## 6.2 Experiment 2

The following images were taken in a different environment than those in section 6.1. Here too the background is a marble floor upon which we placed the small objects seen in figure 6.5. (Again, these images are not available to the algorithm.) The floor in this scenario is less reflective than the one from section 6.1, but the field of view is larger and the objects placed at a greater distance from the cameras. This experiment demonstrates that despite this distance, and despite the fact that the colors of the objects are sometimes similar to the floor, our method is still able to detect these objects with no false positives. Note that as in the previous experiment, the threshold is set automatically and independently for each pixel. The figures in this section exhibit the same flow as in the previous section.



Figure 6.5: The objects in Experiment 2 (A plastic pen, a coin, a whiteboard marker, and a small key).
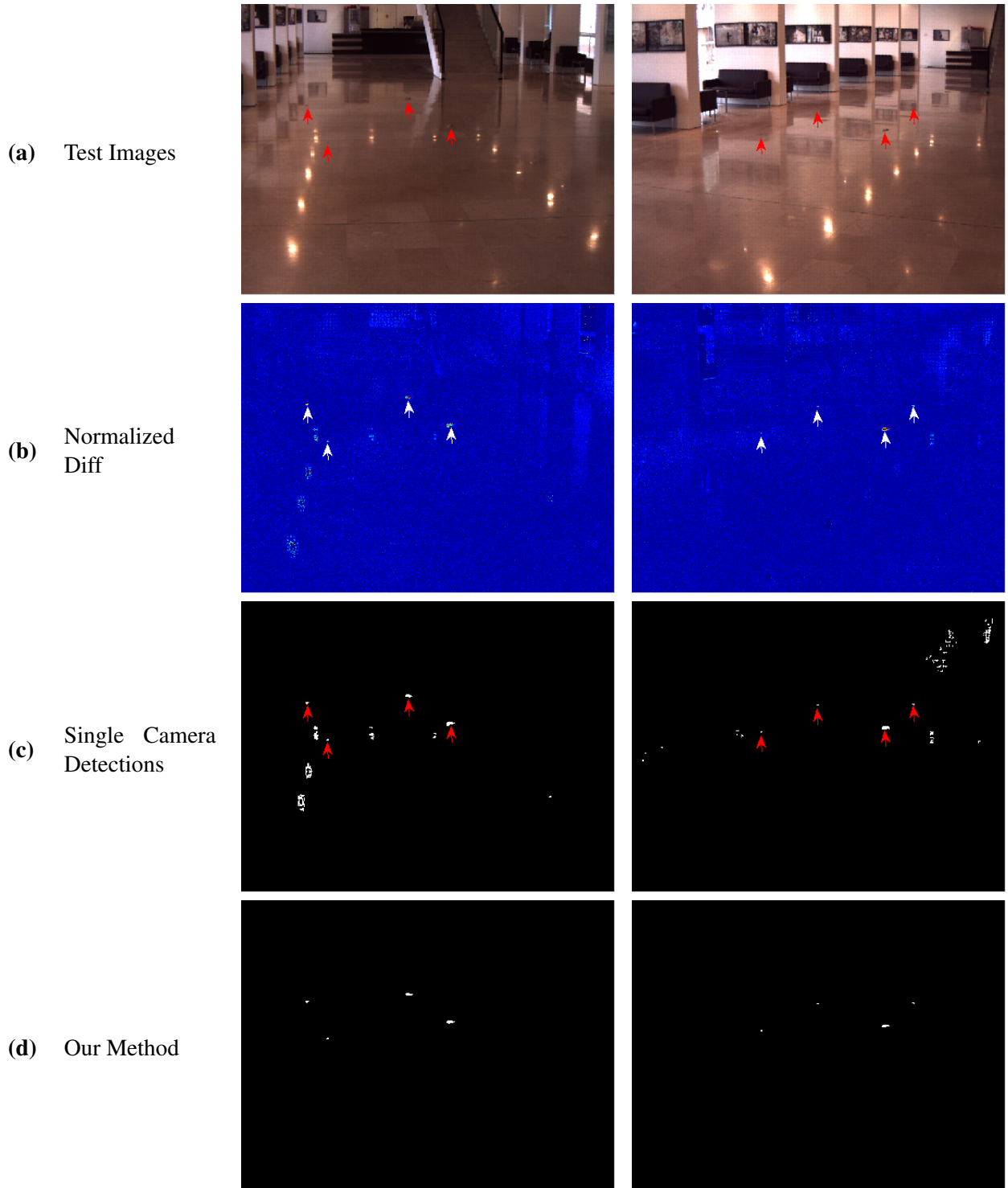
**(a)** Test Images

**(b)** Normalized Diff

**(c)** Single Camera Detections

**(d)** Our Method

Figure 6.6: Results of experiment 2. **(a)** Test images from two cameras. **(b)** Images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Results of the change detection method obtained from each image independently. **(d)** Results of our method.
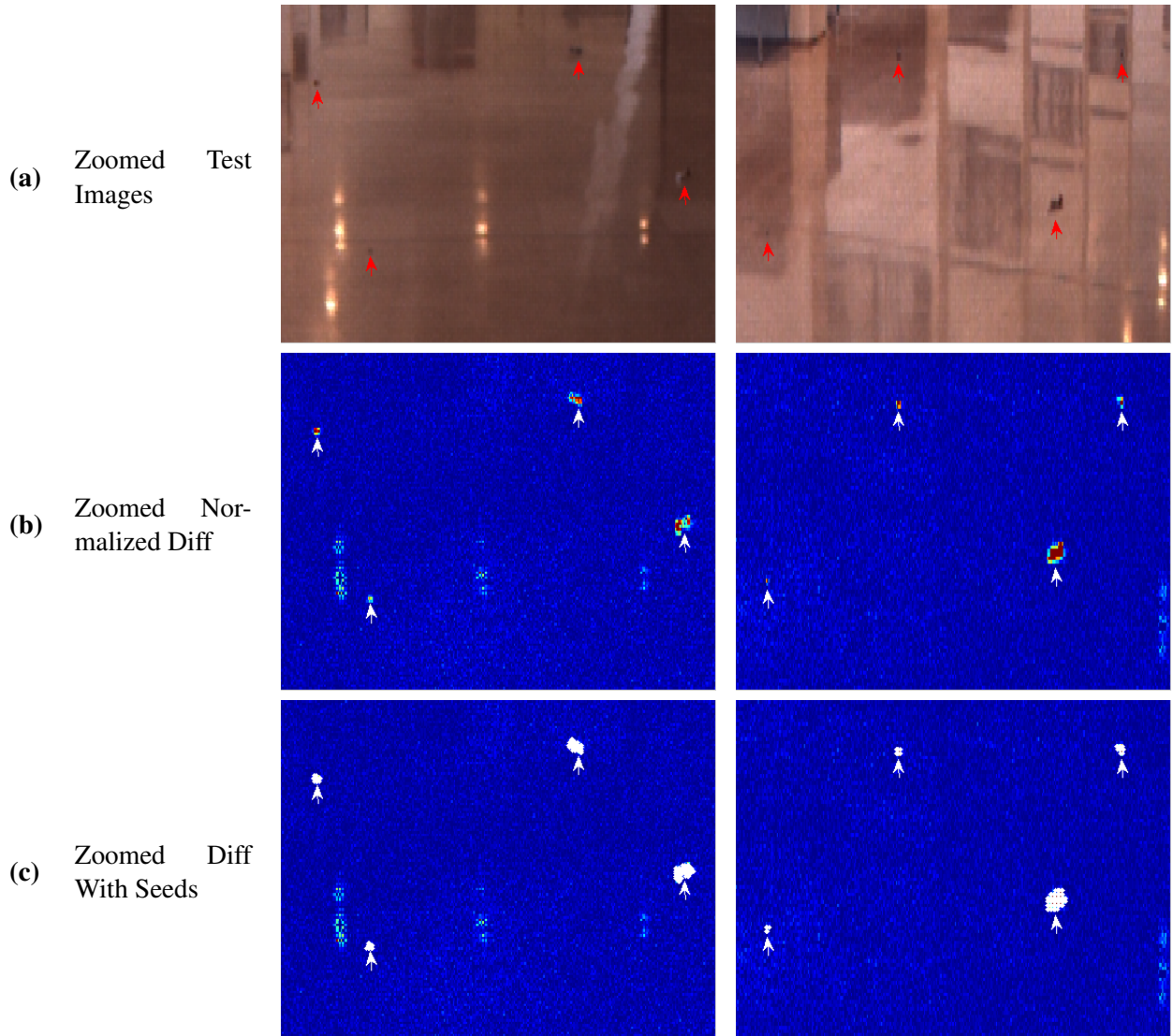
**(a)** Zoomed Test Images

**(b)** Zoomed Normalized Diff

**(c)** Zoomed Diff With Seeds

Figure 6.7: Zoomed results of experiment 2. **(a)** Zoomed images of figure 6.6**(a)**. **(b)** Zoomed images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Difference images superimposed with pixels that were identified as foreground using the image pair change detection method.

