COPPE
UFRJ
Instituto Alberto Luiz Coimbra de
Pós-Graduação e Pesquisa de Engenharia

ANOMALY DETECTION WITH A MOVING CAMERA USING SPATIO-TEMPORAL CODEBOOKS

Mateus Teruyuki Nakahata

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientadores: Eduardo Antônio Barros da Silva
Sergio Lima Netto

Rio de Janeiro
Setembro de 2016

ANOMALY DETECTION WITH A MOVING CAMERA USING
SPATIO-TEMPORAL CODEBOOKS

Mateus Teruyuki Nakahata

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Eduardo Antônio Barros da Silva, Ph.D.

Prof. Sergio Lima Netto, Ph.D.

Prof. José Gabriel Rodriguez Carneiro Gomes, Ph.D.

Prof. Siome Klein Goldenstein, Ph.D.

Prof. Carla Liberal Pagliari, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
SETEMBRO DE 2016

Nakahata, Mateus Teruyuki

Anomaly Detection with a Moving Camera Using Spatio-Temporal Codebooks/Mateus Teruyuki Nakahata. – Rio de Janeiro: UFRJ/COPPE, 2016.

XVIII, 101 p.: il.; $29,7$cm.

Orientadores: Eduardo Antônio Barros da Silva
Sergio Lima Netto

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2016.

Bibliography: p. 96 – 101.

1. video surveillance. 2. moving camera. 3. spatio-temporal codebook. 4. cluttered environment. 5. abandoned object detection. I. Silva, Eduardo Antônio Barros da *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

para minha esposa Liliane e meu filho Bruno.

Agradecimentos

Agradeço aos meus pais, Hidekazu (*in memoriam*) e Takako Nakahata, que sempre me apoiaram e foram um exemplo de perseverança e honestidade.

Obrigado aos meus orientadores, Eduardo Antônio Barros da Silva and Sergio Lima Netto, pela sua paciência, orientação, conhecimento e constante incentivo.

Também gostaria de agradecer aos colegas do SMT que me ajudaram sempre que possível.

Aos meus colegas da PETROBRAS, pelo apoio e incentivo.

Por fim, agradeço a PETROBRAS, empresa na qual me orgulho em trabalhar, e que me apoiou nesta jornada.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

DETECÇÃO DE ANOMALIA COM UMA CÂMERA MÓVEL UTILIZANDO DICIONÁRIOS ESPAÇO-TEMPORAIS

Mateus Teruyuki Nakahata

Setembro/2016

Orientadores: Eduardo Antônio Barros da Silva
Sergio Lima Netto

Programa: Engenharia Elétrica

Esta tese propõe um novo método para detectar anomalias em vídeos utilizando uma câmera montada em um robô de inspeção. O método utilizado tem como base o *spatio-temporal composition* (STC), onde uma amostragem densa é utilizada para decompor o vídeo em pequenos volumes 3D e então é calculada a probabilidade dos arranjos espaço-temporais formados por estes volumes. Esta classe de métodos tem sido utilizada com sucesso em vídeos de vigilância obtidos de câmeras estáticas. No entanto, quando aplicado em vídeos gravados de uma plataforma móvel, o STC fornece um grande número de falsas detecções. Com o intuito de resolver este problema, são propostas melhorias no método STC baseadas em duas frentes. Primeiro, é realizado um treinamento de um dicionário em dois estágios para possibilitar uma detecção de anomalias mais confiável. Segundo, são empregadas características espaço temporais melhoradas, que são extraídas após uma filtragem espaço-temporal otimizada que realiza uma regulagem temporal da sequência. A abordagem proposta obtém bons resultados na identificação de anomalias, sem a necessidade da subtração do plano de fundo, estimativa de movimento ou rastreamento. O sistema foi preciso mesmo sem um conhecimento prévio do tipo de evento a ser observado, sendo robusto a ambientes tumultuados, como ilustrado por vários exemplos. Estes resultados são obtidos sem comprometer a performance da detecção de anomalias no caso de câmeras estáticas.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

ANOMALY DETECTION WITH A MOVING CAMERA USING SPATIO-TEMPORAL CODEBOOKS

Mateus Teruyuki Nakahata

September/2016

Advisors: Eduardo Antônio Barros da Silva
Sergio Lima Netto

Department: Electrical Engineering

This thesis proposes a method to detect anomalies in a video using a camera mounted on an inspection robot. The developed method is based on spatio-temporal composition (STC) method, where a dense sampling is used to break the video into small 3D volumes and it is calculated the probability of the spatio-temporal arrangements of these volumes. This class of methods has been successfully used for surveillance videos obtained by a static camera. However, when applied to videos recorded from a moving platform, STC gives a large number of false detections. In order to solve this problem, we propose improvements to the STC method in two fronts. First, a two stage dictionary learning process is performed in order to allow a more reliable anomaly detection. Second, improved spatio-temporal features are employed, that are extracted after an enhanced temporal filtering that performs a temporal regularization of the video sequence. The proposed approach gives very good results in the identification of anomalies, without the need of background subtraction, motion estimation or tracking. The system was accurate even with no prior knowledge of the type of event to be observed, being robust even in cluttered environments, as illustrated by several practical examples. These results are obtained without compromising the performance for anomaly detection in the static camera case.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nowadays, security is a constant concern, and more and more companies, the government and even ordinary citizens invest financial resources in electronic security systems. Among the electronic devices the most used and popular are the surveillance cameras. There are several factors that helped to popularize the CCTV (Closed Circuit TV) systems, as the large coverage area of a camera, the reduction in equipment costs in the last years and the advent of IP surveillance systems, which have an easy configuration and expansion. IP CCTV systems also allow the remote visualization of images from anywhere in the world, when connected to the Internet. Also in industry, increasingly processes are monitored through cameras. In environments such as refineries and platforms, for example, the working conditions are unhealthy and subject to risks, with the presence of dangerous gases, loud noise, heat, steam and various instruments that can cause injury in case of falls or even when touched. So, in this hazardous environment the use of cameras enables operators to monitor the industrial plant in a safe and comfortable environment. However, in this scenario, the supervisors and operators are exposed to several images, and despite the amount of data being large, the information obtained is little. In the surveillance rooms, these operators work in shifts on 24×7 basis. In addition to being a stressful environment, studies show that after about twenty minutes the operator loses his ability to pay attention to the images of the cameras [1]. The greater the number of cameras, the faster this loss, and it is not uncommon for the operator to monitor hundreds of images at the same time. It is humanly impossible to pay attention to so many images at once. Experience shows that the practical limit for an operator is a maximum of 16 images at the same time, for a period not exceeding one hour [2].

In the next section, we present an overview of the techniques used to assist security system operators in the anomaly identification in a video. In the last section, a summary of the work proposed in this thesis is presented.

1.1 Video Analysis Systems

In order to assist security system operators, automated systems for video analysis have been widely used. However, these systems need to be trained and configured to work correctly. This requires prior knowledge of the events of interest, which is not always possible. That is, in traditional systems it is necessary a prior knowledge of the characteristics of what is considered normal and then one must configure the system to identify when these characteristics are not satisfied. Furthermore, often the monitored environments are cluttered or they change over time, and the video analysis systems need to be constantly reconfigured. This requires a professional with a high degree of technical expertise and knowledge of environmental characteristics, which has a high cost and is difficult to find because this is not the normal profile of an operator or vigilant found in the labor market.

Therefore, it was considered that the automatic identification of anomalies is a promising theme but still has a lot to be developed, because the systems that currently perform this kind of task work well for very specific situations, but are not generic enough. It was also found that most of the methods used for this purpose are based on the tracking of objects, varying the type of descriptor used to characterize the objects of interest and the method used to determine the relationships between them. But most of the methods studied that use tracking as basic function do not have a good performance in situations with great variation in the type of anomaly and in cluttered environment.

On the other hand, new anomaly detection methods which do not use tracking have been developed, such as those using the analysis of small spatio-temporal volumes obtained by dense sampling of the video of interest. These methods perform better in cluttered environments than methods that use tracking [3, 4]. One approach that has been used in the video analysis is the bag of video words method (BOV), where analysis is performed through small spatio-temporal volumes, and the redundancy between them is minimized through the use of a codebook [5]. Such methods tend to perform well in cluttered environments. However, image interpretation by the human being is influenced by spatio-temporal composition between the objects that make up this image [6], and such composition is not considered in the BOV. The spatio-temporal composition method (STC) [7] takes into account a spatio-temporal array of small volumes of videos and performs a modeling using a probabilistic approach. In it, anomalous events are those with a low probability of occurrence. A further characteristic of STC is that it can be trained on-line, being able to adapt as environmental conditions change and requiring little or even no pre-settings for the detection of anomalies. In addition, the STC method is fast enough to be used in real-time. Although it performs very well for anomaly detection in videos taken with static cameras, the STC usually fails when the camera is on a moving platform, in which case the background often changes continuously and at a fast rate. The case of cameras in moving platforms is a hard one. There are not many works in the literature that deal with it. But in surveillance applications, moving cameras reduce costs because a single camera can cover an area where several fixed cameras would be needed to get the same coverage. The downside is that these cameras can show only a small portion of the area of interest at a time. Cameras may also be mounted on inspection robots, which makes necessary the development of anomaly detection algorithms that operate on these cases.

An example of anomaly detection using a moving camera is the one described [8]. There, a camera is mounted on a robot that performs a translational movement. First, one records a reference video of what is considered a normal situation along the path of the camera. Every new video along the path is compared with this reference to detect abandoned objects. Salient points of reference and observation images are obtained through the speeded up robust features (SURF) algorithm [9]. The videos are registered by means of an homography computed among their frames using the detected salient points and the random sample consensus (RANSAC) algorithm [10]. The normalized cross correlation (NCC) [11] is applied between two images (the reference video and observation). After a post-processing operation, a threshold determines the areas in the frame that are considered to be abandoned objects.

1.2 Proposed Work

Given the nice properties of the STC method, a natural development would be to apply it to the moving-camera case. Unfortunately, as will be described later in this thesis, the STC fails when applied to surveillance videos obtained from a camera mounted on a moving robot, such as the ones used in [8]. Several tests were performed with the STC method when the presence of an abandoned object was the event of interest and that method was not able to distinguish the abandoned object from the background.

The first and more important contribution of this work is the introduction of a second codebook, to minimize the number of false identification of abandoned objects. The training video is broken down in small volumes and similar volumes are grouped to create codewords in the first codebook. This codebook is then used to compute the occurrence probability of each volume, and volumes with a probability below a given threshold are considered anomalies. For an abandoned object to be detected, the threshold must be set to a value greater than the probability of the volumes that form this object. But sometimes the abandoned object is similar to some parts of the background with a probability in an average level, and if the threshold is set to this value, parts of the background with probability below it are detected too. To avoid this issue, a second training stage with the same training video is performed, where only the volumes with a probability below the threshold are used to create a second codebook. During the analysis of the target video, a volume is considered anomalous only if it has a probability below the threshold and it is not similar to a codeword present in the second codebook. The second codebook has considerably improved the performance of the method.

The second contribution is the use of a new spatio-temporal descriptor. The STC method uses only temporal gradients, but a better result is obtained when both spatio and temporal gradients are used, especially in the case when the anomaly to be detected is an abandoned object. One problem with the temporal descriptor is that it is very sensitive to variations in the movement of the camera, like shaking or an inconstant moving speed. The spatial gradient is not affected by the movement of the camera, because it is calculated using only the pixels of the same frame. So, even if the camera stops moving, the space derivative is able to detect differences in the image.

The third contribution is the introduction of a Gaussian filtering to deal with misalignments caused by camera shaking. Only the introduction of the spatial derivative in the descriptor is not able to eliminate the effect of the shaking or inconstant speed of the moving camera and a more specific solution had to be developed. But this Gaussian filter must be used with care, because if the temporal derivative is too smoothed, some variations in the scene may be lost, and the identification of anomalies could be compromised. So, a very important activity was the tuning of the parameters of the Gaussian filter and the descriptor. As the parameters are correlated, it was necessary to simulate several times, changing all the parameters, to determine the best configuration set.

The assessment of the proposed method is carried out using seven videos from the publicly available VDAO database [12]. This database contains reference videos without abandoned objects as well as several versions of the same video with different abandoned objects. The spatio-temporal codebooks are computed using the reference video, and their performance is assessed using the target videos containing abandoned objects. During the creation of this database, a software was developed to help mark by hand the position of the objects and to generate a file with the coordinates of the objects in each frame of the video.

To describe the proposed methodology, this work is organized as follows: Chapter 2 presents the main techniques currently used to detect anomalies, problems encountered in detecting anomalies and the works that guided the choice of the theme of this thesis. Chapter 3 presents a short description of the main steps of the

STC method and evaluates its performance in both static and moving camera cases. Chapter 4 presents a description of the proposed methodology, detailing its main steps and important parameters. Chapter 5 describes the VDAO database, the type of environment which this database intends to simulate, the characteristics of the robot used to record the video and it shows some examples of the abandoned objects found in each video. It also describes the software developed to mark the abandoned objects in the video and to create a reference video to assess the performance of the method. Chapter 6 details the methodology used to tune the parameters of the proposed algorithm and presents the results obtained, numerically and graphically. In Chapter 7 the performance of the proposed algorithm is assessed through simulations using the VDAO database, and a comparison with the STC method and the BOV method is performed. Finally, the Chapter 8 concludes the thesis, analyzing the main contributions of the proposed methodology.

Chapter 2

Background

In this chapter we will present the main techniques currently used to detect anomalies, the problems encountered when using these techniques and the works that guided the choice of the theme of this thesis.

2.1 Literature Review

With the increasing number of CCTV systems, it is growing the interest in electronic surveillance systems in which computer programs are able to replace the operator in order to obtain a more reliable and robust system [2]. In this context, robustness is defined as the degree of correction of the outputs of a system in the presence of incorrect entries or a hostile environment [13]. Systems that detect anomalies seek to determine what is abnormal in the scene, which usually is related with a threat, and signal when this happens. The vast majority of techniques aim to create a model of what is considered normal, and an event is considered abnormal according to the degree to which it differs from this model.

Surveillance systems that use cameras as a source of information perform an automatic analysis of images using computer vision programs. The algorithms used can be developed for a single specific activity, such as detecting an abandoned object, the removal of an object from a scene, and detecting a specific object, such as a regular vehicle in an exclusive track for buses, pedestrians walking on prohibited place, etc. Such algorithms can also be developed to detect behavioral aspects such as agglomerations, direction of flow of people, determined trajectory, etc. These algorithms have a limited application to situations where they know the monitored environment and the events to be detected.

A typical monitoring system can be divided into two levels of processing: low level and high level [14]. High-level processes use the low-level information to determine the type or nature of the activity performed by the moving object, such as running, walking, dancing, etc. The lower level consists of motion detection techniques, object classification and tracking. Motion detection generally uses background subtraction algorithms [15, 16], temporal differentiation [17] or optical flow [18] to separate the pixels that participate in a given movement from the pixels of the background. Background subtraction analysis is performed using a comparison between every pixel in the target image and the reference image. This kind of method is usually very sensitive to changes in the illumination and position of the image. In the temporal differentiation, it is calculated the pixel difference between two or more consecutive images. Optical flow is used to compute the apparent motion of the objects in an image. It can be calculated using differential methods, gradients or phase correlation. This method is usually slow and sensitive to noise.

The object classification is the process of classifying objects detected in preset classes [19]. In the tracking [20], the objects detected and classified have their position and trajectory mapped in time and space.

Most articles related to anomaly detection are based on analysis of the trajectory of objects, which demand for tracking methods [21]. In tracking, techniques as Kalman filter [22] or particle filter (*condensation*) [23] are used to identify the foreground pixels over time, and classify them as belonging to a moving object or to a static object. As most real tracking problems are not linear and non-Gaussian, better results are obtained when using methods such as particulate filter with Monte Carlo simulations, and its many variants [24]. Tracking based on contour and patterns are also used [25], as well as those using Bayesian methods [26]. In [14], new events are classified as normal or abnormal through a training of what is considered a normal events. First is made a background subtraction, using Gaussian mixture model (GMM) [27] and then a detection of the objects is performed. To determine the position of the anomaly, the homography of multiple cameras is used. To determine if the event type is normal or anomalous, two analyses are performed: one using support vector machine (SVM) [28, 29] to determine whether the movement is normal, and other using hidden Markov model (HMM) [29] to determine whether the trajectory is normal. The composition of the two analyses is used for the final classification.

To improve the image processing, behavioral analysis is used to identify the agents of the actions (people, animals, machines, etc.) that can be modeled, tracked and transformed into a numerical representation. Among the most popular methods are the HMM and Bayesian networks (BN) [2, 30]. As the behavior of objects depends on the context of the scene, its analysis can improve the performance of tracking and identification algorithms. The contextual information can be obtained as an external input, by operator entry, or through continuous training based on the observation of past samples. With data such as position of entrance and exit, obstacles and dimensions, it is possible to determine or restrict the position and the trajectory of the target objects. One of the most used techniques to model inputs and outputs on the scene is the GMM with expectation maximization (EM) [31, 32].

However, these sophisticated methods that use the relationship between a real object and its representation in the picture by pixels are usually very complex, depending on many factors that affect the accuracy of the tracking. The main tracking challenges [32] are listed in Fig. 2.1. Correct identification difficulties arise due to the similarity of the target object with other objects in the scene, as well as changing the representation of the object due to factors such as changes in position, lighting or occlusion. Tracking techniques have also not presented satisfactory results in cluttered environments with many people or objects [33, 34]. In industrial plants, where it is difficult to differentiate the background, tracking techniques do not get good results in the identification of anomalous events. Fig. 2.2 shows an example of this kind of environment. In these challenging scenarios, analysis performed mainly with low level features has better performance than the ones that use high level features because they can be extracted more reliably over time [3]. Low level features have an interesting application in the video surveillance when applied to automatic identification of abandoned objects in cluttered environments. This kind of application is usually performed by comparing a target video to a reference video containing what is considered a normal situation. Any new object present in the target video and not present in the reference video is considered as an abandoned object.

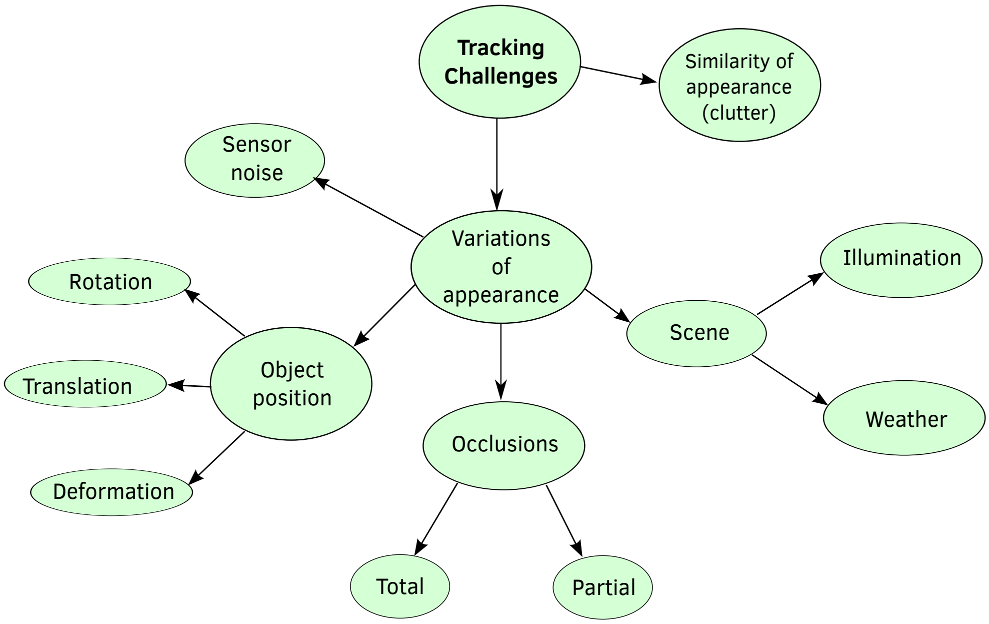


Figure 2.1: The main tracking challenges. Many elements can affect the accuracy of the tracking.

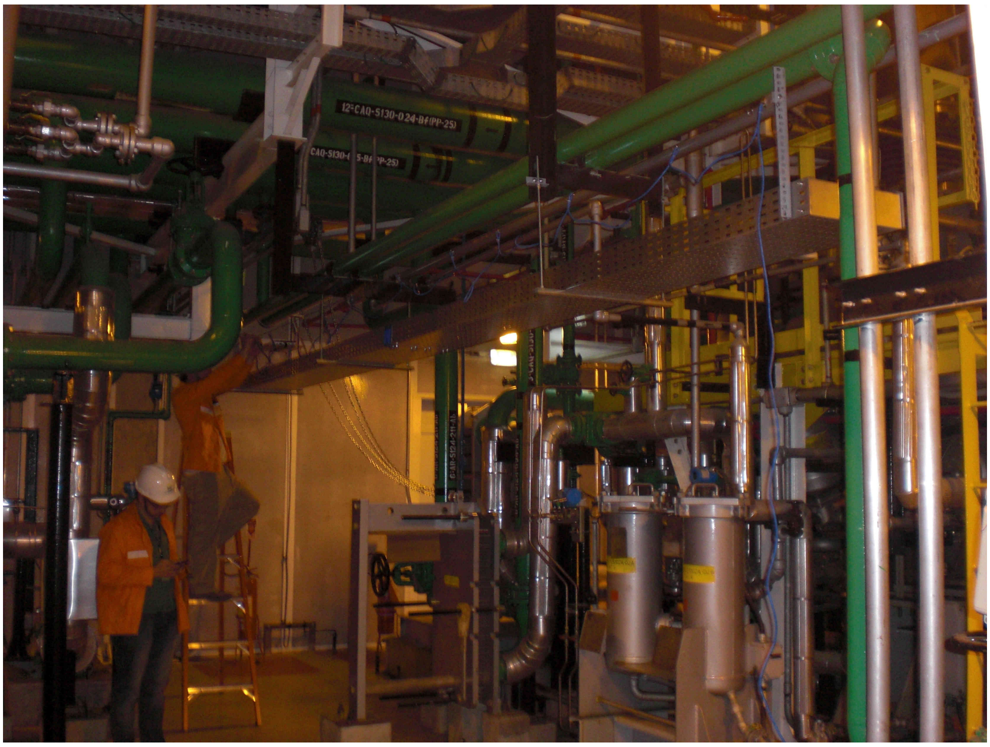


Figure 2.2: An industrial plant, with a very cluttered environment.