## 6.3   Experiment 3

The scene in this experiment is identical to that of experiment 2. However, alongside the small objects (coin, marker, and key), we also place a large handbag, whose image is reflected on the floor. The goal here is to detect all 4 objects but also eliminate the reflective portion of the large one. The same figures are used as in previous experiments to present the images and the results.

Due to its reflection on the floor, the handbag is reflected back to each of the cameras (6.9**(a)**). Indeed these reflections are also detected as a change from the background (Figure 6.9**(b)**). It is quite obvious that distinguishing between the true foreground and the reflective portion of the blob using a single camera would be impossible without any additional information. This is true in particular when the right image is considered. For the left image the contrast of the reflective portion of the object is less than the contrast of the foreground, which makes the difference image slightly easier to segment. However, again, this would be a post-processing step where an adaptive threshold would need to be obtained, again, using some prior constraint about the contrast of the reflection.

The leftmost object was too similar to the noise level of the right image. This can be viewed in Figure 6.10**(b)**. As a result, the seed of this object was not detected and hence the detection in the left image was discarded. In this case our method failed to recognize this tiny object.

In figure 6.9**(d)** it can be seen that the reflection of the bag on the floor was not detected as foreground, as expected.

Figure 6.8: The objects in Experiment 3 (A large handbag, a coin, A whiteboard marker, and a small key).

**(a)** Test Images

**(b)** Normalized Diff

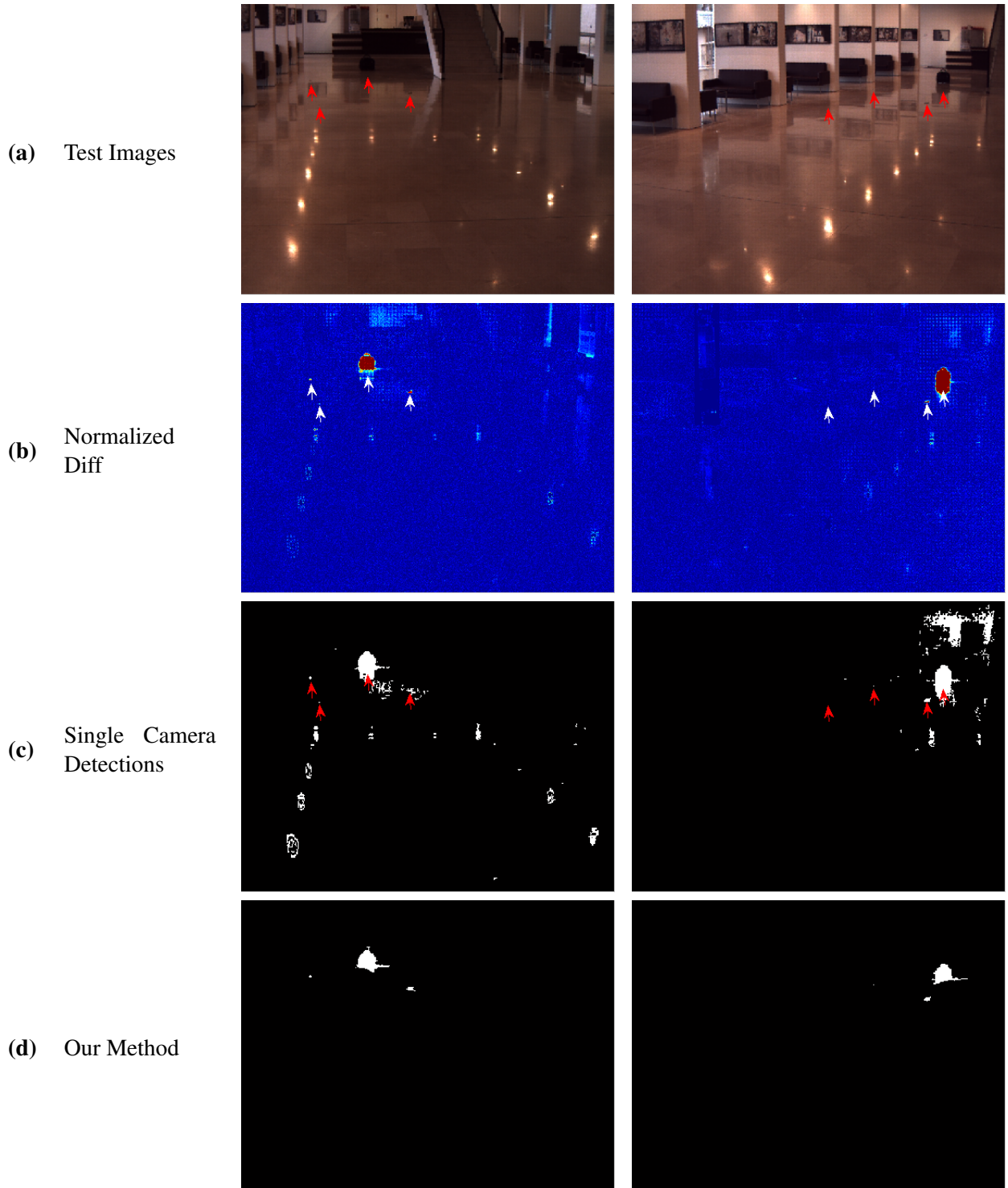**(c)** Single Camera Detections

**(d)** Our Method

Figure 6.9: Results of experiment 3. **(a)** Test images from two cameras. **(b)** Images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Results of the change detection method obtained from each image independently. **(d)** Results of our method.
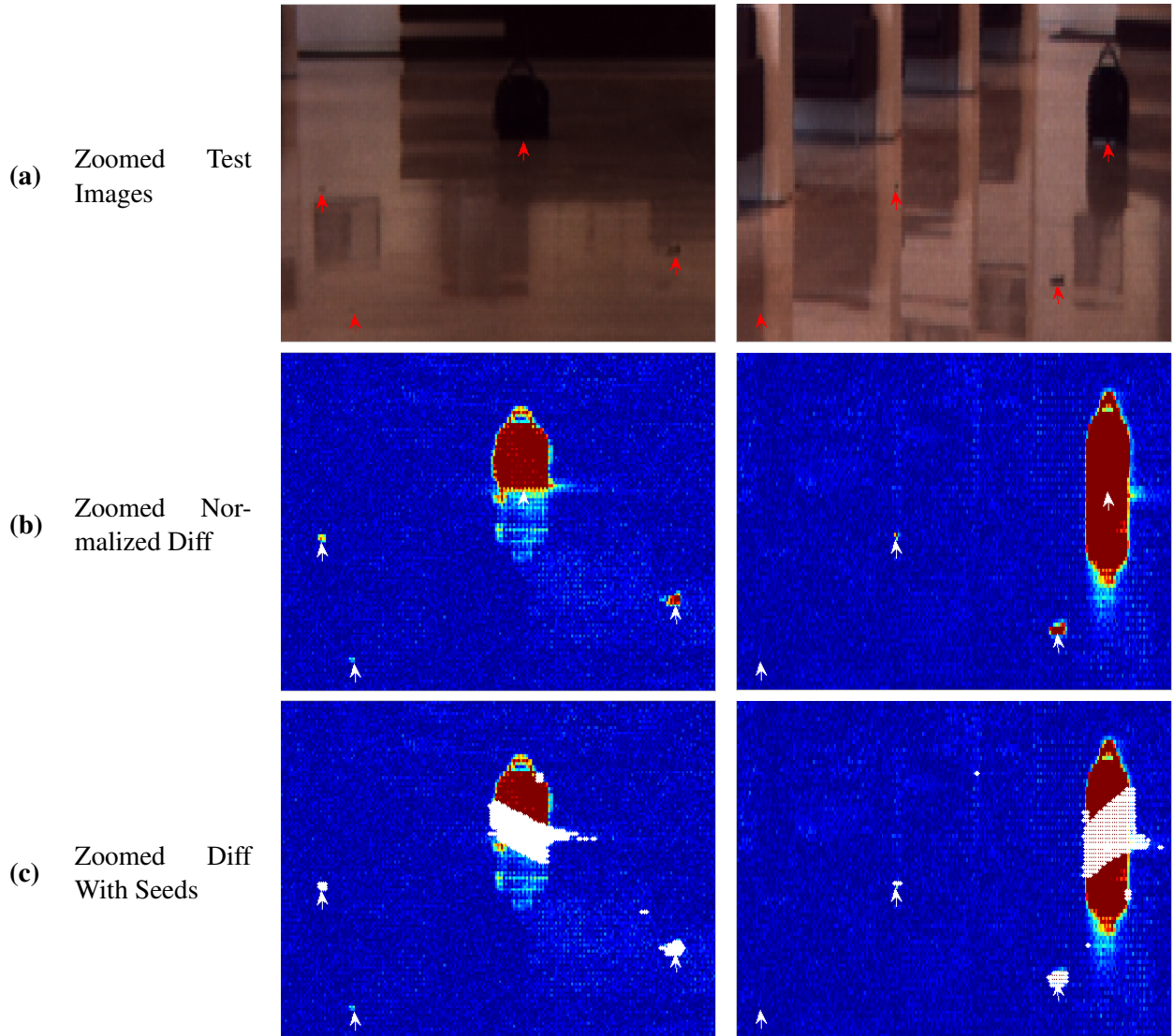
**(a)** Zoomed Test Images

**(b)** Zoomed Normalized Diff

**(c)** Zoomed Diff With Seeds

Figure 6.10: Zoomed results of experiment 3. **(a)** Zoomed images of figure 6.9**(a)**. **(b)** Zoomed images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Difference images superimposed with pixels that were identified as foreground using the image pair change detection method.