2.2 Anomalies Detection Through Spatio-Temporal Arrangements

Alternatively, there are methods that do not use tracking or background subtraction to detect anomalies, but the analysis of local regions of the video. These regions are small spatio-temporal volumes, using the concept of bag of video words (BOV). As descriptors of these volumes one may use features like optical flow, temporal filtering, spatio-temporal filtering, histogram of oriented gradients (HOG) [35, 36] and oriented flow histograms [37]. The use of these low-level features has the advantage of ruggedness for a long period in a large variety of scenes including cluttered environments.

Cluttered environments with a large number of people or objects moving represent a major challenge in finding anomalies. In [37] an anomaly detection algorithm that extracts low-level measures as optical flow is presented, at fixed positions of the image, instead of performing measurements based on objects, since it is not used an object tracking. After a minimum period of sample acquisition, for each point a histogram of samples is created, and if the probability is below a set value, the point is replaced by an alarmed state. If the number of alarmed points is greater than a given Z number, for a number of frames of at least Y, an alarm is generated, and the points of interest are marked in the image. This method has the advantages of the use in real time, the need for few samples to initiate detection, good performance in cluttered environments and the ability to adapt to changes in the environment. However, the detection is limited to one local detection, and it is not able to detect anomalous events formed by several common and simple events.

In [33], an algorithm for detecting anomalies in cluttered environments is presented. The video is divided into spatio-temporal blocks or cuboids, and then the gradient of each spatio-temporal pixel is calculated. As each pixel within the cuboid is a sample, this is represented by a 3D Gaussian modeling with mean μ and variance σ. For each cuboid, at the same position, probable state variations are calculated with respect to time by an HMM model. The spatial relationships between various cuboids are calculated using coupled HMM models. The model presents satisfactory results, but it is complex, resulting in the difficulty of application in real time.

The selection of the spatio-temporal volume can be accomplished in several ways, such as the choice of salient points, or randomly through some dense sampling. According to [38], the best results are obtained by dense sampling. In this method, samples of small video volumes are taken at regular positions and scales in space and time. However, usually the spatio-temporal volumes arrangement is not taken into consideration, which limits the precision of the method. Studies suggest that the analysis of the spatial and temporal relationship of objects in a scene is an important part of understanding of its content by humans [5]. In [4] the reconstruction of video from previous samples and the calculation of the probability of its parts are performed. Each video is decomposed into multiple spatio-temporal volumes. This video is then recomposed using only the volumes previously observed using dense sampling. The arrangement between the neighboring volumes is also taken into consideration. However, this method cannot be used in real time due to the large computational complexity associated with dense sampling and the large number of reference samples.

Roshtkhari and Levine [7] use a similar concept for detecting anomalies, but using a codebook to reduce the number of different reference samples. The video is divided in small volumes and similar volumes are represented by the same codeword of a codebook. They also estimate the probability density function (pdf) of the arrangement of videos volumes. In the analysis phase, a representation of the video is constructed using the codebook and the probability of the arrangements is calculated using the pdf. Then, spatio-temporal arrangements with probability below a given threshold are considered anomalies. Furthermore, spatio-temporal composition (STC) can be trained on-line, being able to adapt as environmental conditions change and requires little or even no pre-settings for the detection of anomalies. In addition, this method is fast enough to be used in real time. The main steps of this method are detailed in the next chapter.

2.3 Anomalies Detection Using Moving Cameras

Most of studied methods in video anomaly detection apply to the case of fixed cameras, but a surveillance system that uses only static cameras to detect anomalies has a limited efficiency where the area to be monitored is wide or the camera is expensive or specialized such as thermal or gas-sensing cameras [39].

The use of pan-tilt-zoom (PTZ) cameras can bring some flexibility to the system. In [40], an example of the use of a moving camera is presented, where background subtraction is used in a PTZ camera, through a similarity matrix. The outliers are removed by RANSAC method. The relationship between consecutive frames is approximated by a 3-parameter similarity transformation, which is separable in the vertical and horizontal axes. Reference [41] uses a commercial PTZ camera to monitor human activity. A motion history images (MHIs) [42] is computed and events whose motion differs from that estimated by the camera are detected as anomalous.

A more flexible system is obtained with the use of a camera mounted in a moving platform, especially when the camera is specialized. In [43], a combination of microphones and cameras is used in a surveillance robot utilizing video and audio information. Microphones determine the direction of the event, and a camera is used to detect people through the use of the particulate filter. The target is a person's profile, and the movement is compared with pre-registered events.

In [44], detection of moving objects from a moving camera is performed based on the scale invariant feature transform (SIFT) [45] to extract the salient points and the RANSAC to remove the outliers. Background subtraction approach is used to perform the object detection and a dynamic background modeling is used to improve this detection.

In [11], a camera mounted on a car is used to detect abandoned objects along a path. First, a video of a path is recorded, without abandoned objects on it. Then, several tests are performed with other videos of the same path, but now with an abandoned object in these videos. The coordinates of a GPS are used to align the reference video and the target video. The salient points are obtained using SIFT, and a homography is calculated using the RANSAC to estimate affine transformation between the two video sequences. Then, the frames are compared using normalized cross correlation (NCC) and a threshold obtained experimentally is used to detect the video anomaly (abandoned object).

Another example of anomaly detection using a moving camera is the one described in [8]. Here, a camera is mounted on a robot that performs a translational movement. First, one records a reference video of what is considered a normal situation along the path of the camera. Every new video along the path is compared with this reference to detect abandoned objects. Salient points of reference and observation images are obtained through the SURF algorithm [9]. The videos are registered by means of an homography computed among their frames using the detected salient points and the RANSAC algorithm. The NCC is applied between the two images (the reference video and observation). After some post-processing operation, a threshold determines the areas in the frame that are considered to be abandoned objects. The block diagram of this method is shown in the Fig.2.3.

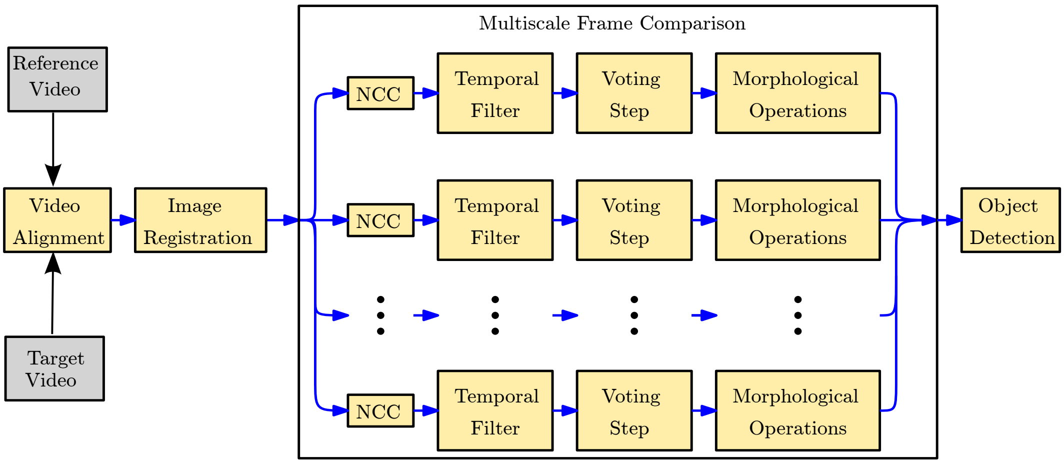


Figure 2.3: The block diagram of the method implemented in [8]. A multiscale system is implemented to detect anomalies in videos obtained from a moving camera in a cluttered environment.

As a result of the initial research, we verified that the STC method constitutes a very interesting approach, with good performance in the task of detecting anomalous events, but it has good performance only in the case of static camera, as will be shown in Chapter 3. So, in this work we developed a new STC-based method capable of detecting abandoned objects in videos obtained from a camera mounted on a moving robot, such as the one used in [8]. Besides that, the new method is also able to detect anomalous events in the static camera case. The database utilized in this thesis was the VDAO [46]. The environment in VDAO is a real industrial plant, cluttered and with lighting variation, what is a challenging scenery in the development of an object detection application. In the next chapters we will present the STC method, the new proposed method and the results obtained when using this new method with the VDAO database.

Chapter 3

Spatio-Temporal Composition Method

In this chapter, we make a brief description of the method proposed in [7] to find anomalous events in videos. We also provide details of our reference implementation of this method. In this method, new samples of video are broken down into small volumes that are represented by codewords from a codebook. Then, the probabilities of occurrence of spatio-temporal compositions formed by these codewords are calculated. Compositions with low probability are candidates to be anomalous. The training is conducted with a small sample video of a normal scene. The initial stages of sampling and creating the descriptors are identical in the training and analysis phases. Fig. 3.1 shows the main steps of the STC method for identifying anomalies in images.

3.1 Features Sampling

The features sampling is based on bag of video words (BOV), consisting of spatio-temporal volumes obtained by random or dense sampling. In the analysis of videos, dense sampling usually has a superior performance, because it is able to maintain the relevant information of a video [47]. In the STC method, dense sampling is carried out, dividing the video into small 3D volumes, $v_i \in \mathbb{R}^{n_x \times n_y \times n_t}$ as shown in Fig. 3.2, where $n_x \times n_y$ is a small frame area and n_t is the length of a small time interval.

In [35] the sampling is performed such that there is an overlap of 50% between adjacent volumes. This leads to satisfactory results, achieving a compromise between accuracy and processing time. In our reference STC implementation, we adopted a spatial overlap of $50\% + 1$ pixel. Fig. 3.2b shows two $3 \times 3 \times 3$ volumes with a 2-pixel overlap.

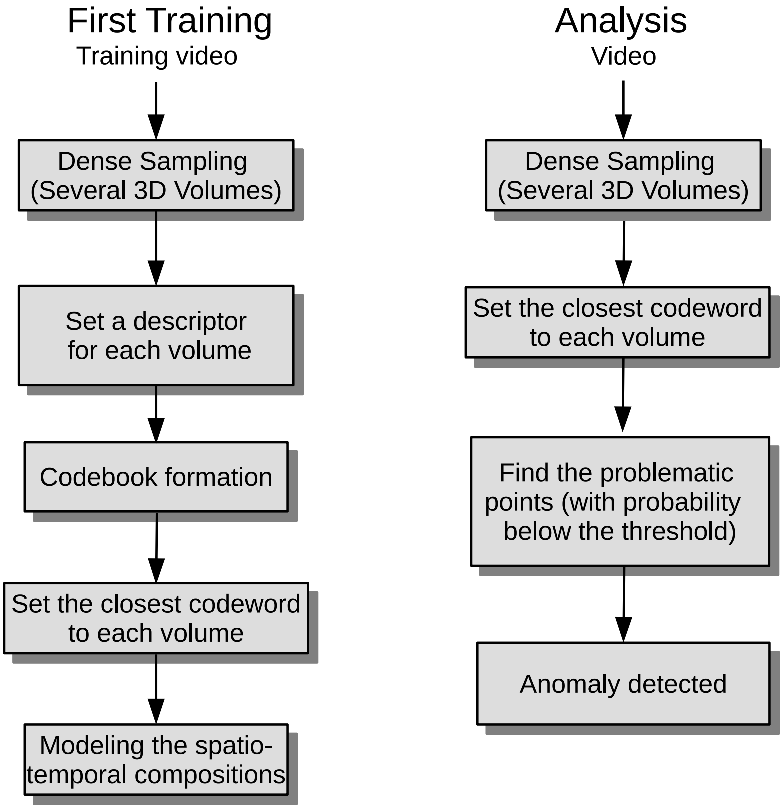

First Training
Training video
Dense Sampling (Several 3D Volumes)
Set a descriptor for each volume
Codebook formation
Set the closest codeword to each volume
Modeling the spatio-temporal compositions
Analysis
Video
Dense Sampling (Several 3D Volumes)
Set the closest codeword to each volume
Find the problematic points (with probability below the threshold)
Anomaly detected

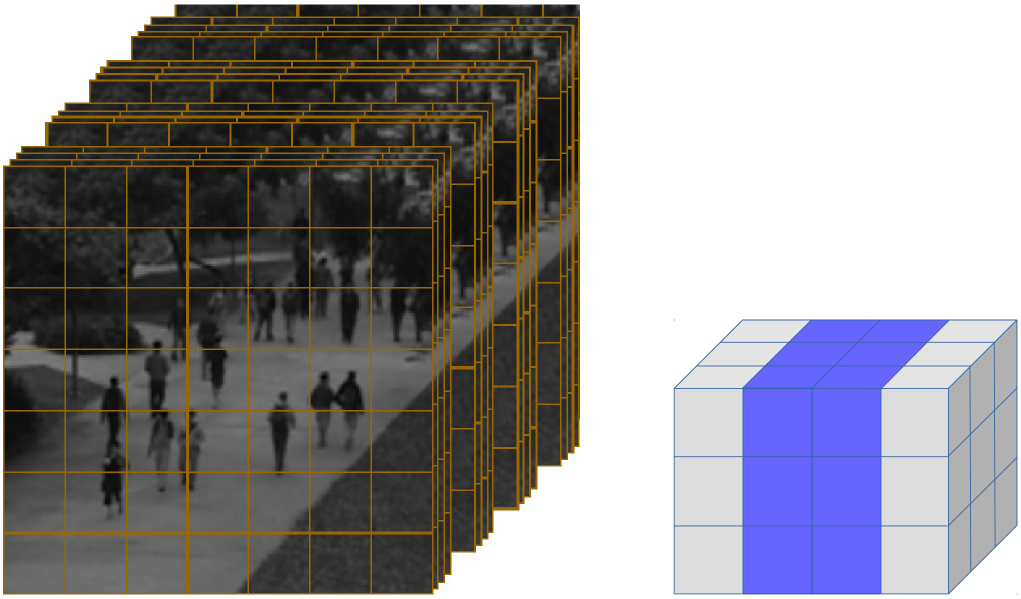
Figure 3.1: The training and analysis steps of the STC method.

3.2 3D-Volume Descriptor

Each volume v_i is represented by a descriptor g_i which is simply the absolute value of the time derivative $\triangle_t$ of each pixel in the volume v_i:

$$\forall v_i, g_i = \text{abs}(\triangle_t(v_i)). \tag{3.1}$$

The values obtained for each pixel of v_i are stacked on a vector, as shown in Fig. 3.3. The volume v_i used in our implementation was of dimensions $7\times7\times5$ pixels, so the descriptor had a dimension of 1×245. These dimensions were empirically defined as in [7]. This descriptor is robust even in a cluttered environment. Although this descriptor works well in many situations, other descriptors may perform better depending on the application; a good example is the one used in [35, 48].



a) Dense sampling

b) Overlap of the volumes

Figure 3.2: a) Dense sampling of a video. b) Two 3 × 3 × 3 volumes with a 2-pixel overlap.

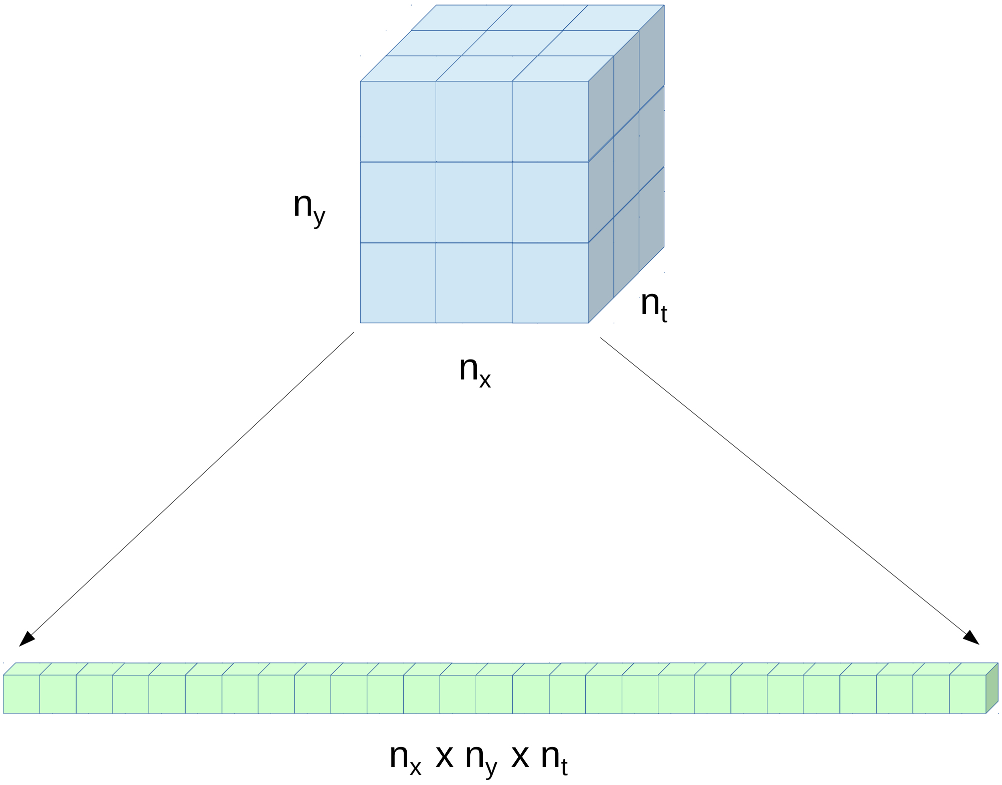


Figure 3.3: The volume v_i is stacked in a vector of size $n_x \times n_y \times n_t$.

3.3 Codebook

Due to the dense sampling, the number of spatio-temporal volumes is too large, and these volumes have a lot of redundancy among them. So, to decrease the complexity, similar volumes are grouped and with each group is associated a codeword from the descriptors of these volumes. The codewords are saved in a codebook. The codebook can be created using a clustering method, such as k-means [29]. In this work, the codebook creation step is carried out using the algorithm described in [7]. Fig. 3.4 shows this algorithm. The only parameter to be set is the maximum distance ε_1 to segregate the codewords from each other. In Chapter 6, the method utilized to find the best value of this distance is described.

```
The training video is broken down into small volumes $v_i$.
$\alpha$ is a percentage of the distance. If two codes are at a distance below $\alpha\varepsilon_1$, they can be merged.
**Initialization**
The first codeword is the first volume $v_0$:
$c_1 \leftarrow v_0$
$f_1 \leftarrow 1$
$P_{t_1} \leftarrow 1$
**Codebook Creation**
The Euclidean distance is used to determine the similarity between the descriptor and the codewords in the codebook
for All volumes $\{v_i\}_{i=1}^{N}$ do
  if $\min_j d(v_i, c_j) > \varepsilon_1$ then
    Create a new code: $c_{j+1} \longleftarrow v_i$
  else
    Calculate $w_{i,j}$ using: $w_{i,j} = \frac{1}{\Sigma_j \frac{1}{d(v_i,c_j)}} \times \frac{1}{d(v_i,c_j)}$
    Update the codebook: $c_j \longleftarrow \frac{f_j \times c_j + w_{i,j} \times v_i}{f_j + w_{i,j}}$
    Update the frequency: $f_j \longleftarrow f_j + 1$
    Calculate the prior probability: $P(c_j) = \frac{f_j}{N}$
  end if
  **Pruning the Codebook**
  for All codewords $\{c_m\}_{m=1}^{M}$ do
    if $\{d(c_i, c_j) < \alpha\varepsilon_1, (0 < \alpha < 1)\}$ and $\{f_j < 0.1 \times \frac{N}{M}\}$ then
      **Merge the 2 codewords:**
      Remove the codewords $c_i$ e $c_j$ from the codebook
      Create a new codeword: $c_{M+1} \longleftarrow \frac{f_i \times c_i + f_j \times c_j}{f_i + f_j}$
      Define the frequency of the new codeword: $f_{M+1} \longleftarrow f_i + f_j$
    end if
  end for
end for
```

Figure 3.4: Codebook Creation algorithm.

After creating the codebook, a code is allocated to each volume of the training image. The criterion for this allocation was the smallest Euclidean distance. After creating the codebook, each volume v_i is related to each codeword c_j with a weight $w_{i,j}$ given by

$$w_{i,j} = \frac{1}{\mathit{\Sigma}_j \frac{1}{d(v_i,c_j)}} \times \frac{1}{d(v_i, c_j)}, \tag{3.2}$$

where $d(v_i, c_j)$ is the Euclidean distance between the volume v_i and the codeword c_j.

Fig. 3.5 shows an example of the initial steps of creating the codebook for a 2D descriptor. The first codeword of the codebook is the descriptor of the first sample volume v_i of the training video. If the next sample distance to the first codeword is greater than a similarity distance ε_1, a new codeword is created with the value of this new sample. If the sample distance is smaller than ε_1, the codeword is updated using $w_{i,j}$, so that the new value of the codeword is the center of mass of the samples inside the similarity circle, as show in Fig. 3.6. The process goes on with each new sample being compared with all codewords in the codebook. In the end all the samples associated with a given codeword will be inside a similarity circle. In the example of the Fig. 3.6, given a distance ε_1, it took seven codewords to represent all the samples. The greater the distance ε_1, the lower the required number of words in the resulting codebook.

3.4 Spatio-Temporal Composition

Most methods using BOV do not take into account the spatio-temporal arrangement between the volumes or limit it to a small volume around the sampling point. In STC, a probabilistic approach is used to determine whether the volume is anomalous or not, based on the probability of the arrangement of the volumes within a larger region.

The representation of the set is made as follows: let E_i be the ensemble centralized at the point (x_i, y_i, t_i) in absolute coordinates and containing K volumes. This central point is used to determine the relative coordinates of the position of the volumes within the ensemble, according to Fig. 3.7a. Given the volume v_k in the set E_i, $\Delta_{v_k}^{E_i} \in \mathbb{R}^3$ is the relative position (in space and time) of v_k located at the point (x_k, y_k, t_k) within E_i:

$$\mathit{\Delta}_{v_k}^{E_i} = (x_k - x_i, y_k - y_i, t_k - t_i). \tag{3.3}$$

Thus, the ensemble of volumes E_i, centered at position (x_i, y_i, t_i), is initially

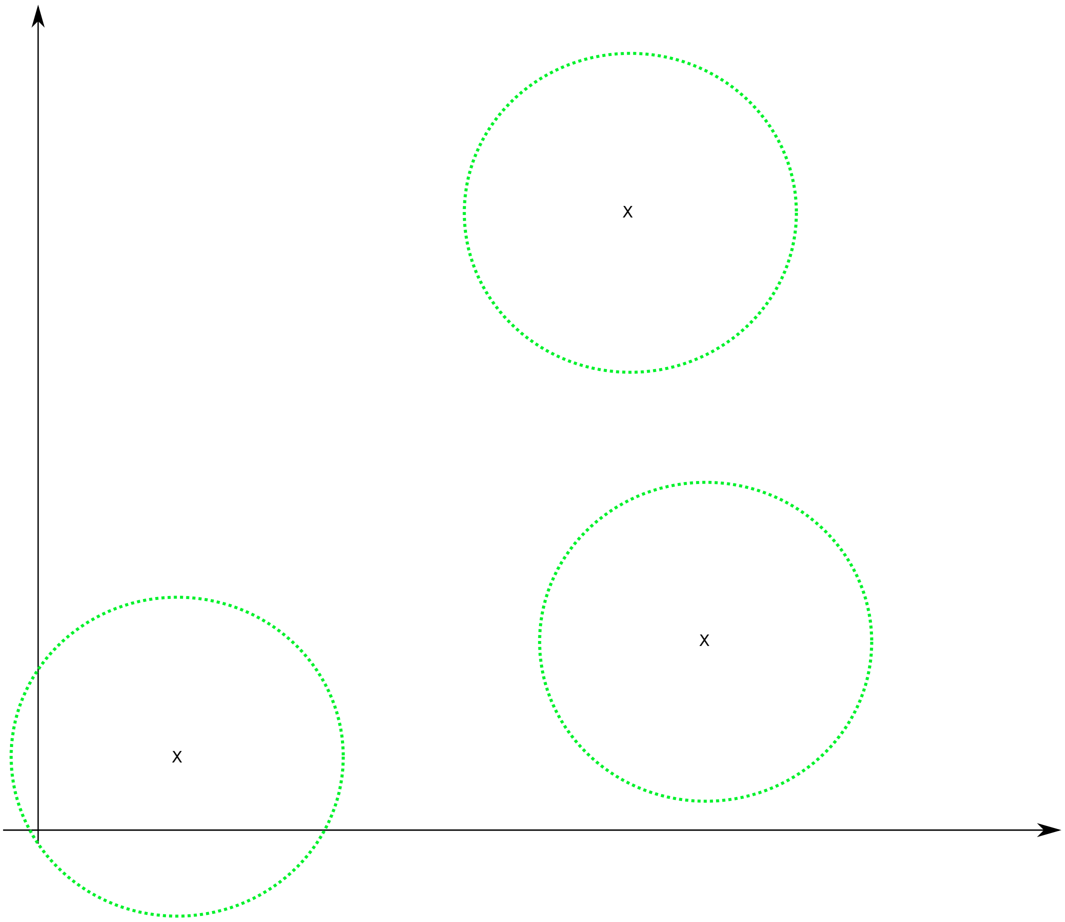


Figure 3.5: Example of the creation of the codebook of a 2D descriptor. All the three samples are at a distance greater than the similarity distance ε_1, so three new codewords are created.

represented as a set of video volumes and their relative positions with respect to the central volume:

$$E_i = \{\varDelta_{v_k}^{E_i}, v_k, v_i\}_{k=1}^{K}. \tag{3.4}$$

Each volume v_k of the set is linked with the codeword $c_j \in \mathbf{C}$ with a weight w_j, representing their similarity. Thus, the arrangement of volumes may be represented by a set of codewords and their spatio-temporal arrangement. Let $\nu \subset \mathbb{R}^{n_x \times n_y \times n_t}$ be the spatio of descriptors of a video volume, and $\mathbf{C}$ the codebook; $c : \nu \rightarrow \mathbf{C}$ defines a random variable that allocates a codeword to a volume of video and $c' : \nu \rightarrow \mathbf{C}$ defines a random variable designating a codeword to the volume in the center of the ensemble. In addition, $\delta : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ defines a random variable representing the relative distance from the central point associated with codeword c' to the point associated with codeword c. Therefore, the ensemble of volumes can be represented as an arrangement of words of the codebook, as shown in Fig. 3.7b. In other words, instead of representing the E_i as an arrangement of volumes, it is represented as a

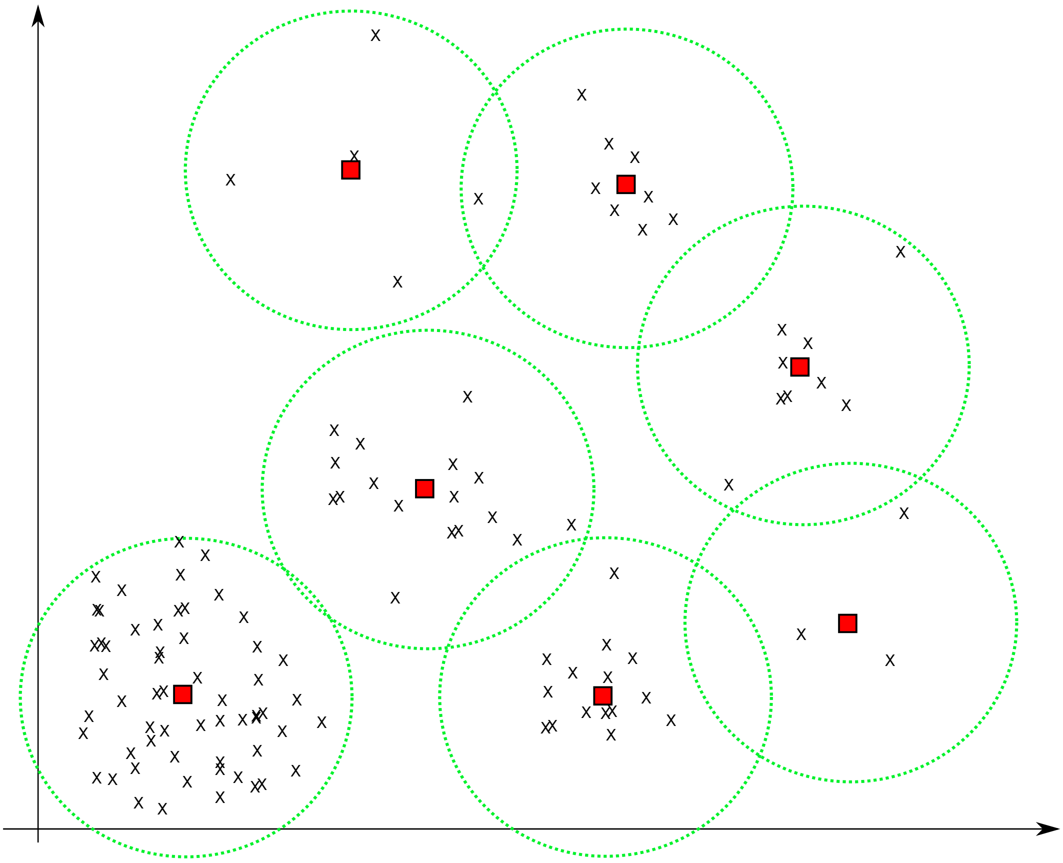


Figure 3.6: As the samples are processed, they are grouped, and the final codeword is the mass center of the samples inside the circle. The squares represent the final codewords.

codeword arrangement.

In this context, $O = (v_k, v_i, \Delta^{E_i}_{v_k})$ represents the observation of the volume v_k from the central volume v_i in the ensemble E_i, and $\Delta^{E_i}_{v_k}$ the relative position of the observed volume v_k with respect to v_i within E_i. The goal is to measure the probability $P(h|O)$ of each hypothesis $h = (c, c', \delta)$ obtained by replacing the volumes by codewords from the codebook, given the observation O, that is

$$P(h|O) = P(c, c', \delta \mid v_k, v_i, \Delta^{E_i}_{v_k}). \tag{3.5}$$

In [7], it is shown that

$$P(c, c', \delta \mid v_k, v_i, \Delta^{E_i}_{v_k}) = P(\delta \mid v_k, v_i, \Delta^{E_i}_{v_k}) P(c' \mid v_i) P(c \mid v_k). \tag{3.6}$$

Hence, in an ensemble around a pixel, with a central volume v_i, and other volumes v_k within this ensemble at a distance $\Delta^{E_i}_{v_k}$ of the central volume, the aim is to calculate the probability of assigning the codeword c' to the central volume and c to the other volumes. The probability $P(\delta \mid v_k, v_i, \Delta^{E_i}_{v_k})$ is determined by the approximation of its pdf by a mixture of Gaussians using the expectation maximization

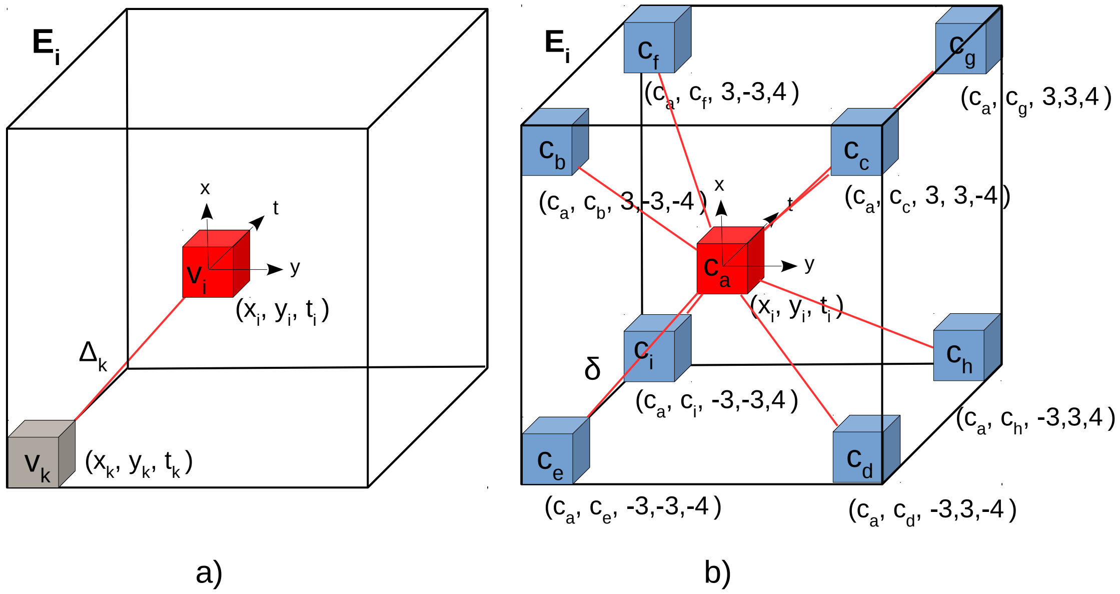


Figure 3.7: a) Relative position of the volumes in the set. The central volume v_i is at a distance Δ_k of the volume v_k. b) After the substitution of the volumes by the closest code, the set is represented by spatio-temporal arrangement of codewords, which are at a distance δ of the central codeword $c' = c_a$, in this example. Only the volumes in the corners are represented at distance $\delta = \sqrt{34}$, contained in an ensemble of $7 \times 7 \times 9$ volumes.

algorithm (EM) [31], where the samples are the codeword arrangements. In other words, the sample vector is of the form $\mathbf{a}(c_i, c_k, \delta)$, where δ is the relative distance between codewords. Several of these samples allow us to estimate the pdf, as shown in Fig. 3.8. The probabilities $P(c' \mid v_i)$ e $P(c \mid v_k)$ of each spatio-temporal volume are calculated during the allocation of codewords.

Therefore, in this method the dictionary is effectively formed not only by words but also by the distribution of the probabilities of the arrangements around each volume in the ensemble.

To calculate the parameters of the mixture of Gaussian using the EM algorithm, the number of Gaussians used in our implementation was three. The samples used to find the parameters form a vector of dimension 1×5, composed as follows: let E_i be the ensemble with a central volume v_i at the position (x_i, y_i, t_i) in absolute coordinates and containing K volumes. The relative coordinates (x_k, y_k, t_k) of the neighboring volumes v_k inside the ensemble are calculated from this point. The

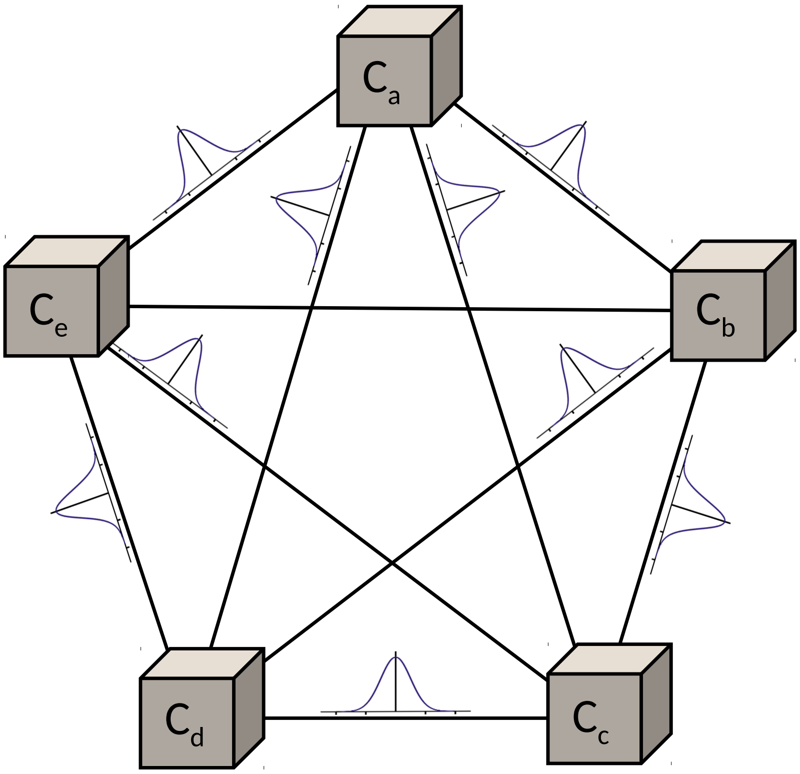


Figure 3.8: The pdf of the 3D composition of the volumes v_k with the associated codewords c_k inside the ensemble E_i is calculated. The codebook is effectively formed by the codewords and the pdf.

volume v_i is represented by a codeword c_i, and v_k by a codeword c_k. Let j_{c_i} be the index of c_i in the first codebook and j_{c_k} be the index of c_k. Using these definitions, the first element of the sample is the index j_{c_k} of v_k, the second is the index j_{c_i} of v_i and the last three elements of the sample are the relative coordinates (x_k, y_k, t_k). Therefore, the pdf is calculated using the relative position inside the ensemble and the codewords, represented by their index in the codebook.

3.5 Anomalous Pattern Detection

In the analysis phase, the steps of sampling and descriptor calculation are the same as in the training phase. Then, using the codebook created in the training phase, the distance between the volume and every codeword is computed using Eq. (3.2).

Next, the Eq. (3.6) is used to calculate the codeword-assigning probability of one volume v_k, that is independent of the relations between the central volume and the other volumes v_k in the ensemble E_i.

Given a video of interest V, E_i^V is an ensemble of video volumes centered at point (x_i, y_i, t_i) and v_i is the central volume of this ensemble. The probability of the volume v_i can be written as:

$$P(c, c', \delta \mid E_i^V) = \prod_k^K P(\delta \mid c, c', \Delta_{v_k}^{E_i^{\mathrm{V}}}) P(c \mid v_k) P(c' \mid v_i), \tag{3.7}$$

where v_k is a volume inside E_i^V, $\Delta_{v_k}^{E_i^{\mathrm{V}}}$ is the relative position of the volume v_k, c' is the codeword attributed v_i, c is the codeword attributed to v_k and δ is the relative distance of the codeword, in the codeword space. The term $P(\delta \mid c, c', \Delta_{v_k}^{E_i^{\mathrm{V}}})$ is the probability of the spatio-temporal arrangement, whose pdf is calculated as given in Section 3.4.

The a posteriori probability is calculated according to

$$P(c_j \mid v_i) = \frac{w_{i,j} \times P(c_j)}{\sum_j w_{i,j} \times P(c_j)}, \tag{3.8}$$

where the weight $w_{i,j}$ is given by Eq. (3.2). Each sample v_i is represented by a codeword of the codebook and also has a probability which is a function of the distance of the descriptor of v_i to the closest codeword. The closer v_i is to this codeword, the higher its probability.

In brief, in the STC method the video V to be analyzed is densely sampled into video volumes v_i. For each v_i a codeword $c_j \in C$ is allocated with a similarity $w_{i,j}$. The probability $P(c, c', \delta \mid E_i^V)$ of each volume to be an anomaly is calculated based on the spatio-temporal arrangement of the volumes within the ensemble E_i^V, centered in v_i.

For each volume the probability of occurrence is computed using Eq. (3.7). Volumes with a probability smaller than a given threshold, obtained experimentally, are considered anomalous. The detection of an anomaly is based only on this threshold. The Fig. 3.9 illustrates the use of the threshold. Ideally, only the anomalous points have a probability below the threshold.

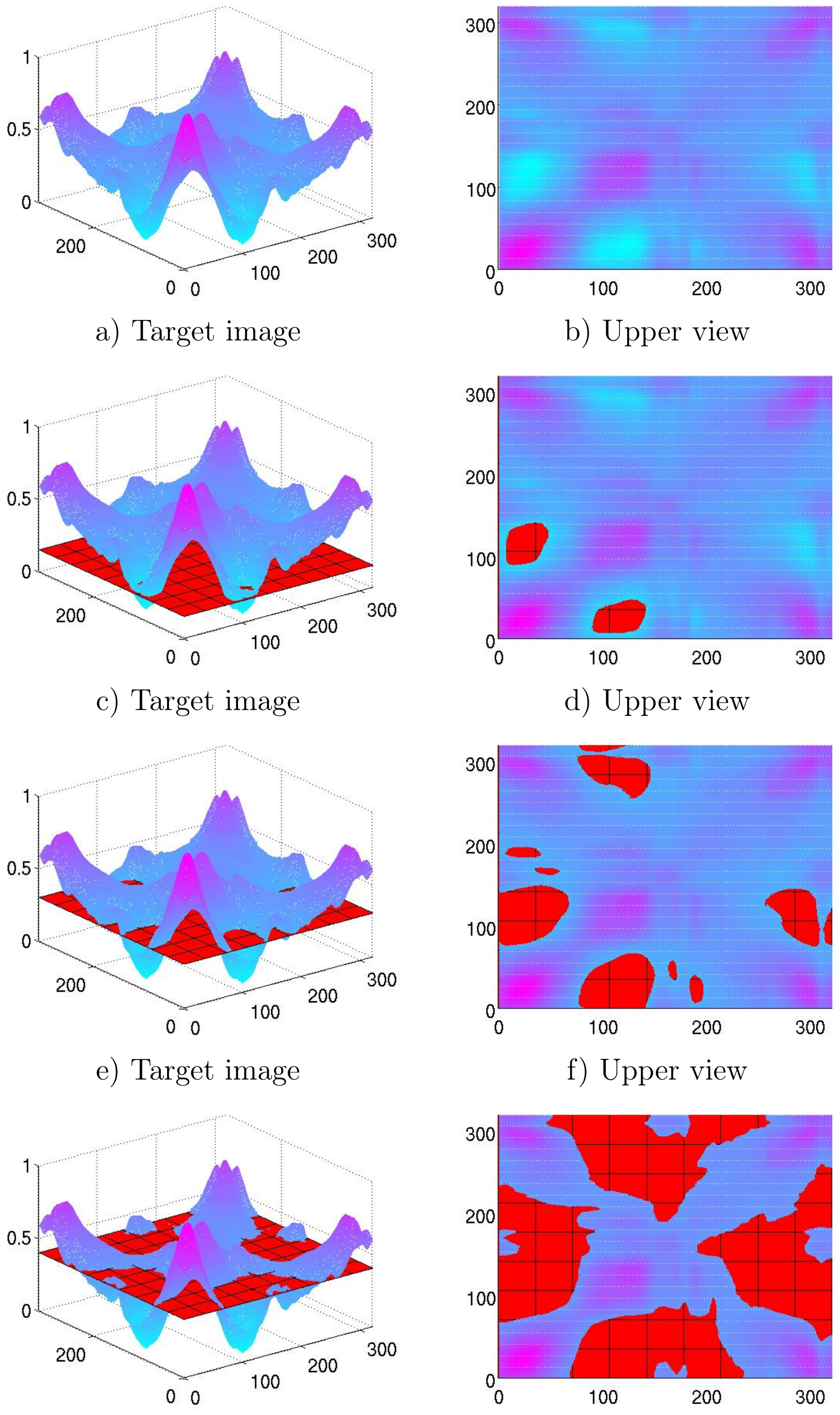
3.6 Results and Conclusions

To illustrate the performance of the STC method, simulations were performed using the UCSD database, available in [49]. Initially, the training was made with a short video of about ten seconds where there were just persons walking. In the test video there were people walking too, but there was also an anomaly consisting of a person riding a bicycle. The results obtained for the first test video are shown in Fig. 3.10.