## 6.4   Experiment 4

The scene in this experiment is the same scene from experiment 2. We use the same objects as in figure 6.5. In this experiment the change consists of a person standing on the floor next to 4 small objects. The challenge here is to detect the four small objects and the person, while discarding his reflection.

In figure 6.12(**c**) it can be seen that the "seeds" are located at the bottom of the foreground objects, where they connect to the ground plane. The reflections of the person in the two cameras do not generate seed pixels and thus the reflections are not detected as foreground objects, as expected. Thus, when using only the single camera to detect the blob, the reflected part of the person on the floor is also detected, as can be seen in 6.11(**c**), while our method in figure 6.11(**d**) correctly removes the person's reflection from the blob. Note, however, that shadows produced by the person are seen by both cameras and thus not removed by our method.
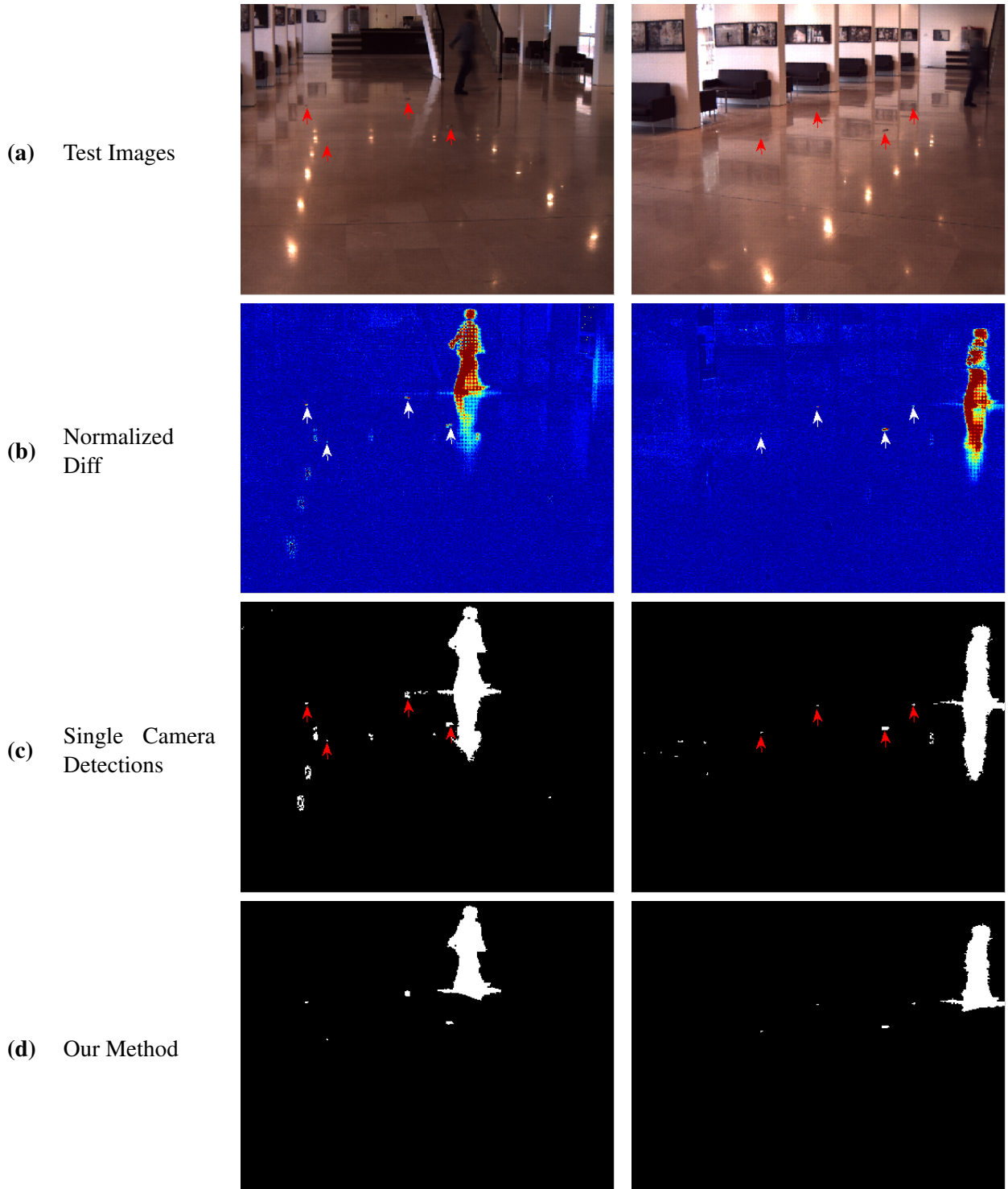
**(a)** Test Images

**(b)** Normalized Diff

**(c)** Single Camera Detections

**(d)** Our Method

Figure 6.11: Results of experiment 4. **(a)** Test images from two cameras. **(b)** Images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Results of the change detection method obtained from each image independently. **(d)** Results of our method.
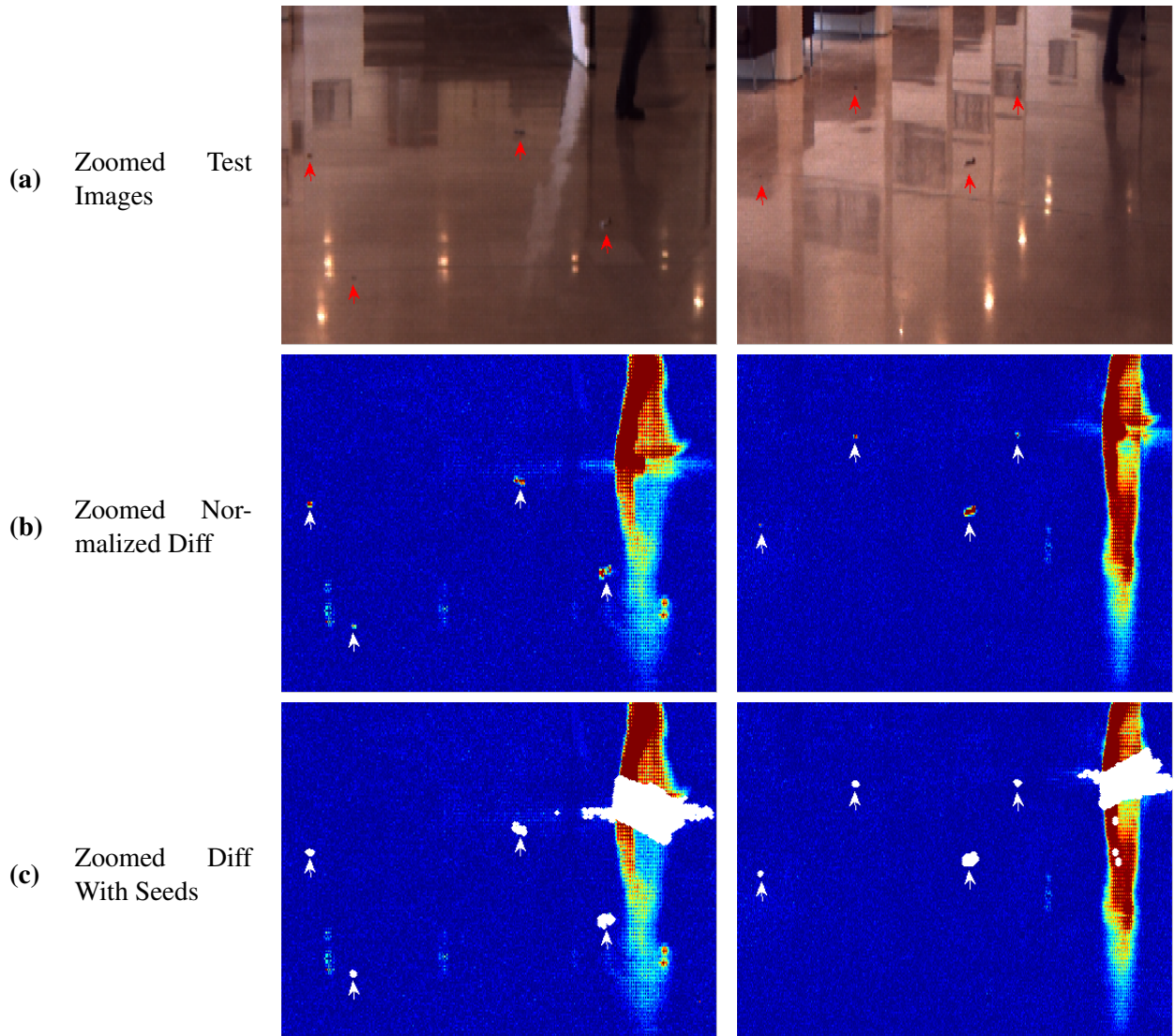
**(a)** Zoomed Test Images

**(b)** Zoomed Normalized Diff

**(c)** Zoomed Diff With Seeds

Figure 6.12: Zoomed results of experiment 4. **(a)** Zoomed images of figure 6.11**(a)**. **(b)** Zoomed images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Difference images superimposed with pixels that were identified as foreground using the image pair change detection method.