We can notice that STC performed well, since only the cyclist was detected. In the Fig. 3.11, the anomalies were the cart and the cyclist. Again, both were well detected, without false identification. This level of performance was obtained for several other videos in the same database.

The STC method has also been applied to the VDAO database [12] (see Chapter 5). The training phase has been performed using a reference video of the environment recorded from a camera mounted on a moving robot that performs a back and forth rectilinear movement. The test was performed using a video recorded in similar conditions, but with abandoned objects added to the environment. Figs. 3.12 to 3.15 illustrate the performance when detecting a whisky box, a sneaker, a bowl and a towel, respectively, over a cluttered background. One can note that STC fails in these cases, generating many false detections. We have observed in our experiments that if the detection threshold is modified in order to avoid the false detections, STC is not able to detect the abandoned objects.

The results in Figs. 3.10 and 3.11 suggest that the STC method is able to detect anomalies when there are movements in the scene, and the camera is static. However, it fails in the VDAO database, that concerns detection of abandoned objects over a cluttered background using a moving camera. One of the main purposes of this thesis is to investigate a solution to this problem. This is done in the next chapter, where we propose the novel algorithm STC-mc (STC-moving camera).



a) Target image

b) Upper view

c) Target image

d) Upper view

e) Target image

f) Upper view

g) Target image

h) Upper view

Figure 3.9: Example of the use of the probability threshold. In the figures the function $P(i,j)$ is the probability of the image points $I(i,j)$. Blue values represent points of low probability and purple ones represent points of high probability. Figures a) and b) represent a map of the probability of a sample frame. These figures have been cut by a plane representing the identification threshold. The higher the threshold, the more points are identified as being possibly anomalous. The points that are candidates to be anomalous are represented in red.

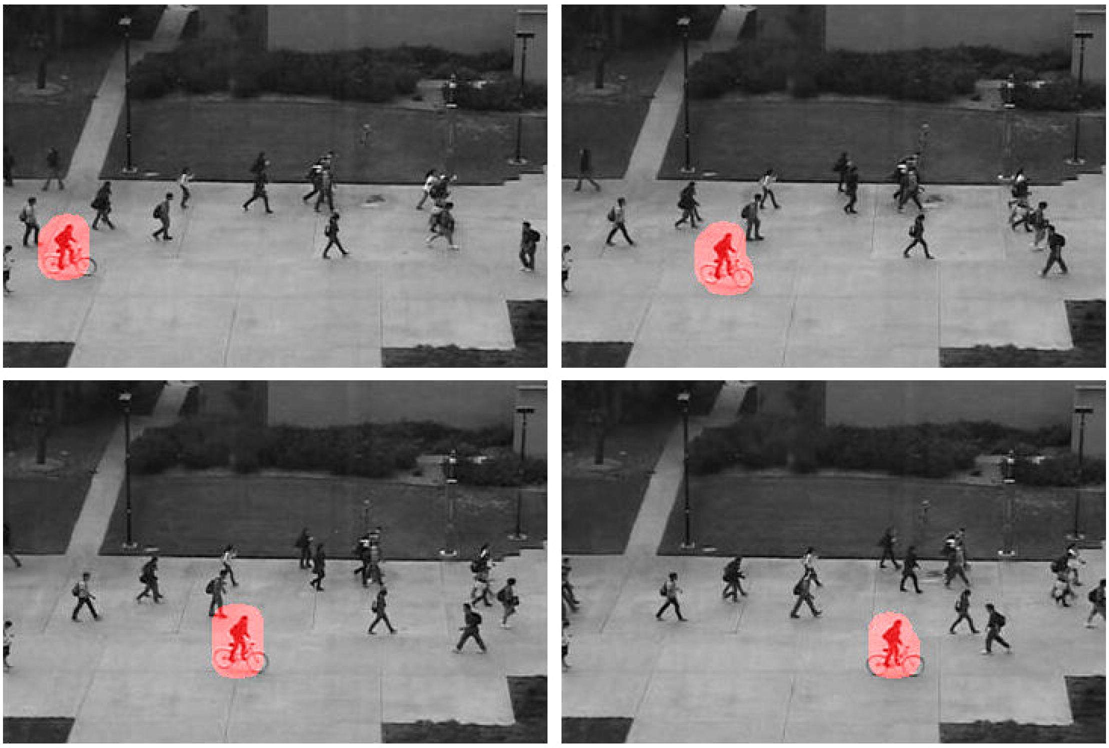


Figure 3.10: Only the cyclist was detected. The people walking were not detected because in the training video there were people walking in a similar way.

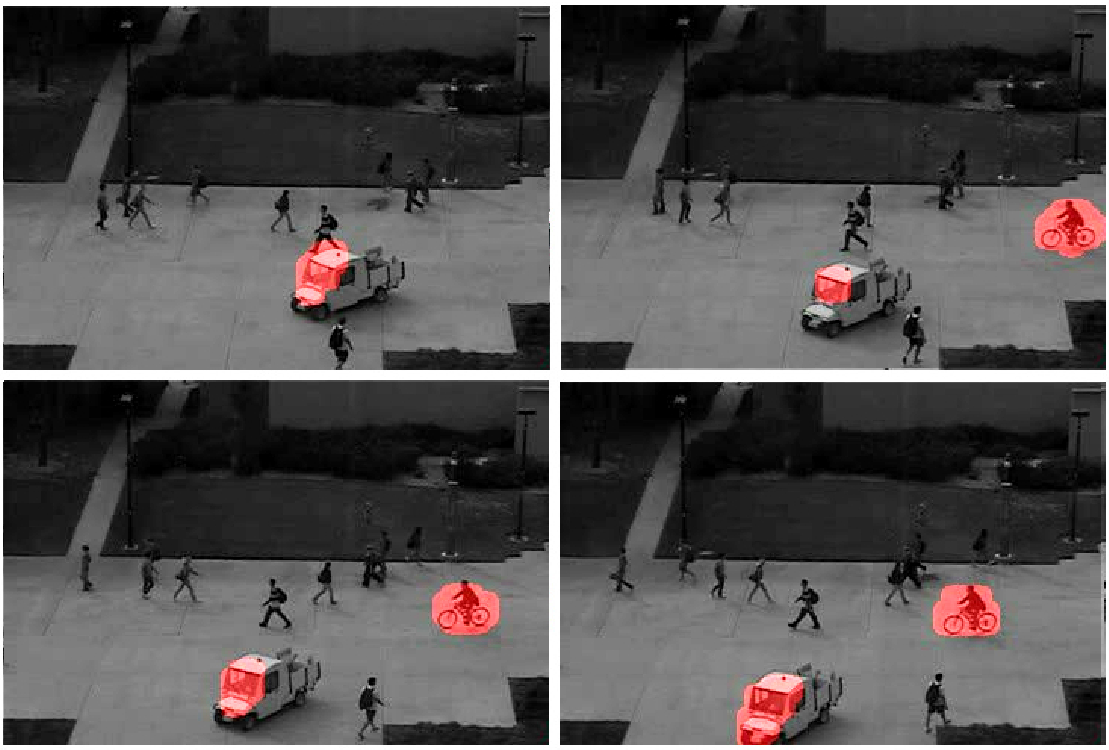


Figure 3.11: Only the cart and the cyclist were detected. The people walking were not detected because in the training video there were people walking in a similar way.

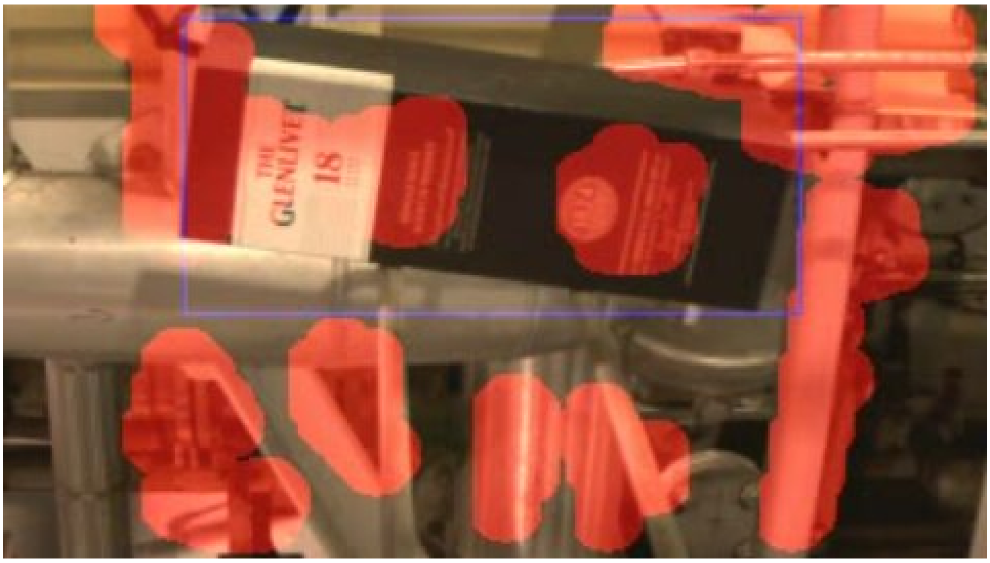


Figure 3.12: Example of results obtained with a moving camera when the abandoned object was a whisky box. In most of the frames, it was not possible to find a threshold where only the abandoned object was detected.

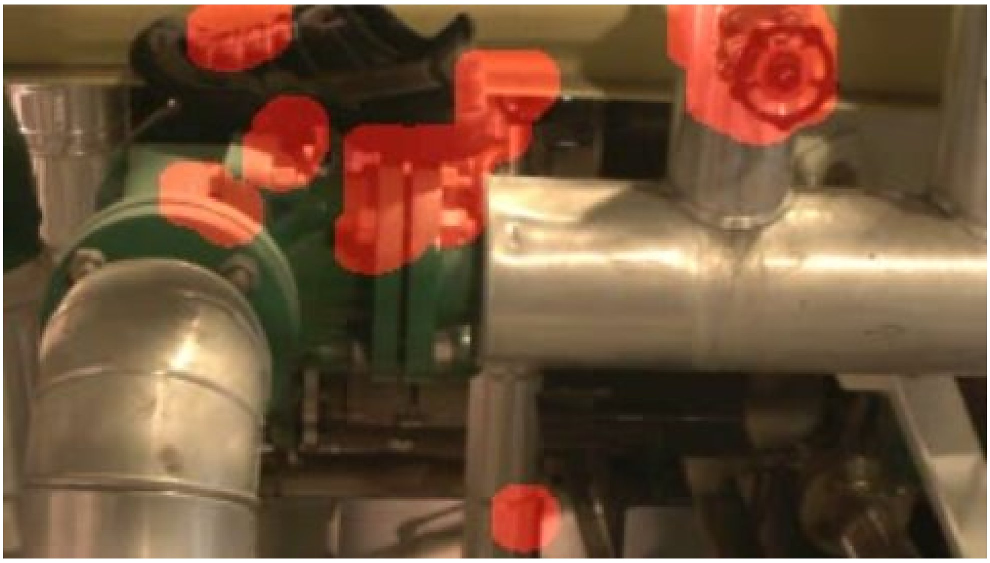


Figure 3.13: Example of results obtained with a moving camera when the abandoned object was a sneaker. In most of the frames, it was not possible to find a threshold where only the abandoned object was detected.

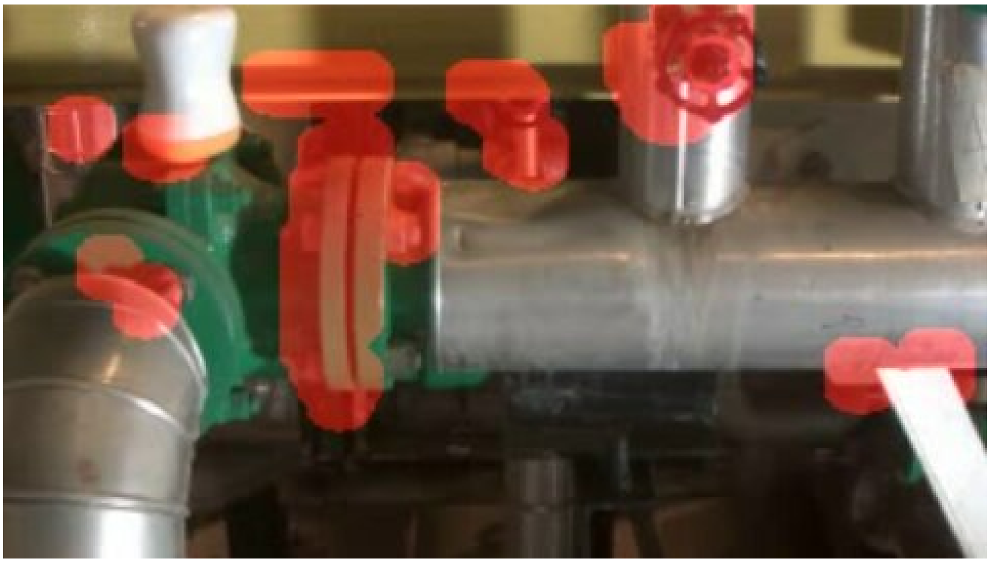


Figure 3.14: Example of results obtained with a moving camera when the abandoned object was a bowl. In most of the frames, it was not possible to find a threshold where only the abandoned object was detected.

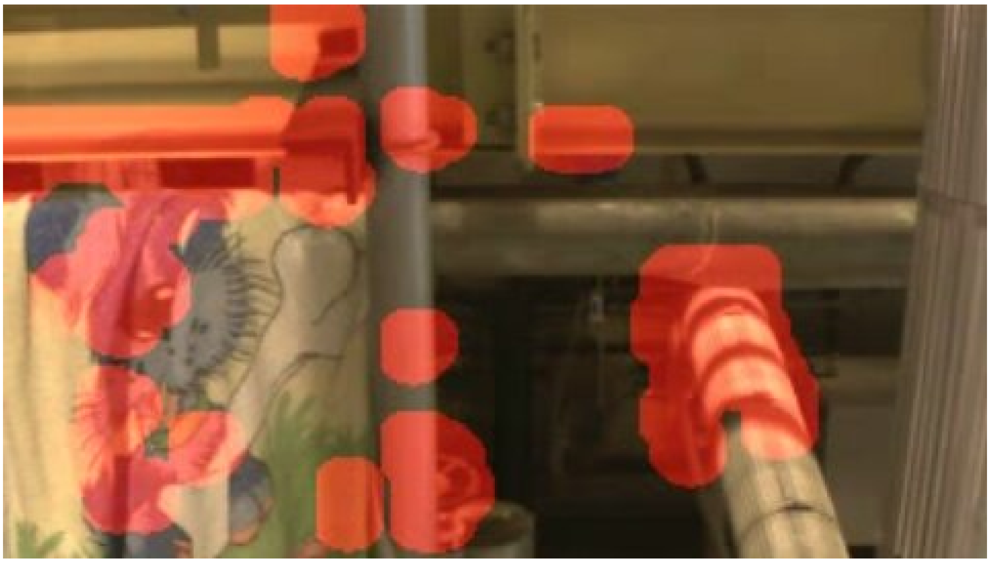


Figure 3.15: Example of results obtained with a moving camera when the abandoned object was a towel. In most of the frames, it was not possible to find a threshold where only the abandoned object was detected.

Chapter 4

Proposed Methodology to Detect Anomalous Events Using a Camera Mounted on a Moving Platform

In this chapter, we propose the STC-mc (STC-moving camera) algorithm. It is a new anomaly detection algorithm based on the same principles of the STC but with enhancements that allow it to perform well in the detection of abandoned objects using videos acquired from a camera mounted on a moving platform. Fig. 4.1 shows the main steps of STC-mc, highlighting its main contributions relative to the original STC. The steps corresponding to blocks in gray are the same as the ones in the STC method and the steps surrounded by a dotted rectangle represent the STC steps that have been enhanced in the STC-mc. The steps corresponding to the white blocks are not present in STC, being entirely proposed in this work. In what follows these blocks are described in detail.

The main modification is that now the training has two phases, generating a second codebook. This new codebook is necessary because there are situations when the probability of the abandoned object can be higher than some points in the background. For example, an abandoned piece of pipe could be less noticeable than a hydraulic valve, even if the valve was in the training video. This is because the valve has many edges that are detected by the spatio-temporal derivative, and this configuration is uncommon to repeat. Therefore, it is necessary to create a codebook of the descriptors of the background points where the probability is below a threshold in the reference video. This codebook is used to exclude these points from the target video, avoiding false detections.

Besides that, a new spatio-temporal descriptor is used. In the STC method a temporal gradient is used to construct the descriptor, but better results are obtained with the use of a spatio-temporal gradient, especially in the case where the anomaly to be detected is a abandoned object. The spatio-temporal descriptor is also less sensible to variations in the movement of the camera, like shaking or an inconstant moving speed.

But only the introduction of the spatio-temporal derivative in the descriptor is not able to eliminate the effect of the variations in the movement of the camera. So, a more specific solution had to be developed, with the introduction of a Gaussian filtering to deal with misalignments caused by the camera shaking.

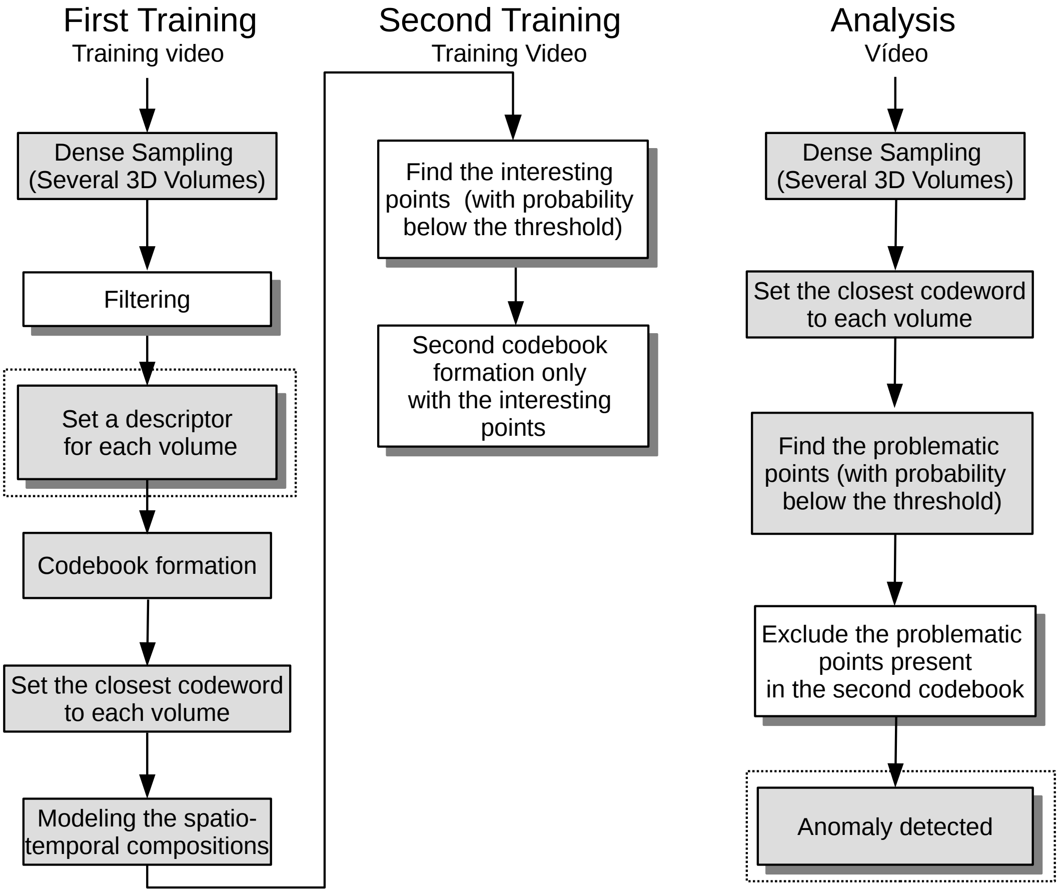

First Training
Training video
Dense Sampling (Several 3D Volumes)
Filtering
Set a descriptor for each volume
Codebook formation
Set the closest codeword to each volume
Modeling the spatio-temporal compositions
Second Training
Training Video
Find the interesting points (with probability below the threshold)
Second codebook formation only with the interesting points
Analysis
Vídeo
Dense Sampling (Several 3D Volumes)
Set the closest codeword to each volume
Find the problematic points (with probability below the threshold)
Exclude the problematic points present in the second codebook
Anomaly detected

Figure 4.1: The training and analysis steps of the proposed method. The gray blocks are part of the original STC method. The gray blocks surrounded by a dotted rectangle represent the STC blocks where enhancements are proposed in this thesis. The white blocks represent novel steps proposed in this thesis.

4.1 Programming Tools

For programming, we used the QT [50] cross-platform application framework. This framework is free for non-commercial purposes, has an intuitive interface and allows visual programming. It also has a lot of code examples and communities on the Internet. The utilized programming language was C++ [51]. As video manipulation programs demand a lot of processing time, it is necessary to use compilers that generate native code, not interpreted. OpenCV library [52, 53] was utilized to help in the programming. OpenCV is an open source library with functions that implement computer vision algorithms. The operating system used was Ubuntu 14.04.

4.2 Second Codebook

When applying the STC method to the videos in the VDAO database, often it is not possible to find a threshold which allows that only anomalous events have a probability value below it. This produces a large number of false detections. This is so because in the VDAO database there are situations when the probability of an anomaly (abandoned object) can be higher than the probability of some points in the background. In the STC, the only criteria to consider a small volume of video v_i as an anomaly is its probability. This probability is calculated using the codebook created in the training phase. Similar volumes are represented by the same codeword of this codebook, and each codeword has a probability computed based on the number of occurrences in the training. The higher the number of occurrences of this codeword, the greater the probability. In the analysis phase, a representation of the video is constructed replacing the volumes by the closest codeword. Suppose that in a training video there are a lot of pipes in the background, and only a small valve is present in this video. In the training phase, after the dense sampling, the codebook will be formed by many codewords similar to a piece of pipe and with high probability. Only a few codewords will be similar to a piece of the valve, and with low probability. In the analysis phase, if the abandoned object is similar to a piece of pipe, its representation will be constructed with codewords with a high probability, because the codebook was formed mainly from pipes. So the pieces of the abandoned object will have a high probability too, higher than the valve that already was in the training video. Besides that, to detect the abandoned object, the threshold should be set to a value higher than the probability of this object. But the valve has a probability smaller then this abandoned object, so it will be detected too. Therefore, in an ideal situation, if the first training is performed with a video, and after that an analysis is performed using this same video, no points should be detected as anomalies. But when the STC is used to detect abandoned objects, this ideal behavior does not happen. Fig. 4.2 shows an example of the result obtained when the analysis is performed in the same video that was used in the first training stage. Some points are marked as anomalies in an improper manner.

So, in order to solve this problem, we propose a two stage training process for the STC-mc. In it, a second codebook is introduced, containing the descriptors of the points where the probability is below a threshold. This additional codebook is used

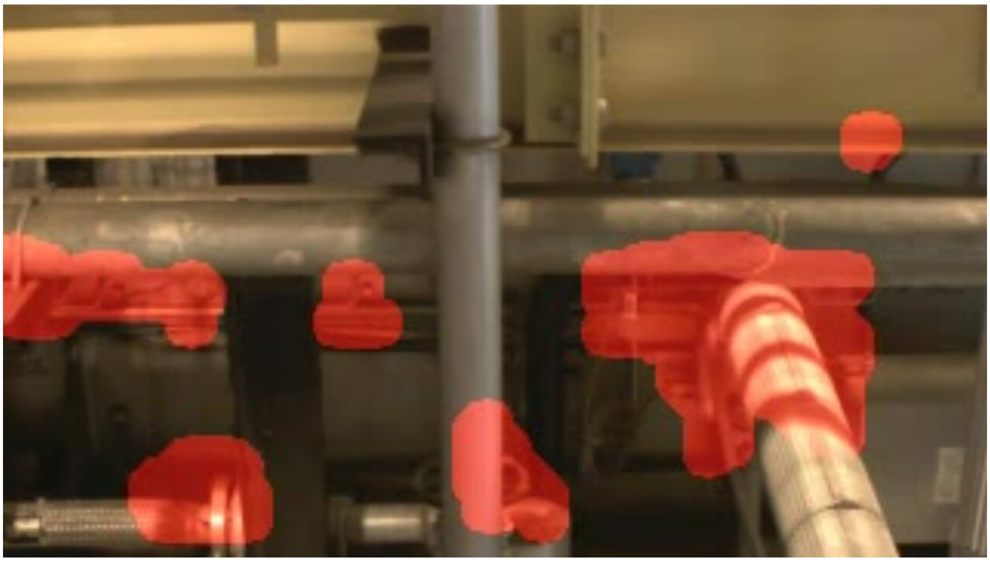


Figure 4.2: Several false detections occur when the analysis is performed in the training video, after the first training stage.

to exclude these points from the target video, so avoiding false detections. This is performed as follows. The main STC codebook is generated during the first training phase, by processing the reference video, as in the original STC method. Then the reference video is processed again, this time to detect abandoned objects. Since obviously the reference video has no abandoned objects, ideally there should be no detections. However, this is not the case. There are several false detections along the reference video. Then, these volumes that have been wrongly detected, that is, the ones with their probability below the initial threshold, have their descriptor and its probability saved in a second codebook. The saved codeword is formed by the descriptor and its associated probability. Then, when processing a video for detecting abandoned objects, a point is considered anomalous only if it has a probability below the threshold and its descriptor and probability are not close to a codeword in the second codebook. This way the false detections are eliminated while keeping the capacity to detect the true anomalies.

After the second training and creation of the second codebook, false detections are discarded and the result obtained when the analysis is made again in the training video of Fig. 4.2 is shown in the Fig. 4.3. No point is detected as been anomalous, as expected.

Fig. 4.4 describes the steps to create the new codebook, where γ is the first-stage threshold and ε_2 is the maximum distance to consider a descriptor as similar to a codeword from the second codebook.

This second codebook must incorporate all the training-video points to be excluded during the anomaly detection phase. This is so because the differences among the probabilities of the volumes tend to be low. Therefore, the training should be

Figure 4.3: After the second training, the analysis is performed in the training video without false detections as desired.

```
Second codebook generation during the second stage training
The Euclidean distance is used to determine the similarity between the descriptor and the codewords in the codebook.
The probability of each volume is calculated in the first training pass.
$\gamma$ is the probability threshold and $\epsilon_2$ is the maximum distance between descriptors that are considered as similar.
for All volumes $\{v_i\}_{i=1}^{N}$ do
  if $p_i < \gamma$ then
    if $\min_j d(v_i, c_j) > \varepsilon_2$ then
      Create a new code: $c_{j+1} \longleftarrow v_i$
      Create a new code probability: $p_{c_{j+1}} \longleftarrow p_i$
    else
      if $p_i > p_{c_j}$ then
        $p_{c_j} \leftarrow p_i$
      end if
    end if
  end if
end for
```

Figure 4.4: Second codebook creation algorithm.

done using a reference video including the full surveillance path of the robot. Thus, the number of codewords generated is high, about 15 new codewords per frame. To speed up the search in the codebook, a paging scheme is employed. Every 50 frames a new page is created, with only the codes generated in these 50 frames. So, the search in the codebook is faster, but the drawback is that the frames of the reference and target video must be roughly synchronized. This synchronization does not have to be exact, because the dictionaries do not take into account the position of the volumes in time, but the scene in the frame cannot change much.

The proposed second codebook works well when the video of interest is similar to the training video. Performance variation could happen if there are many differences such as the ones caused by illumination changes, position of the camera, velocity of the robot, etc.

4.3 Probability Threshold

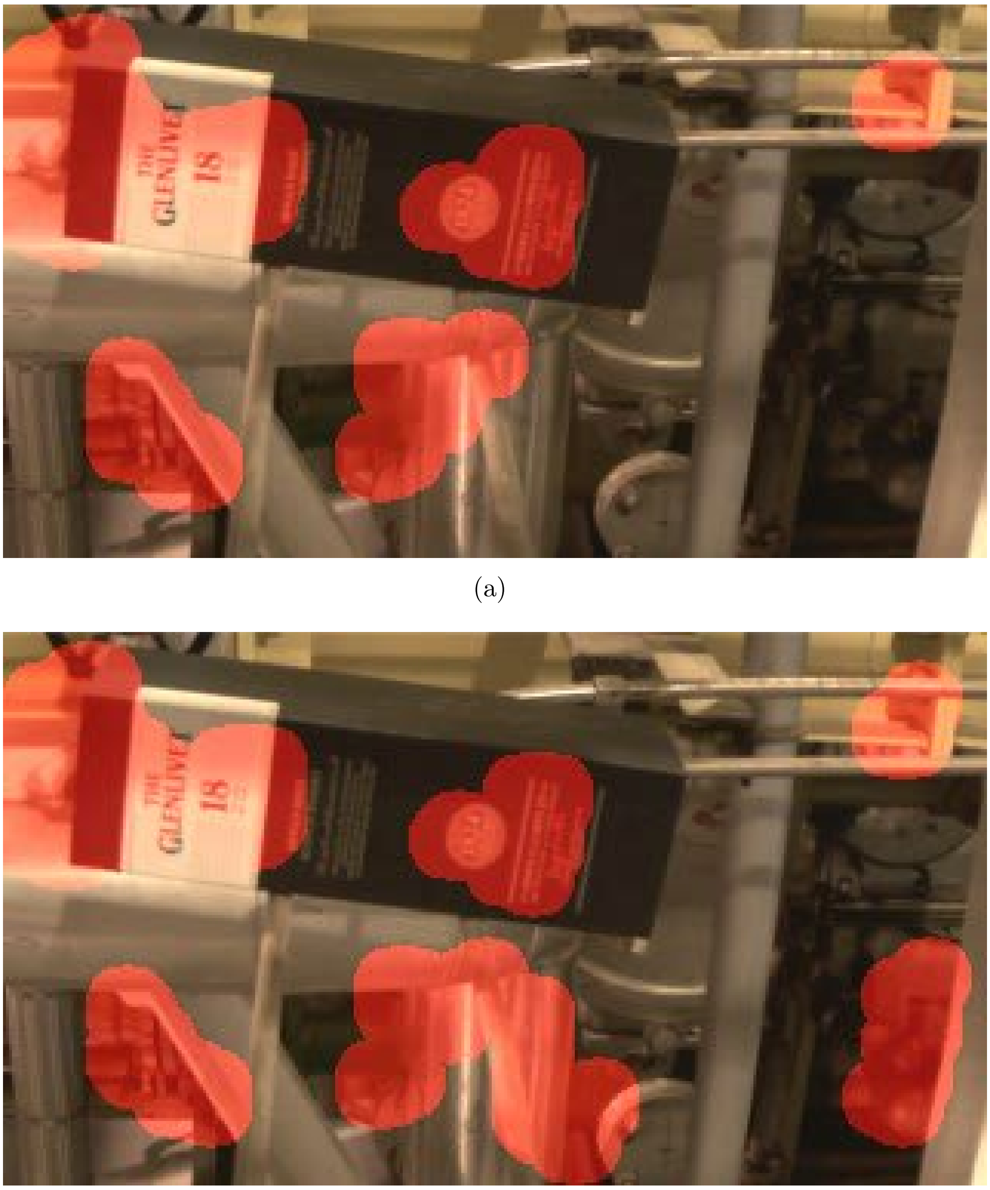
In the STC method, the probability threshold γ directly influences the classification of a volume as anomalous or not. In this work, the threshold is also used to choose the points of the second dictionary. Only the points with a probability below the threshold γ are used to create the second codebook of points of the background with low probability, as explained in the previous section. The strategy to find a good threshold value was first to perform a normal training using a video without the abandoned object. Then several simulations were realized using a video where there was an abandoned object, varying the threshold, until the abandoned object was detected. With the abandoned object, usually several false anomalies are also detected. With this threshold, an analysis was realized with the same training video, without the abandoned object. The points that are detected are saved in a new codebook, used to exclude these points in the next analysis. After several tests, the chosen probability threshold had a value of $\gamma = 1 \times 10^{-7}$. If a higher value of threshold is used, more points are below this threshold and consequently more points are used to create the second codebook in the second stage, and the computational complexity increases. If a lower value is utilized, some points of the anomaly can be discarded, and the performance can decrease.

4.4 Enhanced Spatio-Temporal Descriptor

As described in Section 3.2, the sampling of the video content is based on the bag of video words (BOV) method, which consists of spatio-temporal volumes obtained through dense sampling. The next step is to create a codebook in order to reduce redundancy between video volumes. For this, the video is divided into small 3D volumes, where $n_x \times n_y$ is a small area and n_t is the length of a small time interval. Each volume v_i is represented by a descriptor. Initially, several tests were realized to determined the best descriptor to be used to improve the performance of the STC Method.

4.4.1 Initial Tests of the Descriptors

The first test was performed using the same descriptor of the original STC method, calculated using Eq. (3.1). The tests using this descriptor gave bad results, always with some false detections. The results were very sensible to variations in the threshold γ value. Small changes in the value of the threshold could modify the results. One possible cause of this is that with only the temporal derivative, the system is less capable to detect the variations from one volume v_i to the other. Besides that, the results oscillate, with the points detected varying from one frame to the next, as shown in Fig. 4.5. To choose a good descriptor for our application, a key issue to be taken into account is that, although the camera moves, the abandoned objects



(a)

(b)

Figure 4.5: Using the descriptor of the Eq. (3.1), the results vary from one frame to the next.

do not move relative to the background. Thus, the relative motion of the objects in the scene was only caused by the parallax shift caused by the movement of the robot. Since the parallax shift is a spatio-temporal effect, a descriptor based only in the temporal derivative as used by [4] and [7] is not the most adequate. Therefore, the next test was performed using a descriptor which is based on a spatio-temporal derivative:

$$\mathrm{D} = \sqrt{\left(\frac{dI}{dx}\right)^2 + \left(\frac{dI}{dy}\right)^2 + \left(\frac{dI}{dt}\right)^2}. \tag{4.1}$$

The detection results using this descriptor are shown in Fig. 4.6. Although more points are detected as being anomalous, the results are more stable, with little variation from a frame to the next one. A more stable result is better to investigate possible further improvements, because the result of each change can be measured with higher accuracy.

Some tests were performed using a histogram of oriented gradients (HOG) as descriptor too. In [35, 36] this kind of descriptor was robust to changes in illumination and in many situations had a better performance than optical flow and spatio-temporal gradients. For each volume v_i, the gradients $G_x = \left(\frac{dI}{dx}\right)$, $G_y = \left(\frac{dI}{dy}\right)$, $G_t = \left(\frac{dI}{dt}\right)$, the magnitude and polar coordinates of each pixel are calculated:

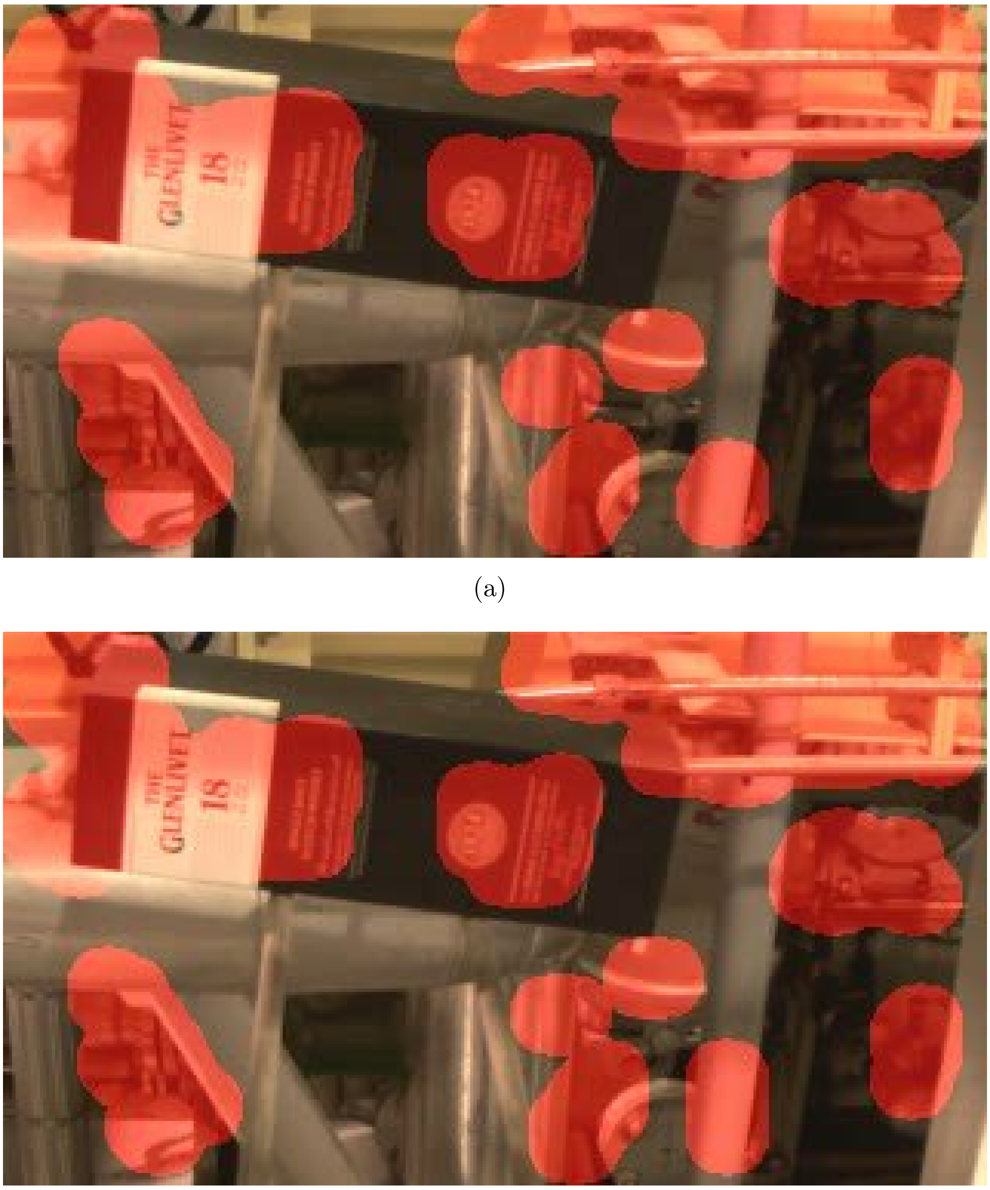
$$\mathrm{M} = \sqrt{\left(\frac{dI}{dx}\right)^2 + \left(\frac{dI}{dy}\right)^2 + \left(\frac{dI}{dt}\right)^2}, \tag{4.2}$$

$$\phi = \tan^{-1}\left(\frac{\mathrm{G_t}}{\sqrt{\mathrm{G_x^2} + \mathrm{G_y^2}}}\right), \tag{4.3}$$

$$\theta = \tan^{-1}\left(\frac{\mathrm{G_t}}{\mathrm{G_x}}\right). \tag{4.4}$$

The variables ϕ with range $(\frac{-\pi}{2}, \frac{\pi}{2})$ and θ with range $(-\pi,\pi)$ are quantized in 8 and 16 bins, respectively, of a histogram. So, the final descriptor is a histogram of 24 bins. Each pixel of v_i add two values in the histogram, one for ϕ and other for θ. These values are weighted by the magnitude M.

However, in our implementation the results using this alternative descriptor were bad, with a large number of false detections. But the main problem was that the detection was very unstable. In some situations, no abandoned object was detected, and in other there were a large number of false detections. So, it is impossible to determine the position of the abandoned object, as depicted for instance in Fig. 4.7.



(a)

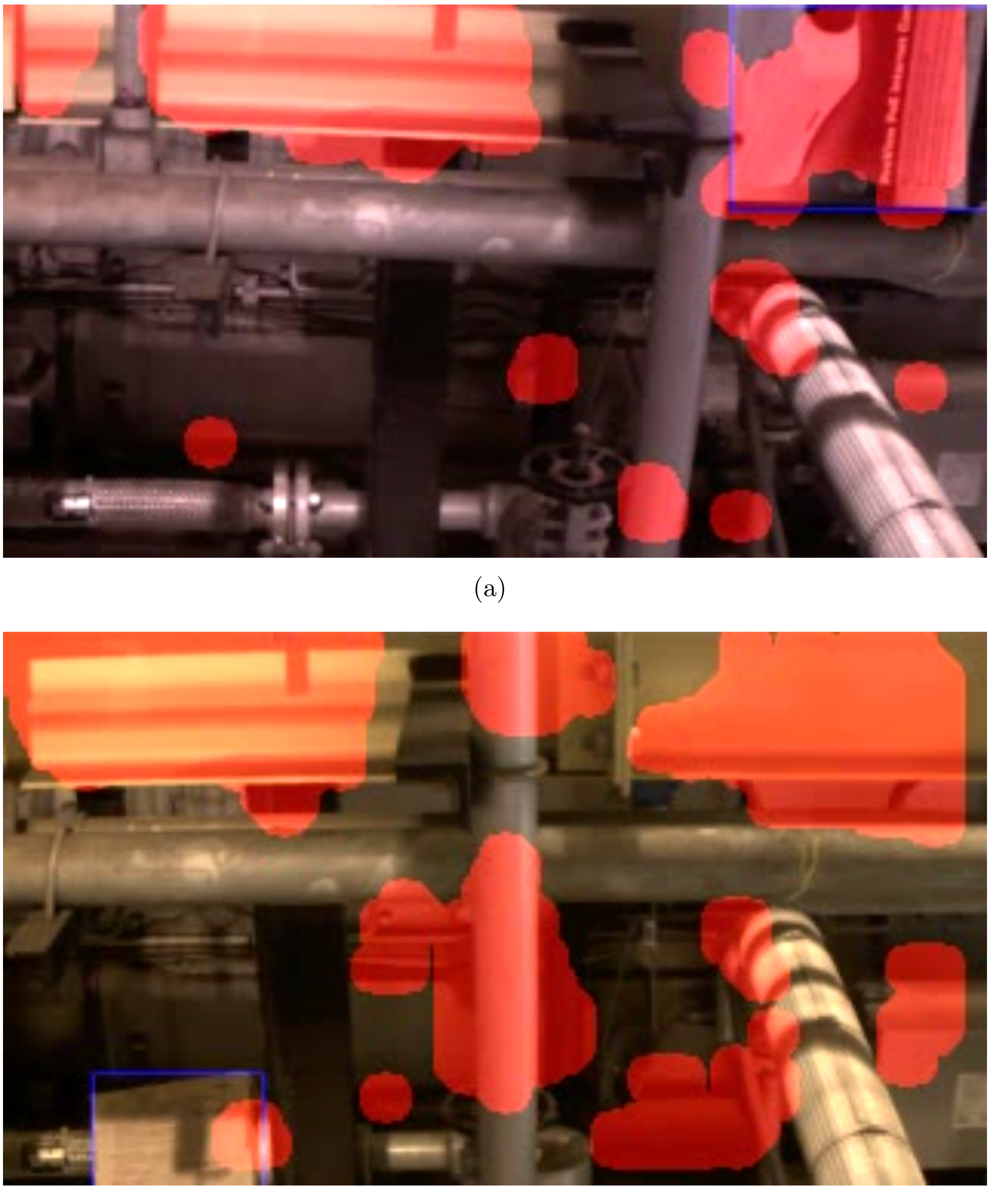
(b)

Figure 4.6: Using the descriptor of the Eq. (4.1), the results are more stable, with little variation from a frame to the next.