## 6.5 Experiment 5

The scene in this experiment is similar to experiment 1. In this experiment we have a person walking outside of the region of interest, i.e., a transient background object. The person is seen via a glass window and is reflected on the floor. The reflection of the person on the floor is seen by one of the cameras, which causes a false positive in a single camera change detection algorithm. The challenge here, using our method, is to identify the person's reflection as a false detection.

In the right image of Figure 6.13**(a)** a reflection of a person is seen on the floor. This produces the large change values that can be seen in the Figure 6.13**(b)** on the right. Figure 6.14**(a)** shows the difference images after a homography transformation was applied such that each pixel originating from the ground plane matches the corresponding pixel in the other image. The area with the large change values that are due to the person's reflection corresponds to an area on the floor in the other camera's FOV. However, the other camera does not see any reflections on those pixels since reflections depend on the camera location. Thus, a foreground object is not produced.

Figure 6.13**(c)**, shows the output of the single camera change detection method. It can be seen that the right camera falsely detects the person as a foreground object, while our method correctly ignores it (Figure 6.13**(d)**). However, small reflections of swaying trees outside the window produce a false foreground detection. This agrees with our analysis that for smaller values of $P(B_P)$ our method's performance degrades (section 4.2). Indeed, this is a limitation of our method. Using a background model that is more robust to background motion is expected to improve this result. Nevertheless, using the pair of cameras removes most of the false detection, in contrast to the single camera detection method.

**(a)** Test Images

**(b)** Normalized Diff

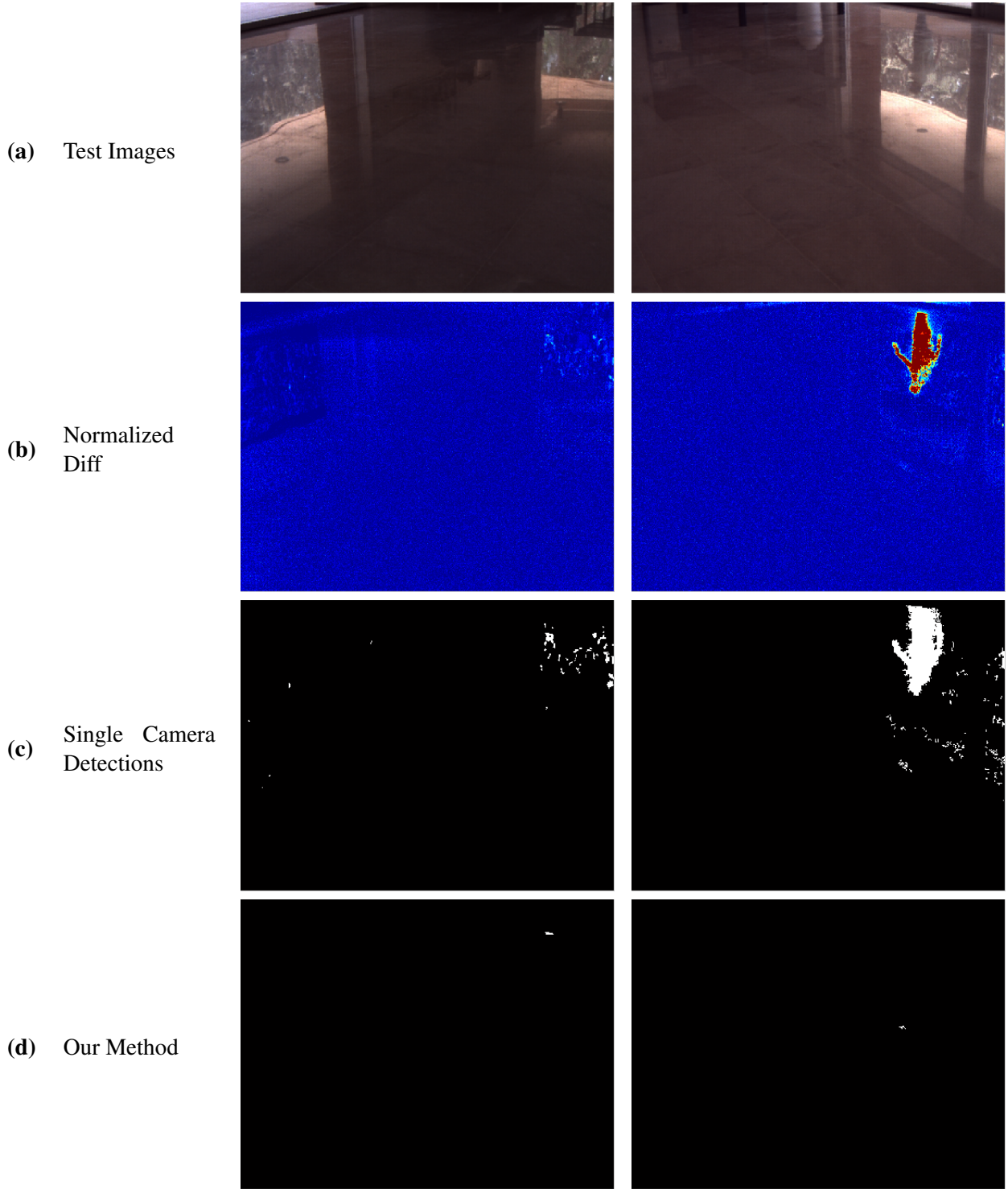**(c)** Single Camera Detections

**(d)** Our Method

Figure 6.13: Results of experiment 5 - **(a)** Test images from two cameras. **(b)** Images obtained after background subtraction normalized by the temporal variance of each pixel. **(c)** Results of the change detection method obtained from each image independently. **(d)** Results of our method.
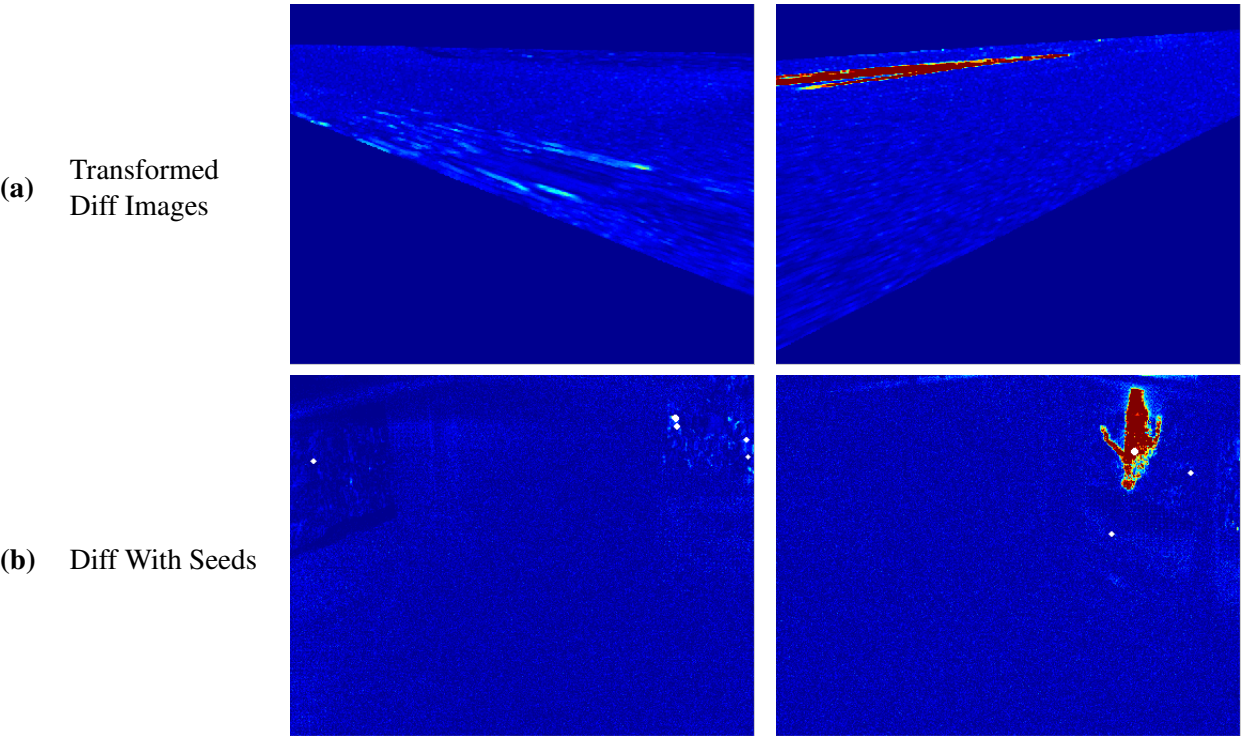
**(a)** Transformed Diff Images

**(b)** Diff With Seeds

Figure 6.14: Additional results from experiment 5. **(a)** Difference images from figure 6.13**(b)** transformed to the corresponding image coordinates. **(b)** Difference images superimposed with pixels that were identified as foreground using the image pair change detection method.