4.4.2 The Chosen Descriptor

After the initial test, the descriptor of the Eq. (4.1) was chosen because it was able to detect all the abandoned objects. Although this descriptor generates more false detections, it was more stable, which allows a better determination of the effects of further improvements. Better results were obtained when spatio-temporal gradients of the Eq. (4.5) were used to compute the descriptor. The results obtained using this kind of descriptor are shown in detail in the Chapter 7.

$$\mathrm{D} = \sqrt{\left(\frac{dI}{dx}\right)^2 + \left(\frac{dI}{dy}\right)^2 + \left(\frac{\lambda dI}{dt}\right)^2}. \tag{4.5}$$



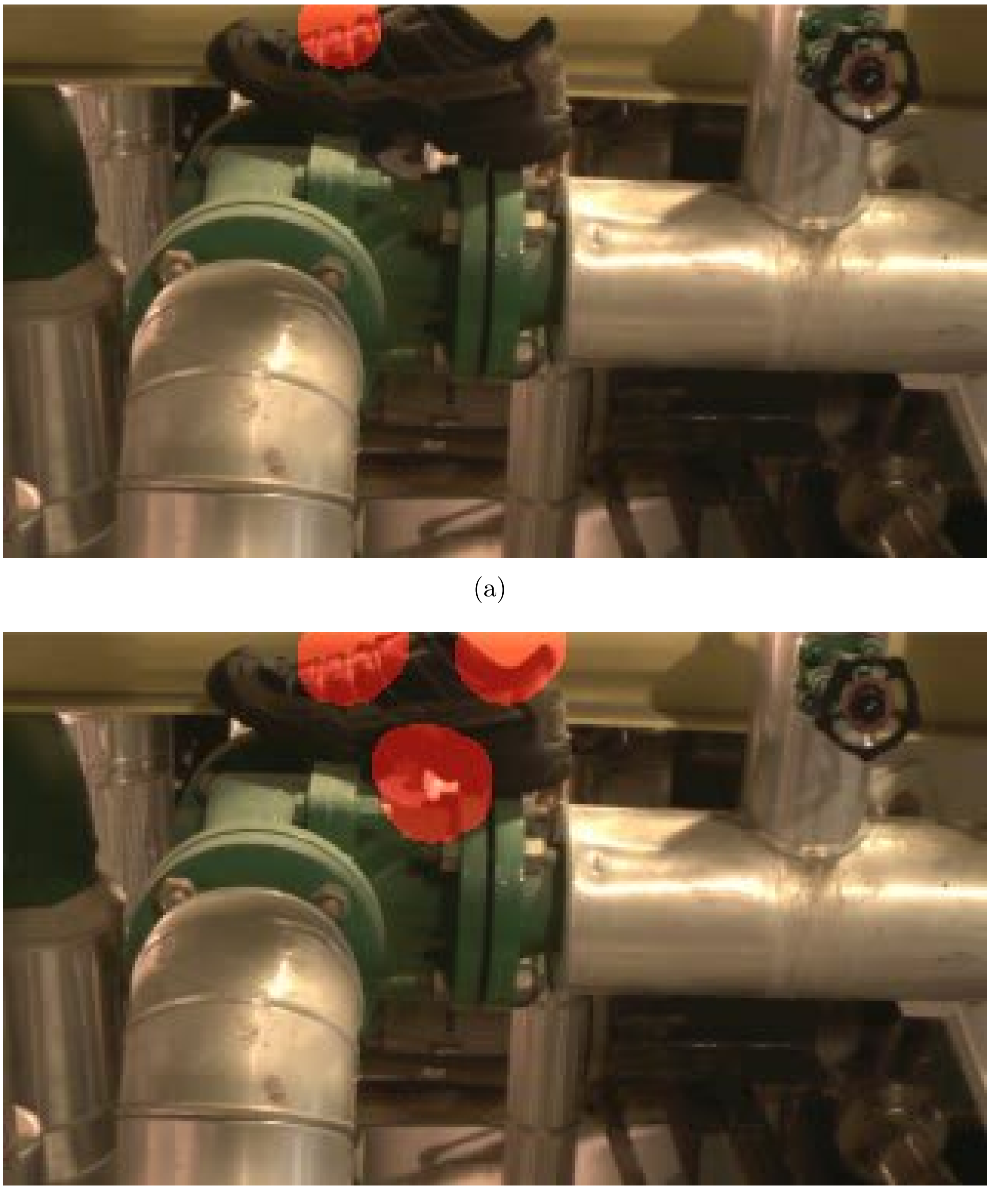
(a)

(b)

Figure 4.7: Test with the HOG descriptor in a video where the abandoned object is a) a camera box, b) a white box. The results have a large number of false detections.

To construct such a descriptor, D in Eq. (4.5) is computed for each pixel inside the volume v_i and the results are stacked in a vector. Note that this descriptor extracts useful information even in the more challenging case of environments where the background is cluttered and not static.

It is important to observe that the difference to the Eq. (4.1) is that the temporal derivative is multiplied by a constant λ, whose value is to be determined as described in Chapter 6. This constant performs an adjustment to account for relative effects of the spatial resolution, frame rate and Gaussian smoothing on the computation of the descriptor. In the Fig. 4.8a the Eq. (4.1) was used to calculate the descriptor and in the Fig. 4.8b it was used the Eq. (4.5), with $\lambda = 2$. The latter is able to detect better the abandoned object, without false detections. Similar results were obtained in Figs. 4.9a and 4.9b, when the abandoned object is a whisky box. These results were obtained after the use of a Gaussian filtering, as described in Section 4.5.

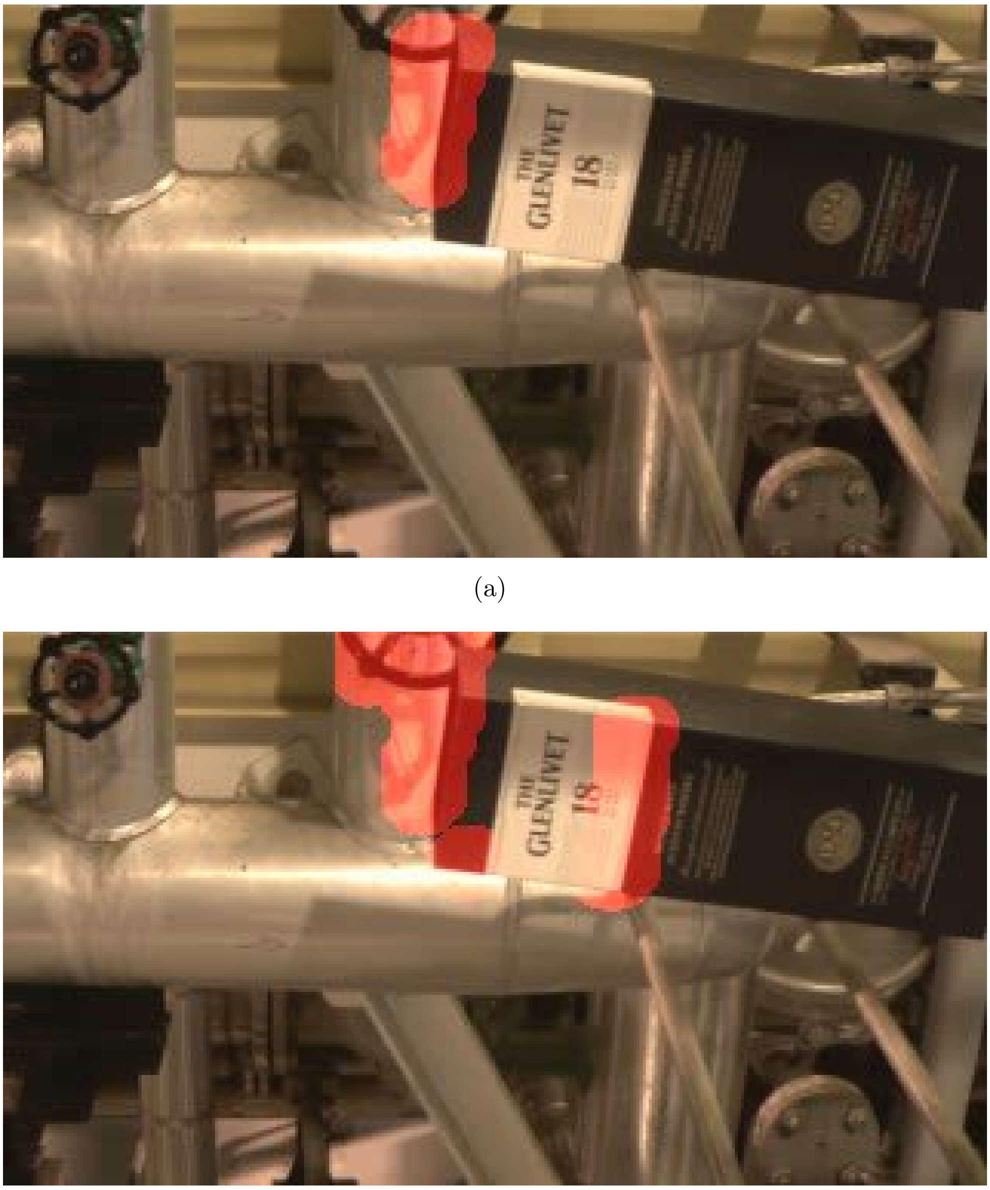


(a)

(b)

Figure 4.8: Results obtained when the abandoned is a sneaker: a) using the Eq. (4.1), b) using the Eq. (4.5). In b), more points in the abandoned object are detected, without false positives. The results were obtained after a Gaussian filtering.

The volume v_i is given by $n_x \times n_y \times n_t$ and the values used were $7 \times 7 \times 5$ pixels. Thus, the descriptor had a dimension of 1×245. The volume size in pixels was empirically defined, according to [7].



(a)

(b)

Figure 4.9: Results obtained when the abandoned object is a whisky box: a) using the Eq. (4.1), b) using the Eq. (4.5). In b), more points in the abandoned object are detected, without false positives. The results were obtained after a Gaussian filtering.

4.5 Gaussian Filtering

As seen in Eq. (3.1), the STC descriptor of a spatio-temporal volume is given by the time derivative of each pixel in the volume. In the STC-mc, the descriptor is given by Eq. (4.5). When the camera is on a moving platform, the camera tends to shake along its path, especially if the rail has small irregularities. This may cause a large random variation on the values of this derivative from frame to frame. Fig. 4.10, which shows the temporal derivatives of four consecutive frames from a sequence from the VDAO database, confirms this. With such a high variation from frame to frame, clearly a descriptor that uses only a temporal derivative is unsuitable for the task at hand. Although the introduction of the spatial derivative in the descriptor of the STC-mc helped to minimize the effect of the shaking, a more specific solution had to be developed.

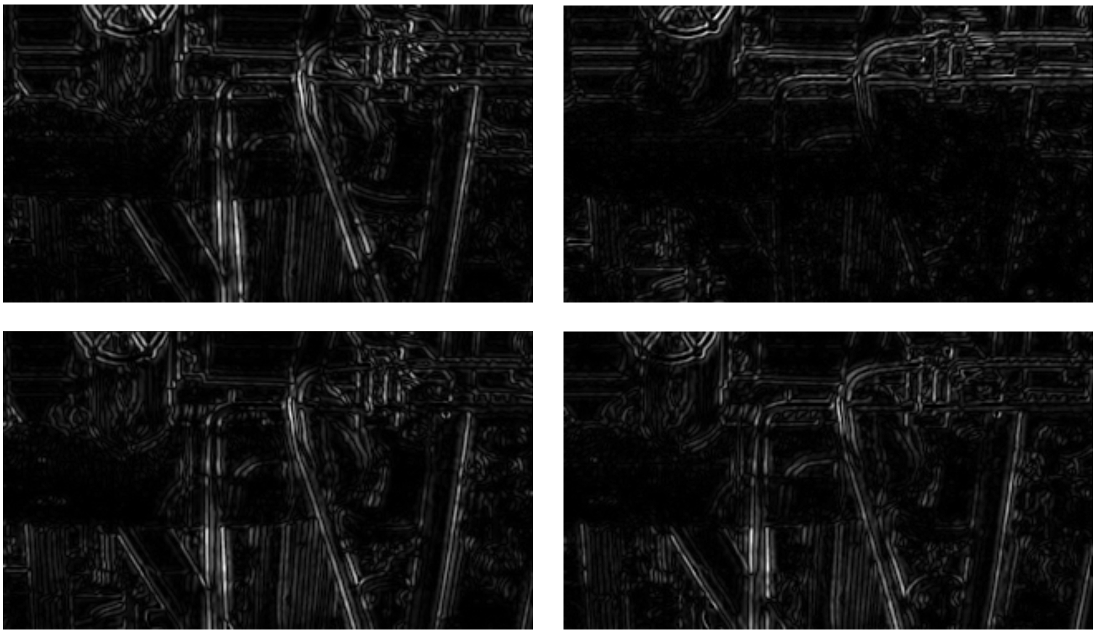


Figure 4.10: Temporal derivatives of four consecutive frames from the VDAO database. One can see the large variation of this descriptor.

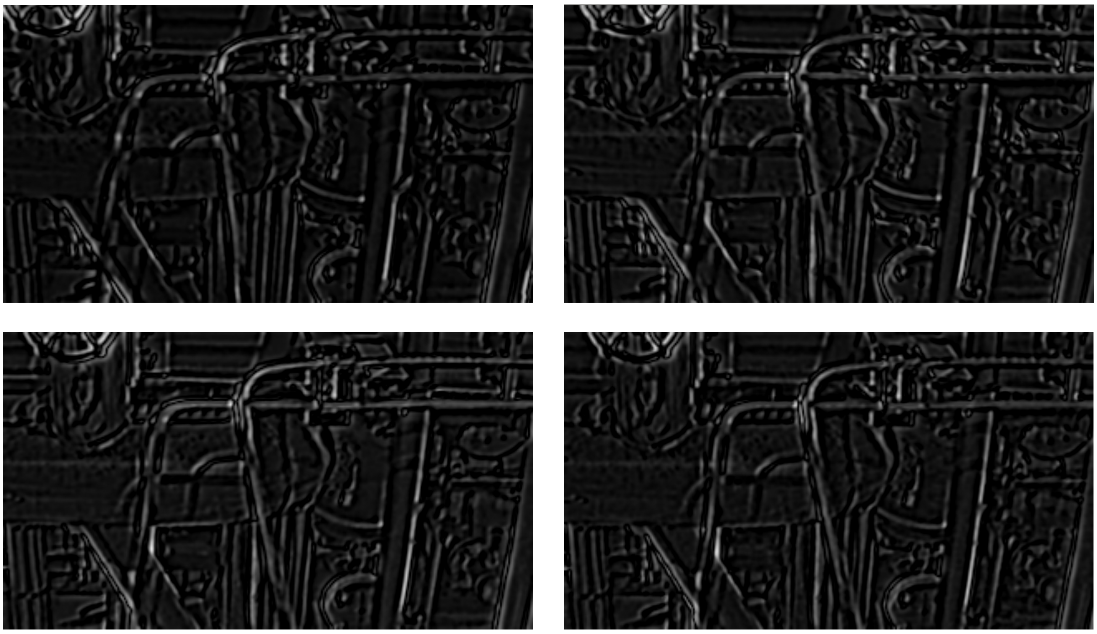


Figure 4.11: Temporal derivative after a Gaussian filtering. The result is more stable.

To attenuate such variation in the temporal derivative, we propose to perform a temporal smoothing prior to the derivative computation. This can be done employing a Gaussian filter. The size of the filter kernel was set to five, and the value of the standard deviation σ was tuned as described in Chapter 6. Hence, the impulse response of the Gaussian filter is given by

$$h(n) = \begin{cases} ke^{\frac{-n^2}{2\sigma^2}}, & -2 \leq n \leq 2, \\ 0, & |n| > 0 \end{cases} \tag{4.6}$$

where k is such that $\sum_{n=-2}^{2} h(n) = 1$.

Fig. 4.11 shows the time derivatives of the frames of Fig. 4.10 after the temporal Gaussian filtering. Clearly this derivative is much more stable, and thus suitable for being incorporated in the proposed spatio-temporal descriptor.

To test if the Gaussian filter really improves the performance, the percentage of true positives (TP) and false positives (FP) are used to plot a point of a (FP× TP) curve [54]. Each test of the detection of an abandoned object corresponds to a point on the curve. In our analysis we consider the best result the one closest to the point (0,1), which is the point with 0% of false positives and 100% of true positives.

The Table 4.1 shows the result of the comparison of the use of the Gaussian filter when $\lambda = 1$ in the Eq. (4.5). The results with and without the Gaussian filter are almost the same. The tests with the Gaussian filter are better in two of five tests, and worse in two of five tests too. However, when $\lambda = 2$, the tests using the Gaussian filter give better results in all the cases, as shown in Table 4.2. Therefore, the best configuration is when the Gaussian filter is used and the descriptor has $\lambda = 2$ in Eq. (4.5). This result is confirmed in Chapter 6.

Table 4.1: Comparison of the results obtained when the weight $\lambda = 1$. TP is the true positive rate, FP is the false positive rate and DIS is the distance to the point (0,1) in the FP×TP plane, which is the best possible point. The results obtained with the Gaussian filter are better in two of five tests and worse in two of five tests.

Object	With Filter			Without Filter		
	TP	FP	DIS	TP	FP	DIS
Sneaker	0.97	0.01	0.03	0.98	0.02	0.03
Whisky box 1	0.83	0.21	0.27	1.00	0.33	0.33
Whisky box 2	0.39	0.13	0.62	0.58	0.31	0.52
Camera Box	1.00	0.01	0.01	1.00	1.00	1.00
Towel	0.52	0.00	0.48	0.59	0.00	0.41

Table 4.2: Comparison of the results obtained when the weight $\lambda = 2$. TP is the true positive rate, FP is the false positive rate and DIS is the distance to the point (0,1) in the TP×FP plane, which is the best possible point. The results obtained with the Gaussian filter are better in all the tests.

Object	With Filter			Without Filter		
	TP	FP	DIS	TP	FP	DIS
Sneaker	0.97	0.01	0.03	1.00	0.21	0.21
Whisky box 1	1.00	0.22	0.22	1.00	0.33	0.33
Whisky box 2	0.56	0.13	0.46	0.86	0.72	0.73
Camera Box	1.00	0.01	0.01	1.00	1.00	1.00
Towel	0.93	0.00	0.07	1.00	0.18	0.18

4.6 Anomaly Detection

As in the training stage, the video to be analyzed passes through the same steps of descriptor creation: allocation of a codeword of the first codebook to each spatio-temporal volume and calculation of the probabilities of these volumes using the pdf of the spatio-temporal arrangement. Then, the second codebook is used to determine the points that should not be considered as anomalous, as they are already part of the background in the reference video. Points with a probability below a given threshold γ and not present in the second codebook are considered anomalous. The algorithm is shown in Fig. 4.12.

```
The Euclidean distance is used to determine the similarity between the descriptor and the codewords in the codebook.
The probability of each volume is calculated in the first training pass.
$\gamma$ is the probability threshold and $\epsilon_2$ is the maximum distance between descriptors that are considered as similar.
$\mu$ and $\nu$ are the distance and probability tolerance, respectively.
for All volumes $\{v_i\}_{i=1}^{N}$ do
  if $p_i < \gamma$ then
    if $\min_j distance(v_i, c_j) > \mu\varepsilon_2$ then
      if $p_i > \nu p_{c_j}$ then
        This is an anomalous point. Mark it.
      end if
    end if
  end if
end for
```

Figure 4.12: Anomaly Detection algorithm.

Fig. 4.13 shows the result using only the first codebook. There are a lot of false detections together with the abandoned object, in this case a whisky box. Fig. 4.14

shows the results when the second codebook is used too. The false detections are eliminated and only the true detections are marked. The total area marked in the abandoned object is smaller, but the objective is to detect the presence of anomalies, with the fewest possible number of false detections. The Figs. 4.15 to 4.20 show other examples of the difference when the second codebook is also used.

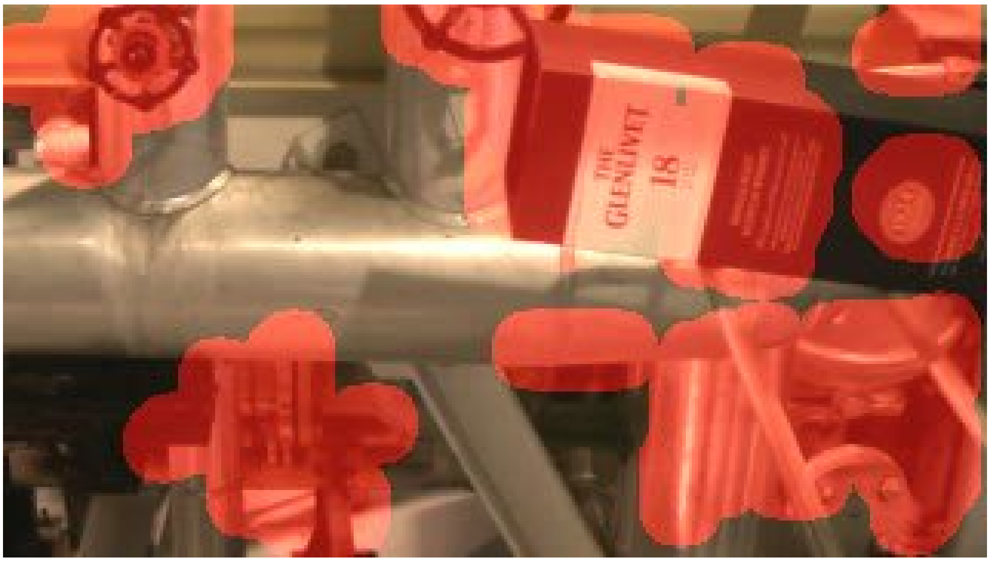


Figure 4.13: Result with only the first codebook when the abandoned object is a whisky box. There are several false detections.

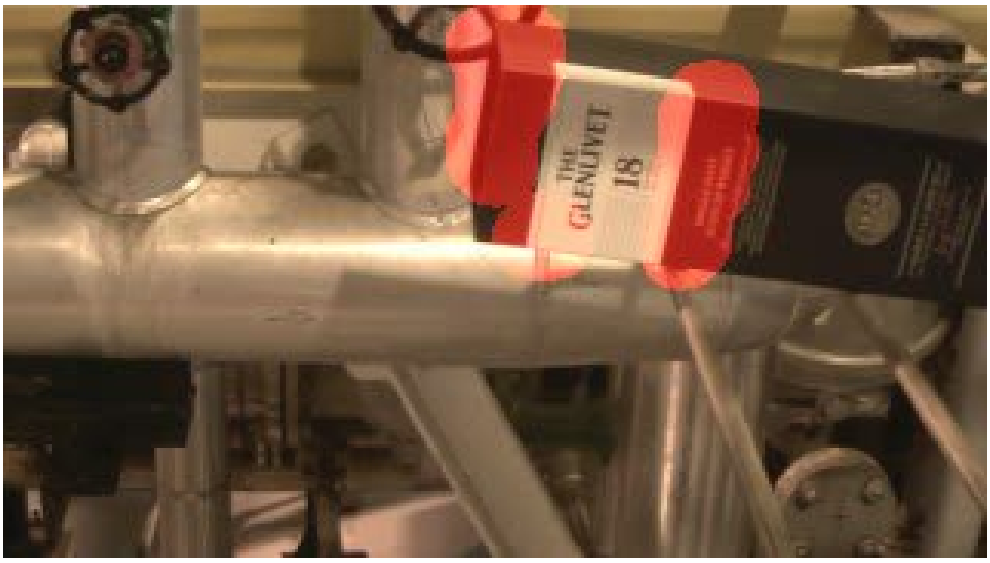


Figure 4.14: Result with the introduction of the second codebook when the abandoned object is a whisky box. The detected area is smaller, but the false detections disappear.

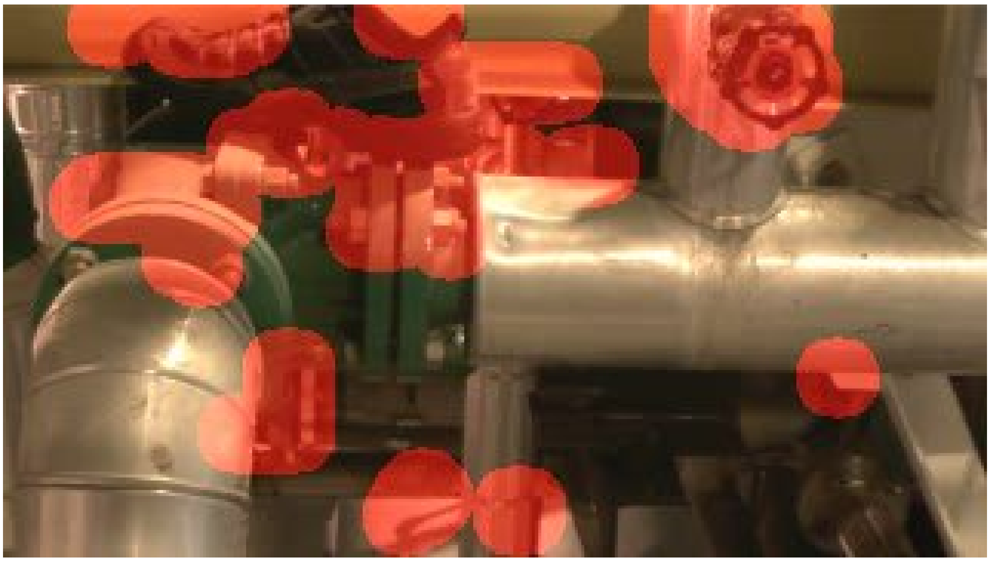


Figure 4.15: Result with only the first codebook when the abandoned object is a sneaker. There are several false detections.

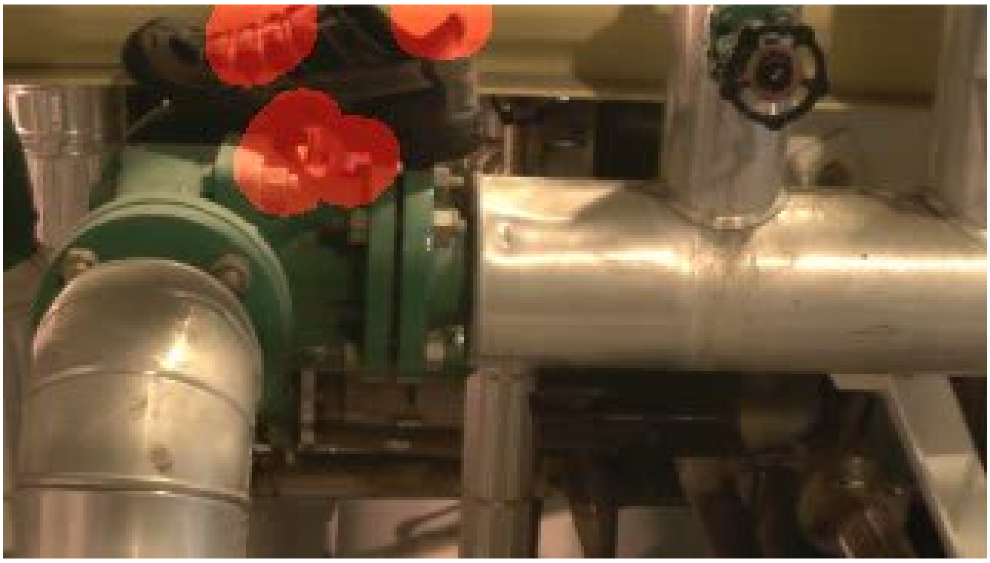


Figure 4.16: Result with the introduction of the second codebook when the abandoned object is a sneaker. The detected area is smaller, but the false detections disappear.

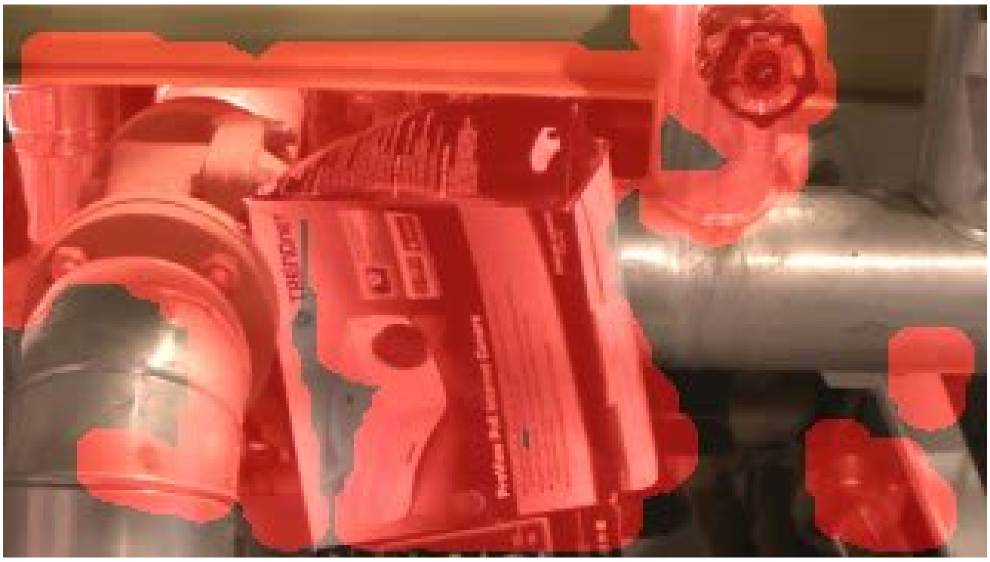


Figure 4.17: Result with only the first codebook when the abandoned object is a camera box. There are several false detections.

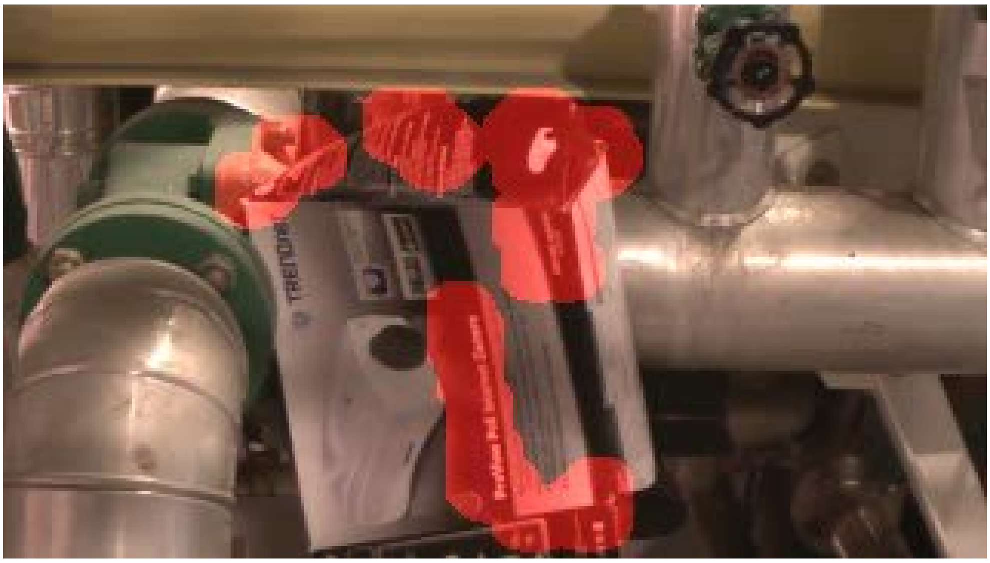


Figure 4.18: Result with the introduction of the second codebook when the abandoned object is a camera box. The detected area is smaller, but the false detections disappear.

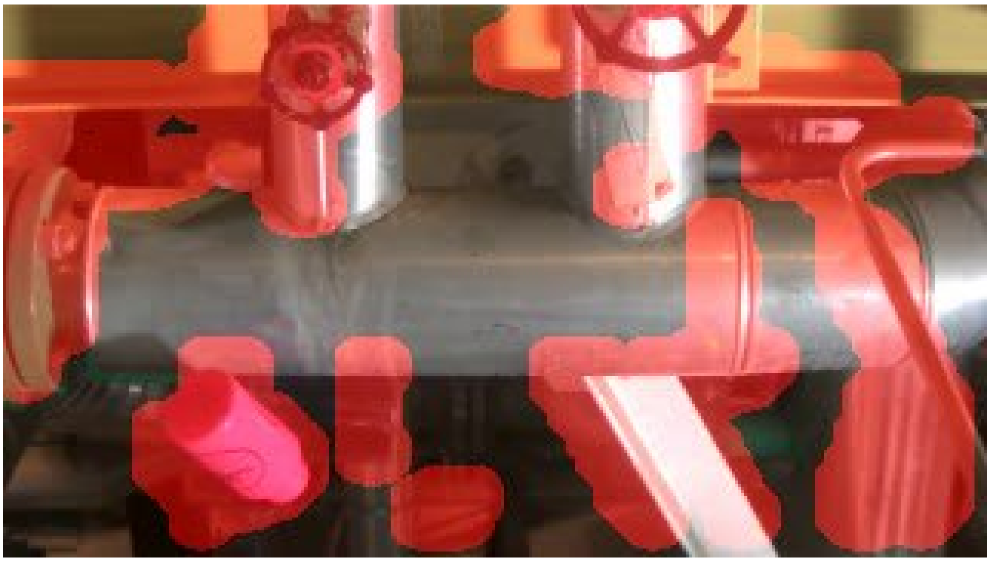


Figure 4.19: Result with only the first codebook when the abandoned object is a bottle. There are several false detections.

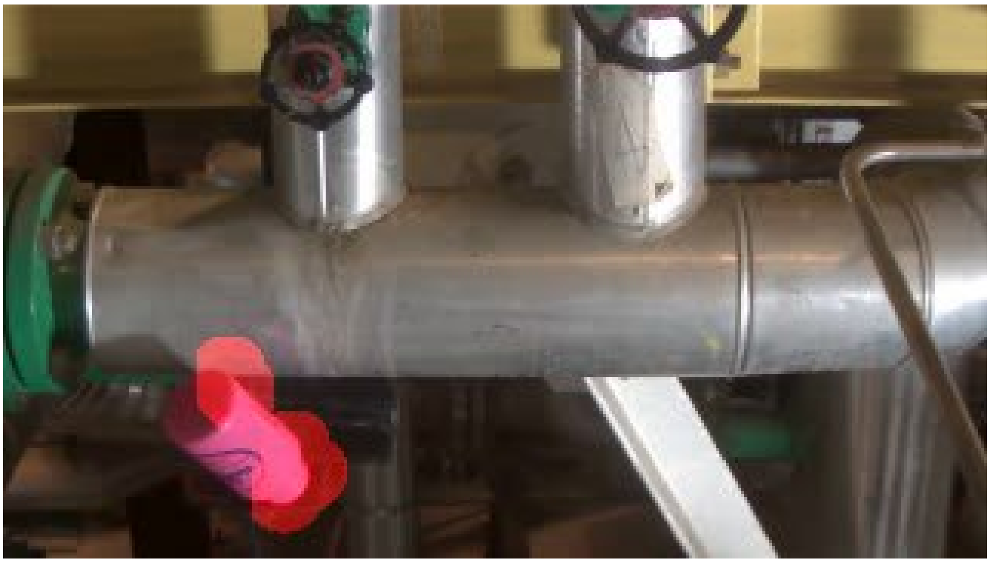


Figure 4.20: Result with the introduction of the second codebook when the abandoned object is a bottle. The detected area is smaller, but the false detections disappear.

After the detection of the anomalous points, a voting procedure is performed to improve the final result. The reason to execute this stage is that an anomaly can be detected in a certain position of a frame and in the next frame may not be present. This is likely to be a false detection, because in most of the cases the objects do not move so fast that they could disappear from one frame to appear again in another. These false detections could be caused by illumination changes or the noisy movement of the robot. To perform the voting, the frames are analyzed in groups of nine. Each anomalous object is marked, being identified by its area and centroid position. In order to an object be considered anomalous, it needs to appear in seven out of nine consecutive frames. As the object may be moving, a variation of 10 pixels in the centroid position is allowed in any direction. Fig. 4.21 shows the result without the voting and Fig. 4.21 shows the result after the voting. One can see that the false positives tend to disappear due to this procedure.

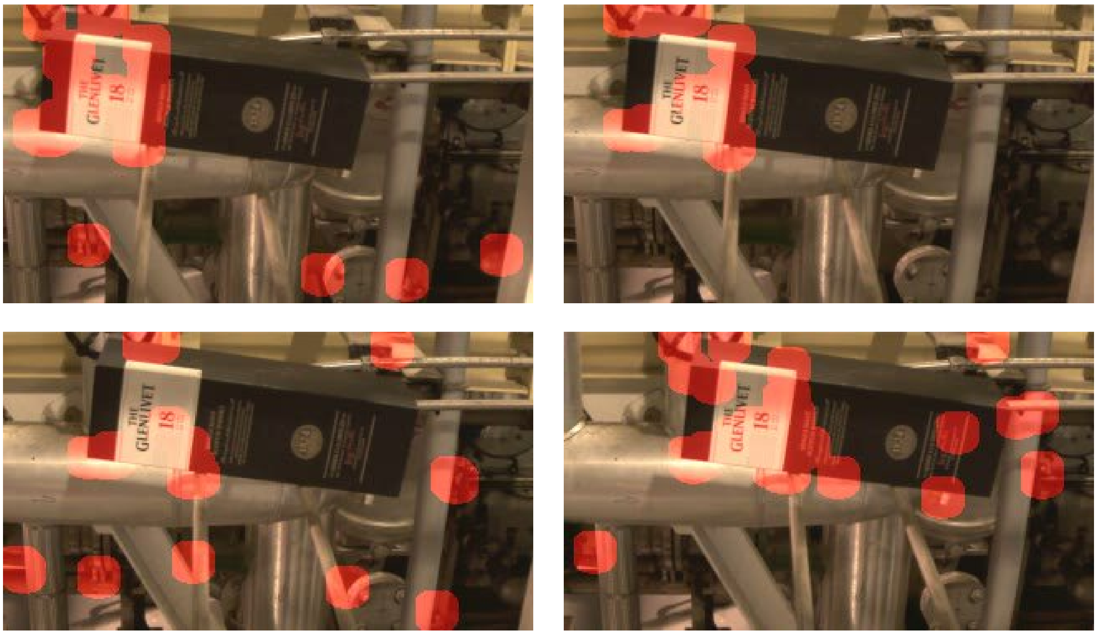


Figure 4.21: Without a voting, several blobs appear and disappear from frame to frame.