## 6.6 Experiment 6

We compare the results of our method to one of the state-of-the-art change detection algorithms, PBAS [Hofmann et al., 2012], which was ranked the highest in the CVPR 2012 change detection challenge [Goyette et al., 2012]. We tried running the PBAS method, downloaded from the author's website, using the parameters for which the method ranked highest in the challenge. However, with these parameters, the PBAS method did not handle our low SNR objects well, and many missed detections occurred. We then ran the PBAS method using two values, $R_{lower} = 7$ and $R_{lower} = 15$, which are lower than the value $R_{lower} = 18$ used in [Hofmann et al., 2012].

The results of running both the PBAS and our methods on the test data from experiments 1-5, are presented in the following five figures (6.15 - 6.19). It can be seen that the threshold value $R_{lower} = 7$ is too low and produces false positives. This is especially clear in Figure 6.15, where using $R_{lower} = 7$ produced many false positive but the four true objects were, in contrast to our method, not detected. Using $R_{lower} = 15$ reduces the false positives, but results in many miss detections instead. In figures 6.16, 6.17, and 6.18, it can be seen that for both thresholds the PBAS method falsely detects the large object reflections on the floor, as expected, since no attempt is made to discriminate between reflected and real objects. The five tests show that our method outperforms the PBAS method for the challenges considered in this thesis, with a higher detection rate and lower false positive rates. In addition, our method is able to discriminate between reflected and real foreground objects. By effectively using the information available from both cameras, our method can use a relatively simple change detection method and still out perform the PBAS method.
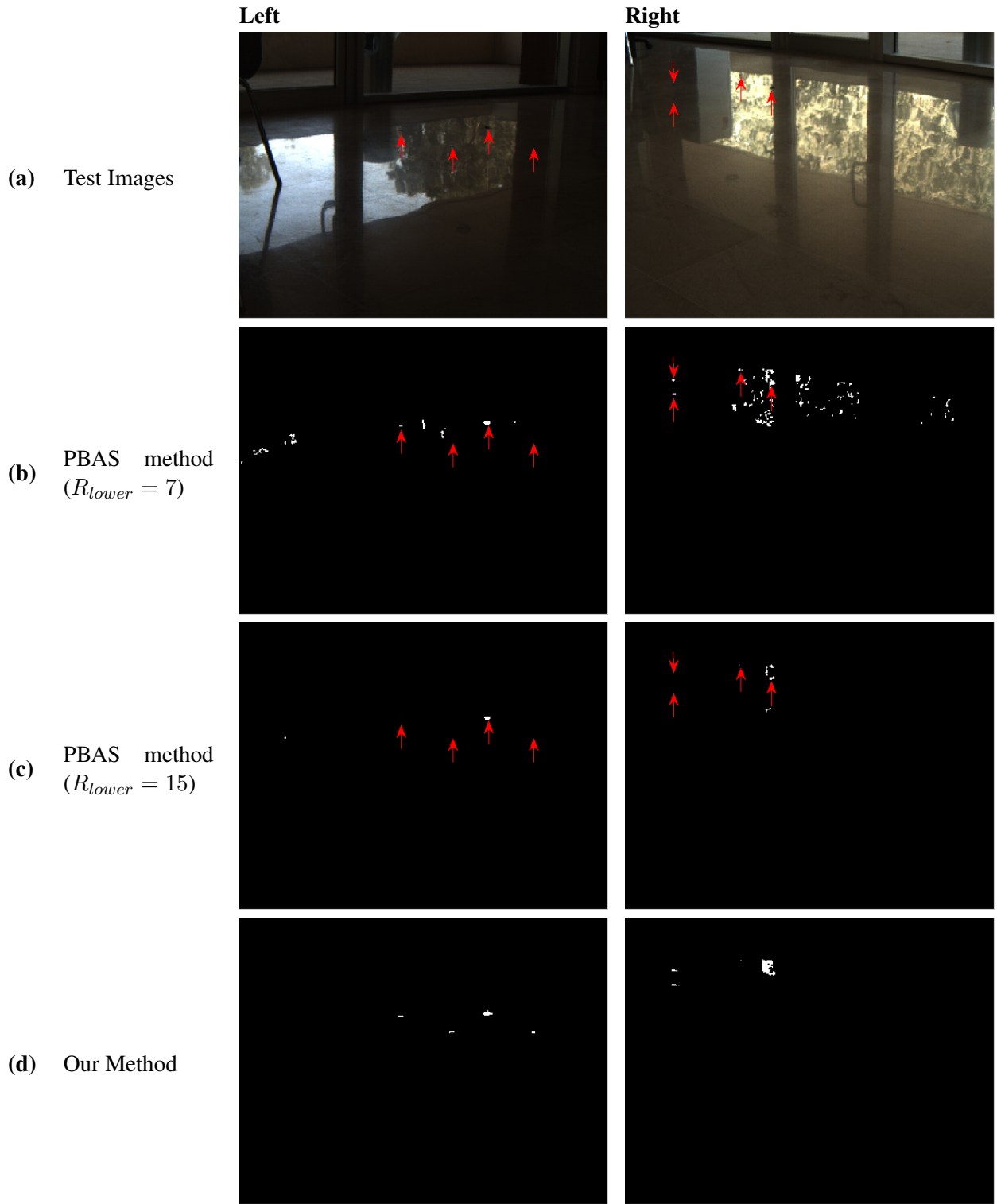
**Left**  **Right**

**(a)** Test Images

**(b)** PBAS method $(R_{lower} = 7)$

**(c)** PBAS method $(R_{lower} = 15)$

**(d)** Our Method

Figure 6.15: Results from experiment 6, Test 1. **(a)** Test images from two cameras. **(b)** Results obtained after running the PBAS method using parameter $R_{lower} = 7$. Many false positives can be seen in both images, but not all foreground objects were detected. **(c)** Results obtained after running the PBAS method using parameter $R_{lower} = 15$. There are fewer false positives, but also fewer foreground detections. **(d)** Results of our method. No false positives while all four foreground objects were detected.
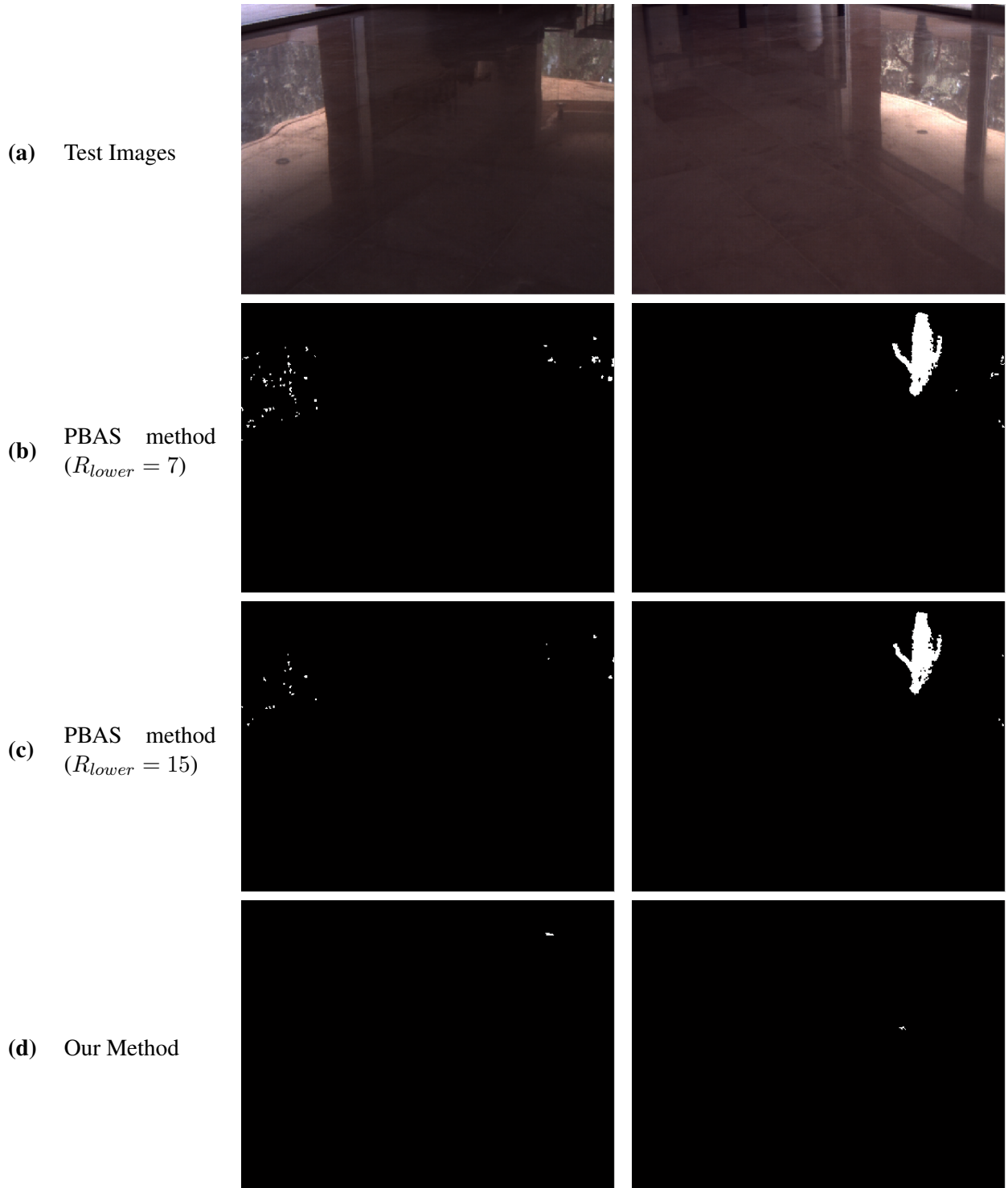
**(a)** Test Images

**(b)** PBAS method $(R_{lower} = 7)$

**(c)** PBAS method $(R_{lower} = 15)$

**(d)** Our Method

Figure 6.16: Results from experiment 6, Test 2. **(a)** Test images from two cameras. In the right image, a person is reflected on the floor. **(b)** Results obtained after running the PBAS method using parameter $R_{lower} = 7$. The person's reflection is detected in addition to many other false positives dues to swaying trees being reflected on the floor. **(c)** Results obtained after running the PBAS method using parameter $R_{lower} = 15$. Although there are fewer false positives produced by the swaying trees, the person's reflection is still detected. **(d)** Results of our method. The person's reflection is correctly discarded, and only one small false positive is detected.
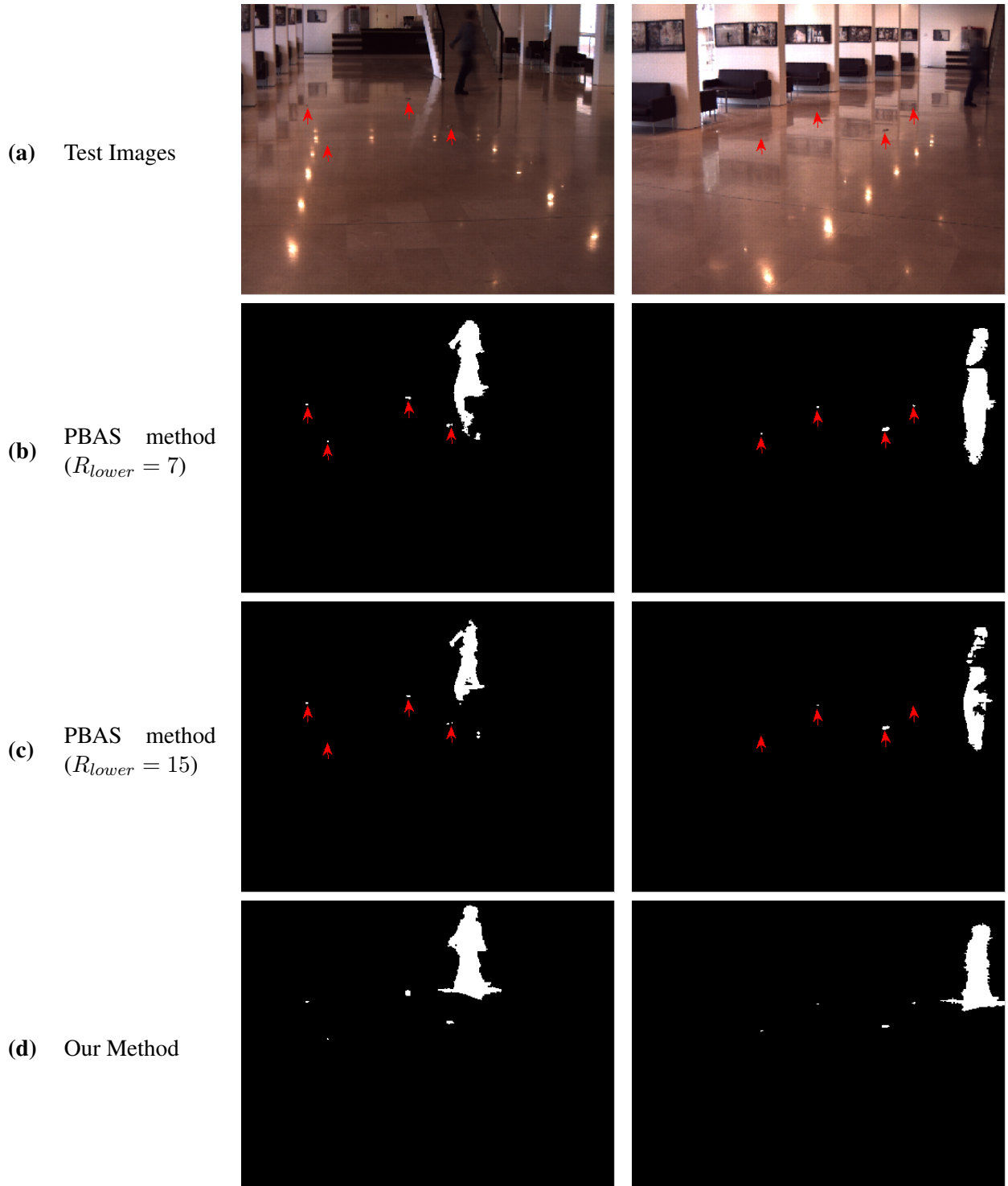
Figure 6.17: Results from experiment 6, Test 3. **(a)** Test images from two cameras. Four small objects and a walking person are located on the floor. **(b)** Results obtained after running the PBAS method using parameter $R_{lower} = 7$. All four objects are detected. The person's blob is not complete and his reflection on the floor is detected too.**(c)** Results obtained after running the PBAS method using parameter $R_{lower} = 15$. Some of the small objects are not detected and the person's blob has many gaps. In the right image, the person's reflection is still detected. **(d)** Results of our method. All four objects are detected, with no false positives. The person's blob is fully detected while his reflection is correctly discarded.
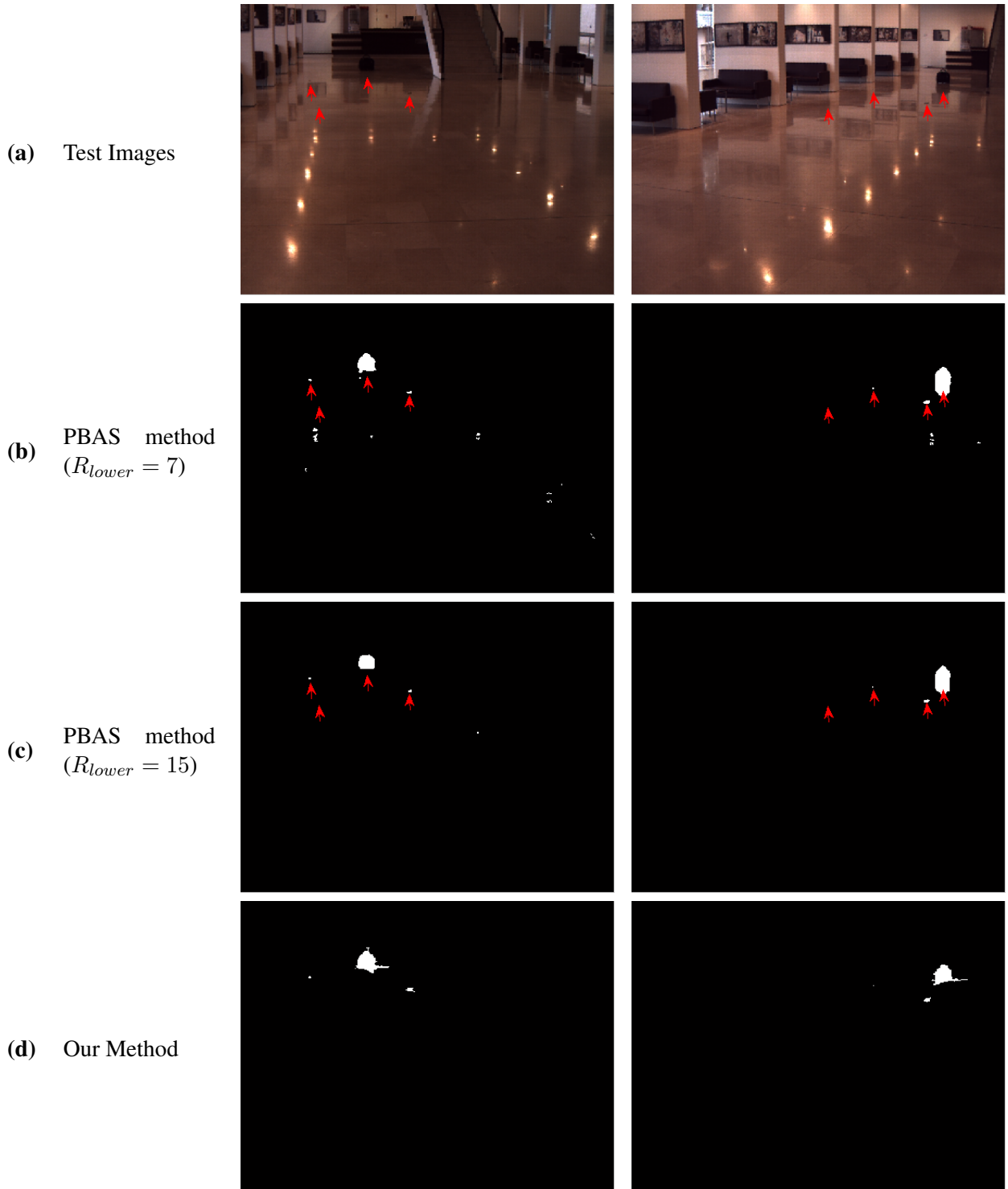
**(a)** Test Images

**(b)** PBAS method $(R_{lower} = 7)$

**(c)** PBAS method $(R_{lower} = 15)$

**(d)** Our Method

Figure 6.18: Results from experiment 6, Test 4. **(a)** Test images from two cameras. Three small objects and a large handbag are placed on the floor. **(b)** Results obtained after running the PBAS method using parameter $R_{lower} = 7$. Two small objects only, the handbag, and many false positives are detected. **(c)** Results obtained after running the PBAS method using parameter $R_{lower} = 15$. Two small objects only, the handbag, and 1 small false positive were detected. **(d)** Results of our method. No false positives, two small objects, and the handbag were detected.
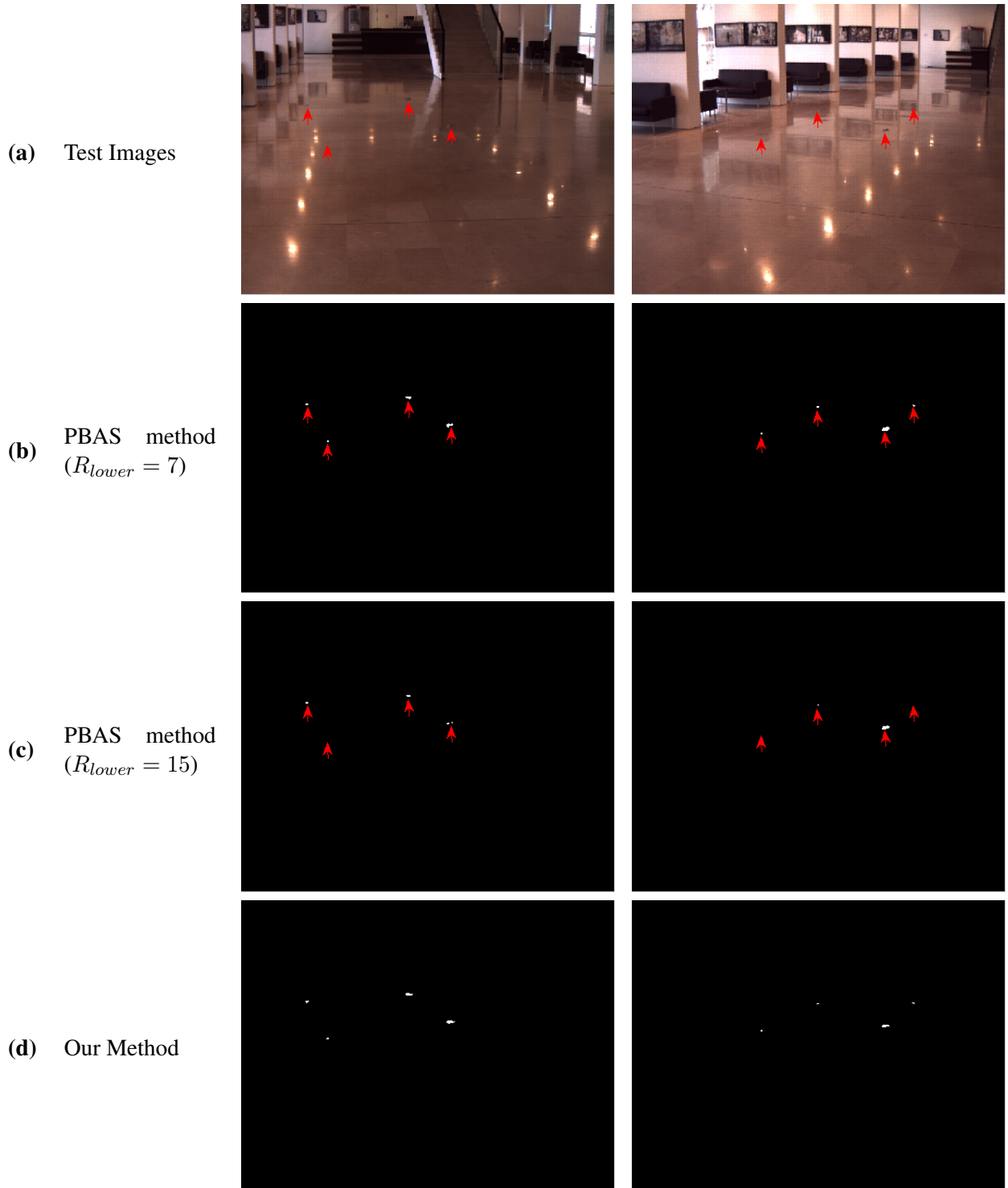
**(a)** Test Images

**(b)** PBAS method $(R_{lower} = 7)$

**(c)** PBAS method $(R_{lower} = 15)$

**(d)** Our Method

Figure 6.19: Results from experiment 6, Test 5. **(a)** Test images from two cameras. Four small objects are placed on the floor. **(b)** Results obtained after running the PBAS method using parameter $R_{lower} = 7$. All four objects were correctly detected with no false positives. **(c)** Results obtained after running the PBAS method using parameter $R_{lower} = 15$. No false positives but some of the foreground objects were missed. **(d)** Results of our method. All four objects were correctly detected with no false positives.

# Chapter 7

# Summary and Conclusions

In this thesis we deal with change detection on reflective surfaces such as shiny floors, glasses, and mirrors in an indoor environment and wet surfaces outdoors. Although such scenes are very common, they were hardly ever explicitly considered in previous studies. This is probably because most change detection algorithms are based on information from a single camera, which makes the problem hard to solve. In particular we demonstrated that it is impossible to achieve acceptable performance without using additional prior knowledge about the object to be detected. This was our main motivation do develop a multi-view method.

In order to deal with reflections, we first developed a model of a reflective background using a mixture of a Gaussian and uniform distributions. We then showed how such a model would perform poorly using a single camera. Next, we developed a background model using two cameras in a wide baseline configuration, for which we assume a very low probability that transient objects will be reflected on the ground plane at the same place and at the same time in both images. We also assume a high probability that true foreground objects will appear in both images. Our method uses these assumptions to improve the foreground/background segmentation performance.

Another motivation for our work was to detect objects which are very small or have low contrast to the background. This is also considered to be a difficult problem when a single view is used. The tradeoff that exists between the detection of such objects and the considerable amount of false positives was shown both theoretically (Chapter 3) and empirically (Chapter 6). A set of experiments demon-

strated the limitations of our single camera change detection method and that of the state-of-the-art single camera change detection method ([Hofmann et al., 2012]). When using a pair of cameras, our method's detection performance drastically improved while producing almost no false detections. It is important to note that a naive use of a pair of cameras for computing 3D structure of the scene is not adequate for the considered scene, since objects seen from different views have different photometric properties. Our method also uses the fact that the background noise seen by each camera is independent, to improve the probability of detecting small and low contrast objects.

When developing our reflective background model, we chose to develop a very simple parametric model based on a single Gaussian. We chose this simple model, rather than more expressive state-of-the-art methods, for two main reasons. First, the single Gaussian model was simpler to analyze mathematically and it was simple to generate the decision region based on the likelihood ratio function. Second, this model made it simpler to demonstrate the performance gain achieved by adding an additional camera to the change detection process. Using a more expressive single camera model might have masked this gain. We concluded that the performance gain using two images for change detection is so drastic as to outperform even a state-of-the-art single camera method despite the very simple and very fast algorithm that we used.

When analyzing our algorithm we came across a number of limitations and improvements which we address here:

- Our method does not perform well when the probability for transient background is high. This caused, for example, the false detection in experiment 5. Such false detections could be eliminated in several ways. We could adapt our method to use more than 2 cameras. Using additional views would decrease the chance of false detections due to multiple transient objects, since the chances of transient objects corresponding in all views is low. In cases where the transient background is due to moving items such as swaying trees or bushes, a very simple improvement would be to replace the single image change method that we use as a first step in our algorithm with a more advanced state-of-the-art method, which uses a more expressive background model. Since such methods would produce fewer false detections, and since the algorithm would be used as the first step of our method, the overall number of false detections in our method would decrease.

A further improvement might be to use a more complicated background model in our image pair algorithm. Instead of using a single Gaussian we could model our reflective background as a mixture of Gaussians and adapt our likelihood ratio based decision region accordingly. This should lower the chance of foreground detections in our method's second step.

- Our method assumes that the ground surface is planar. This enables us to use a homography transformation that maps corresponding pixel in each image. When the ground surface is not planar then there is no projective transformation that maps corresponding pixels. If dense correspondence is available using stereo methods, we can use it instead of the homography based correspondence. No changes would be required to the remainder of the algorithm. This process needs to be done only once offline. Thus, CPU intensive stereo matching algorithms can be used with no need for specialized hardware assistance. This relatively easy and simple extension of our method could only be used if the dense stereo based correspondence is available.

To summarize, we believe that using multiple cameras has great potential for improving the performance of change detection methods in general, especially where reflective surfaces are present.

# Bibliography

[Aach and Kaup, 1995] Aach, T. and Kaup, A. (1995). Bayesian algorithms for adaptive change detection in image sequences using markov random fields. *Signal Processing: Image Communication*, 7:147–160.

[Akman et al., 2008] Akman, O., Alatan, A., and Ciloğlu, T. (2008). Multi-camera visual surveillance for motion detection, occlusion handling, tracking and event recognition. In *Workshop on Multi-Camera and Multimodal Sensor Fusion Algorithms and Applications*.

[Bahadori et al., 2007] Bahadori, S., Iocchi, L., Leone, G., Nardi, D., and Scozzafava, L. (2007). Real-time people localization and tracking through fixed stereo vision. *Applied Intelligence*, 26:83–97.

[Brutzer et al., 2011] Brutzer, S., Hoferlin, B., and Heidemann, G. (2011). Evaluation of background subtraction techniques for video surveillance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1937–1944.

[Cristani et al., 2010] Cristani, M., Farenzena, M., Bloisi, D., and Murino, V. (2010). Background subtraction for automated multisensor surveillance: a comprehensive review. *EURASIP Journal on Advances in Signal Processing*, pages 1–24.

[Cucchiara et al., 2001] Cucchiara, R., Grana, C., Piccardi, M., Prati, A., and Sirotti, S. (2001). Improving shadow suppression in moving object detection with hsv color information. In *Proceedings of the IEEE Intelligent Transportation Systems*, pages 334–339.

[Duda et al., 2001] Duda, R., Hart, P., and Stork, D. (2001). *Pattern Classification*. John Wiley, New York.

[Faugeras, 1993] Faugeras, O. (1993). *Three-Dimensional Computer Vision: a Geometric Viewpoint*. The MIT Press.

[Goldlucke and Magnor, 2003] Goldlucke, B. and Magnor, M. (2003). Joint 3d-reconstruction and background separation in multiple views using graph cuts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 683–688.

[Gordon et al., 1999] Gordon, G., Darrell, T., Harville, M., and Woodfill, J. (1999). Background estimation and removal based on range and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 454–459.

[Goyette et al., 2012] Goyette, N., Jodoin, P.-M., Porikli, F., Konrad, J., and Ishwar, P. (2012). Changedetection.net: A new change detection benchmark dataset. In *Proceedings of the IEEE Workshop on Change Detection*, pages 1–8.

[Hartley et al., 2003] Hartley, R., Zisserman, A., and ebrary, I. (2003). *Multiple View Geometry in Computer Vision*, volume 2. Cambridge University Press.

[Harville et al., 2001] Harville, M., Gordon, G., and Woodfill, J. (2001). Foreground segmentation using adaptive mixture models in color and depth. In *Proceedings of the IEEE Workshop on Detection and Recognition of Events in Video*, pages 3–11.

[Hofmann et al., 2012] Hofmann, M., Tiefenbacher, P., and Rigoll, G. (2012). Background segmentation with feedback: The pixel-based adaptive segmenter. In *Proceedings of the IEEE Workshop on Change Detection*, pages 38–43.

[Ivanov et al., 2000] Ivanov, Y., Bobick, A., and Liu, J. (2000). Fast lighting independent background subtraction. In *Proceedings of the International Journal of Computer Vision*, volume 37, pages 199–207.

[Khan and Shah, 2006] Khan, S. and Shah, M. (2006). A multiview approach to tracking people in crowded scenes using a planar homography constraint. *European Conference on Computer Vision*, pages 133–146.

[Krumm et al., 2000] Krumm, J., Harris, S., Meyers, B., Brumitt, B., Hale, M., and Shafer, S. (2000). Multi-camera multi-person tracking for easyliving. In *Proceedings of the Third IEEE International Workshop on Visual Surveillance*, pages 3–10.

[Lanza et al., 2007] Lanza, A., Di Stefano, L., Berclaz, J., Fleuret, F., and Fua, P. (2007). Robust multi-view change detection. In *Proceedings of the British Machine Vision Conference*.

[Lim et al., 2005] Lim, S., Mittal, A., Davis, L., and Paragios, N. (2005). Fast illumination-invariant background subtraction using two views: Error analysis, sensor placement and applications. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1071–1078.

[Neyman and Pearson, 1933] Neyman, J. and Pearson, E. (1933). On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, pages 289–337.

[Piccardi, 2004] Piccardi, M. (2004). Background subtraction techniques: a review. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 3099–3104.

[Qunyu et al., 2009] Qunyu, X., Huansheng, N., and Weishi, C. (2009). Video-based foreign object debris detection. In *Proceedings of the IEEE International Workshop on Imaging Systems and Techniques*, pages 119–122.

[Radke et al., 2005] Radke, R., Andra, S., Al-Kofahi, O., and Roysam, B. (2005). Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14:294–307.

[Stauffer and Grimson, 1999] Stauffer, C. and Grimson, W. (1999). Adaptive background mixture models for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1–2.

[Wren et al., 1996] Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. (1996). Pfinder: Real-time tracking of the human body. In *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, pages 51–56.

[Yoon and Kweon, 2004] Yoon, K. and Kweon, I. (2004). Voting-based separation of diffuse and specular pixels. *Electronics Letters*, 40:1260–1261.

[Zhao et al., 2005] Zhao, T., Aggarwal, M., Kumar, R., and Sawhney, H. (2005). Real-time wide area multi-camera stereo tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 976 – 983.

# תקציר

רצפות מבריקות, זכוכיות ומראות בתוך מבנים, ומשטחים רטובים בחוץ, משקפים אובייקטים שאינם בהכרח נמצאים באזור העניין  של הסצנה. למרות שמשטחים רפלקטיביים כאלה נמצאים בכל מקום, הם כמעט אף פעם לא טופלו באופן מפורש במחקרים קודמים בהקשר של גילוי שינויים. בתזה זו אנו מטפלים בבעיה של גילוי שינויים במשטחים רפלקטיביים, כאשר האובייקטים לגילוי הינם קטנים יחסית בזירה וצבעם אינו מובטח להיות שונה באופן משמעותי מהרקע.

לאובייקטים מסוג זה נאמר שיש להם יחס אות לרעש (SNR) נמוך. המטרה שלנו הינה לזהות את השינויים האמיתיים אשר נובעים מאובייקטים הנמצאים על המשטח ולהתעלם מאלה המשתקפים ממנו. במסגרת העבודה אנחנו מבצעים שתי תרומות לקראת השגת מטרה זו. התרומה הראשונה הינה ניתוח תיאורטי של גילוי שינויים ברמת הפיקסל בנוכחות משטחים משתקפים. הניתוח מבוצע עבור מצלמה אחת וזוג מצלמות בהקשר של יחס אות לרעש נמוך. התרומה השנייה הינה אלגוריתם לביטול השתקפות עצמית של אובייקטים באמצעות זוג מצלמות. על ידי שילוב גילוי שינויים רגיש ברמת פיקסל עם אלגוריתם שתי המצלמות, אנו מסוגלים לזהות אובייקטים בעלי יחס אות לרעש נמוך תוך כדי התעלמות מהשתקפויות לא רלוונטיות. אנחנו מציגים תוצאות מוצלחות של האלגוריתם בסצנות מאתגרות ביותר.

# שימוש בשתי מצלמות עבור גילוי שינויים ברגישות גבוהה על משטחים רפלקטיביים

סיכום עבודת גמר המוגשת כמילוי חלק מהדרישות לקראת קבלת
תואר מוסמך במסלול מחקרי במדעי המחשב

על-ידי עודד הנסון

העבודה בוצעה בהנחיית ד"ר יעל מוזס

אוקטובר,2012