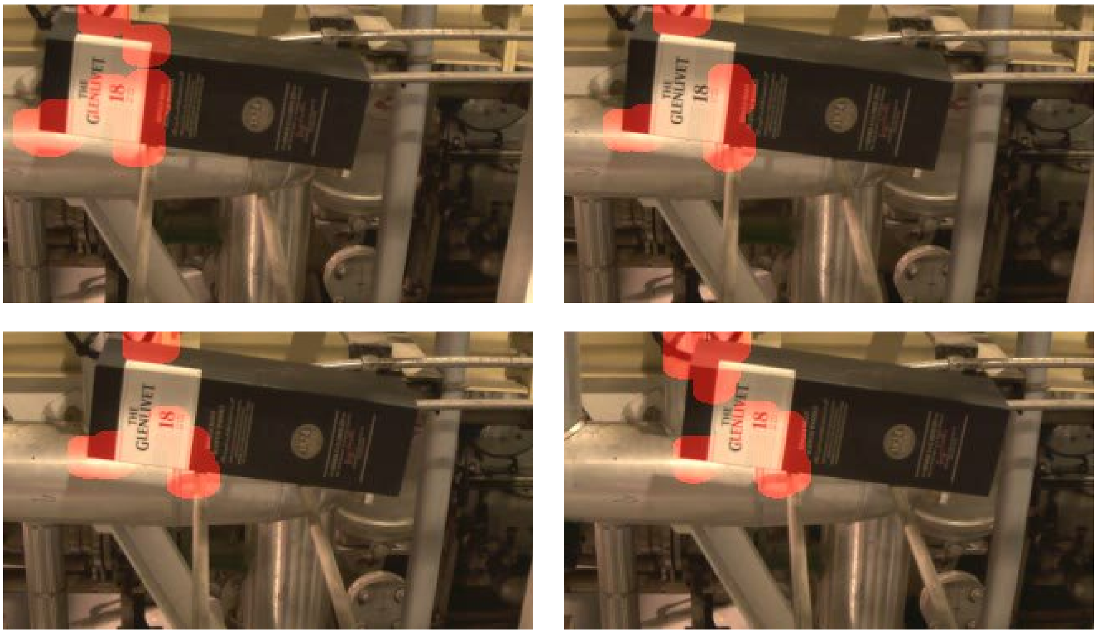
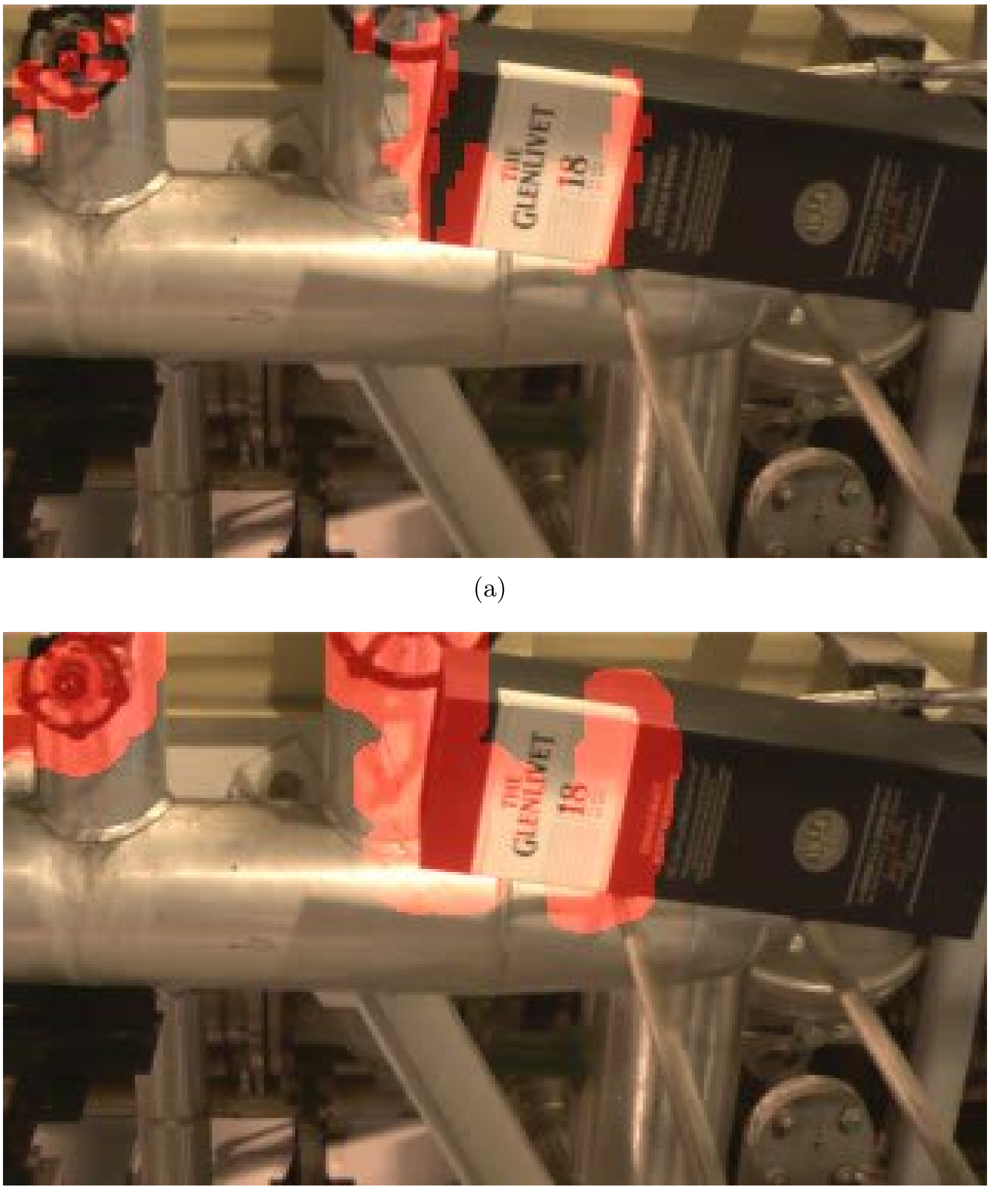


Figure 4.22: With the voting, the detected area of the abandoned object decreases, but the false detections disappear.

In some situations, an object can be detected as many separated regions, because the probability of occurrence of the points inside an object may vary and the anomalies usually is detected in the borders of the objects. To minimize this effect, a morphological binary closing operation [55] is performed to connect the detected regions. In this procedure, a round structuring element with radius equal to 20 pixels is used. Although only one pixel marked as anomalous is sufficient to detect an anomaly, this closing is intended to highlight the anomaly to the operator, and this does not interfere in the final result. The Fig. 4.23 shows the result when the closing operation is performed in an image.



(a)

(b)

Figure 4.23: The use of a morphological binary closing operation. The visual identification is improved.

In brief, in this chapter one can observe that the development of STC-mc method involved the search for different solutions for each new problem. Starting from STC method, improvements were made to reduce the detection of false positives, false negative and also some random variations in the detected points. The sum of these improvements resulted in the STC-mc method. This method is able to detect abandoned objects correctly, with a low number of false detections, as shown in the examples. The Chapter 6 presents the methodology used to adjust the parameters of the STC-mc method to obtain the best results.

Chapter 5

The VDAO Database

This chapter presents the database used to perform the simulations in this thesis. It also describes the software developed to mark the abandoned objects in the video and to create a reference video to assess the performance of the STC-mc method.

The target application of this work is to detect abandoned objects using a camera that is mounted on a moving robot. The robot surveys an industrial environment by performing rectilinear, back and forth movements along a fixed rail. The camera is arranged so that the image depicts a lateral view. Fig. 5.1 shows an example of the type of scene used.

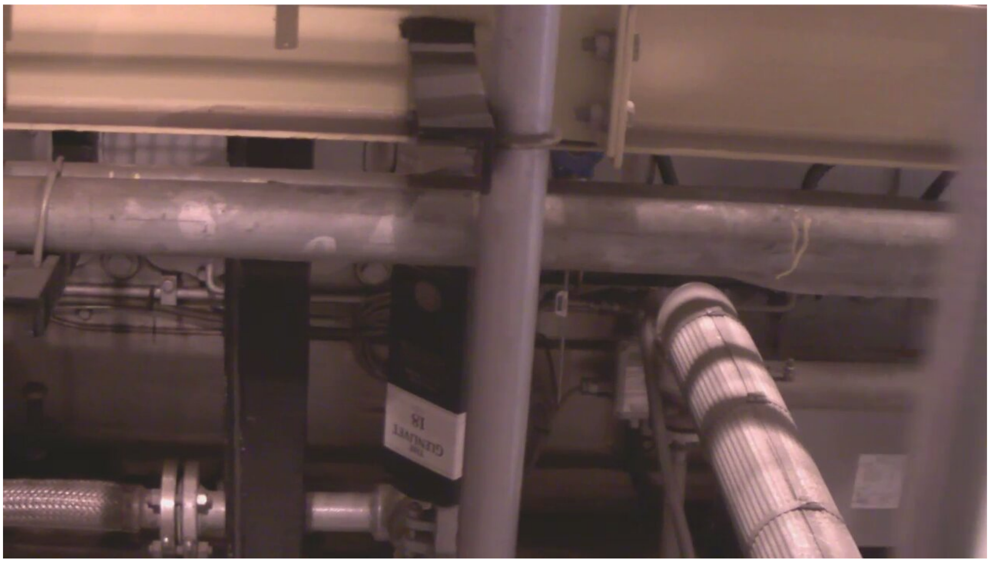


Figure 5.1: An example of the VDAO database. The video was acquired during an inspection of an industrial facility using a camera mounted on a robot that moves along a rail.

The system consists of a camera mounted in a robotic platform called Roomba$^{®}$ [56]. Fig. 5.2 shows the robot on the rail, in an industrial plant. The rail was hanged in a height of approximately 2.5 m. The environment was very cluttered, with several pipes, valves and metallic structures, being similar to an off-shore platform or an oil refinery, as shown in Fig. 5.3.

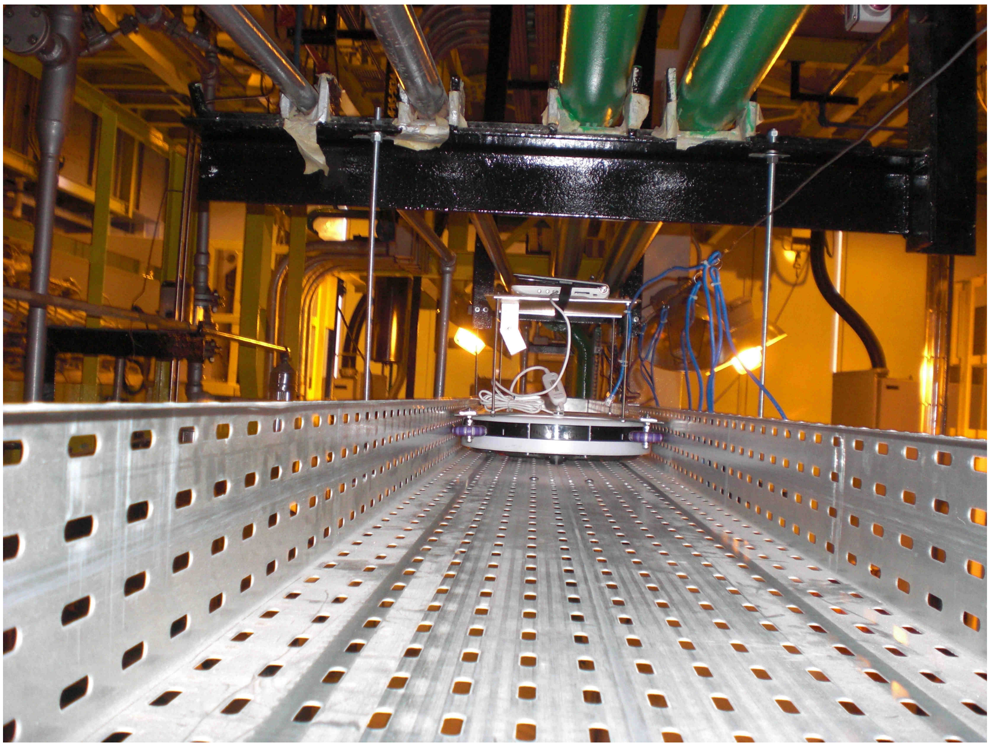


Figure 5.2: The robot used to record the videos of the VDAO database.

Using this robotic system, several videos were recorded to create a database called VDAO (database of abandoned objects in a cluttered environment) described in [12] and available in [46]. This database, besides containing reference videos without abandoned objects, also has videos containing different objects with different colors, shapes and textures (e.g., a sneaker, a towel, a box, etc). Fig. 5.4 shows examples of these objects. The position of objects inside the video frame and the time when it is displayed varies from video to video. Moreover, there are variations in brightness between the videos caused by the difference of natural lighting or the use of an artificial lighting. Another important characteristic of the VDAO database is that the positions of the abandoned objects in all frames of the videos are also provided, in the form of the coordinates of the bounding boxes containing the objects. Figs. 5.5 to 5.8 show some frames of the videos from the VDAO database. It is important to highlight that the situations present in this database are common in practical applications of surveillance robots, tending to be quite challenging for video surveillance algorithms.

The VDAO database was developed as follows:

- Two different cameras were used, both with the resolution of 1280× 720 pixels and rate of 24 frames per second;
- In half of the videos, a spotlight was used, making a brighter recording, and

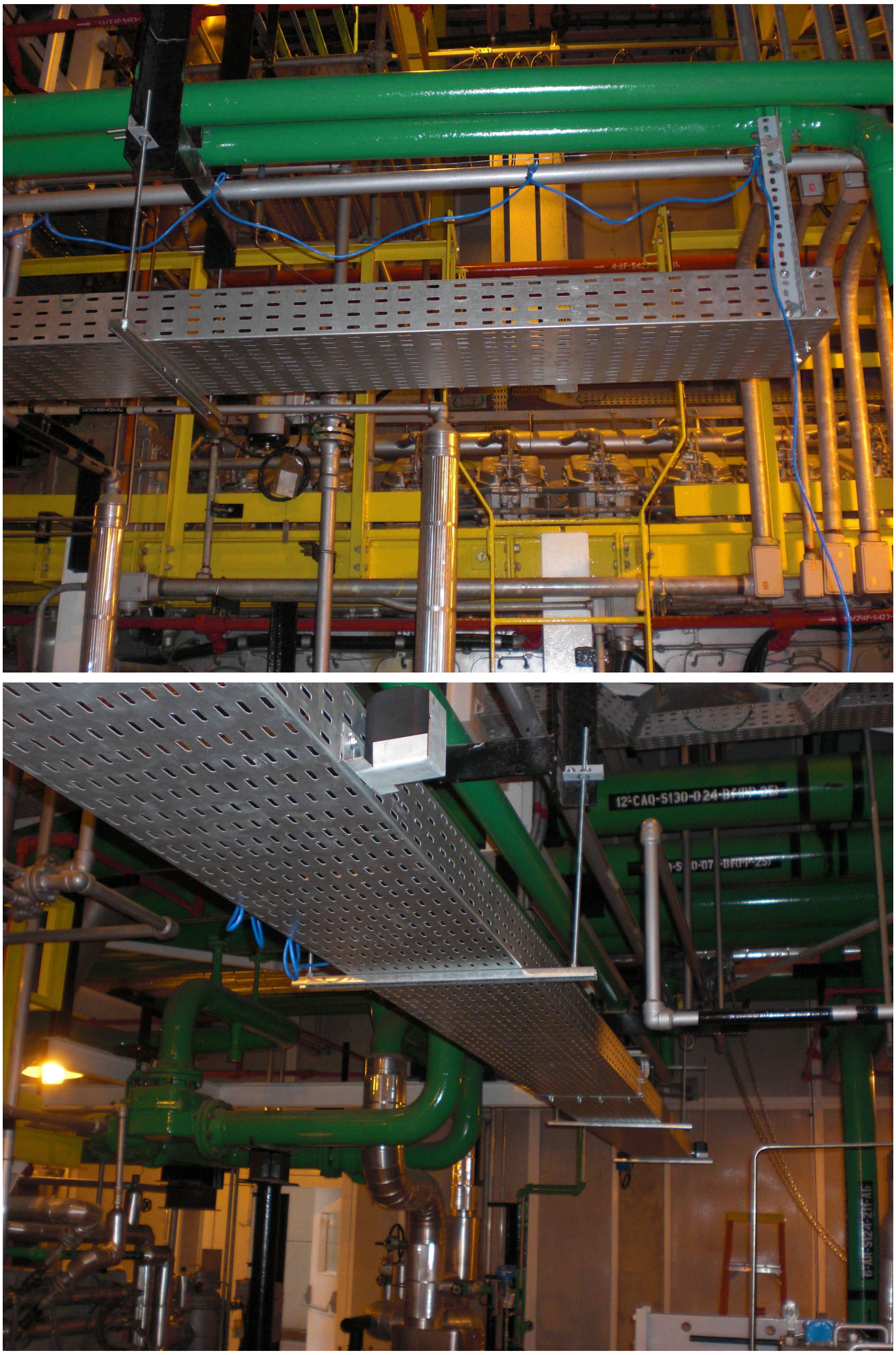
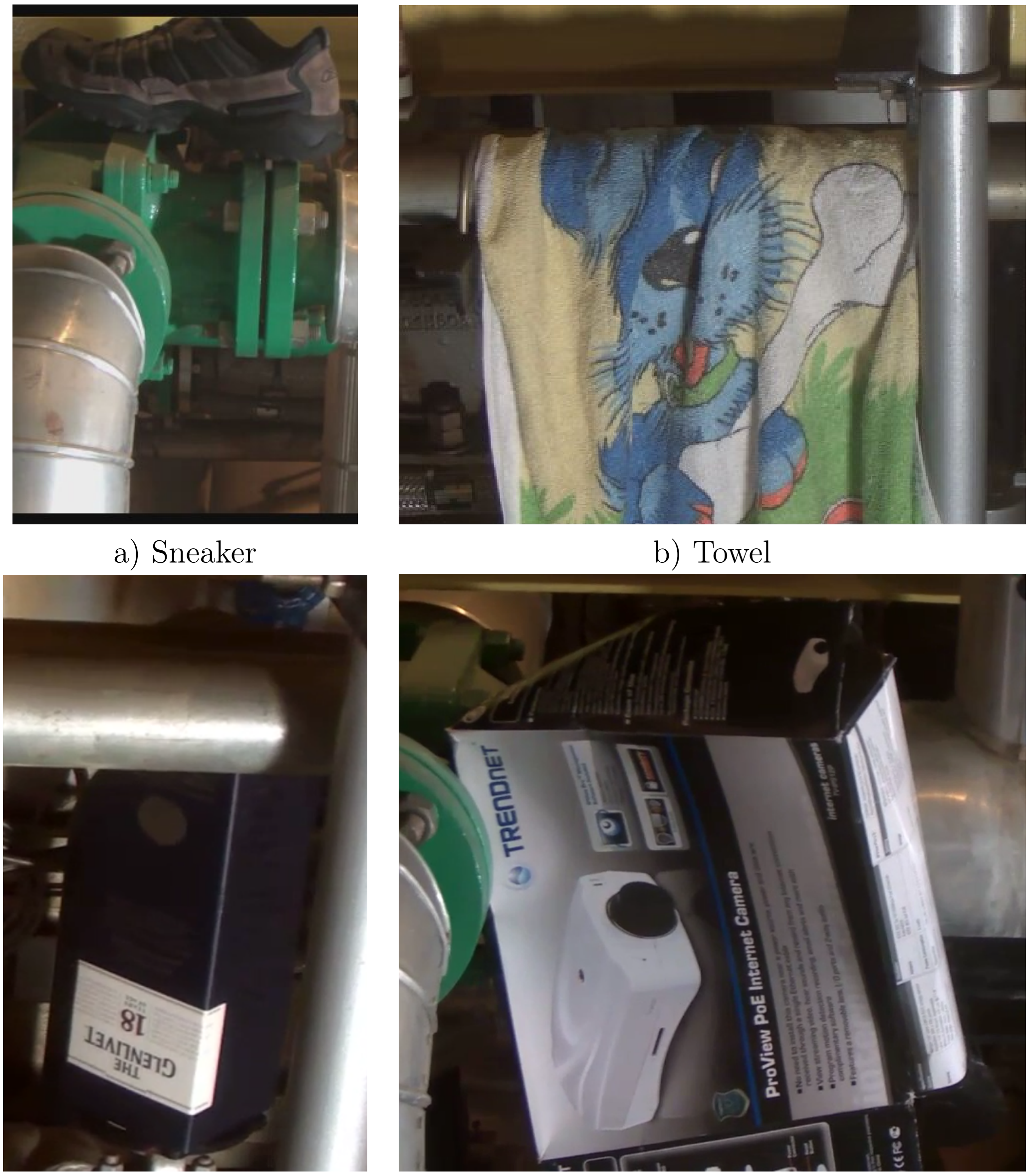


Figure 5.3: The industrial plant where the videos of the VDAO database were recorded.

in the other half a natural light was utilized;

- With one camera (Axis P1346), six multiple-objects and two no-object (reference) videos were recorded, and with other camera (Dlink DCS-3717), 54 single-object and two no-object (reference) videos were recorded;



a) Sneaker

b) Towel

c) Drink box

d) Camera Box

Figure 5.4: Examples of abandoned objects present in the VDAO database.

- The single objects videos only have one passage in each direction, and the multi-object videos include six full passages of the camera;
- The six multiple-object videos were recorded with the same 15 objects placed in three different positions, with the spotlight or natural light, making completely distinct arrangements. The different positions also caused some objects to change size in the footages, as they may be farther or closer to the camera. The 54 single-object videos work in a similar way, as each of the nine objects was recorded in three different positions and with/without the use of a spotlight;
- All multiple-object videos were devised in a way that most of the frames include at least two objects.

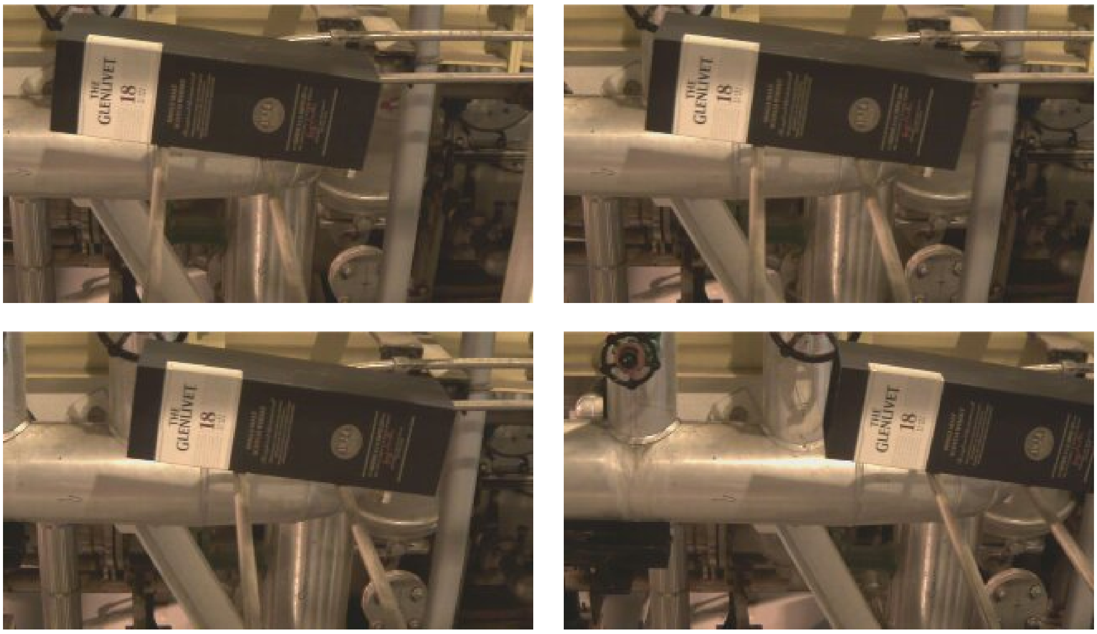


Figure 5.5: Frames of a video from the VDAO database, where the abandoned object is a whisky box.

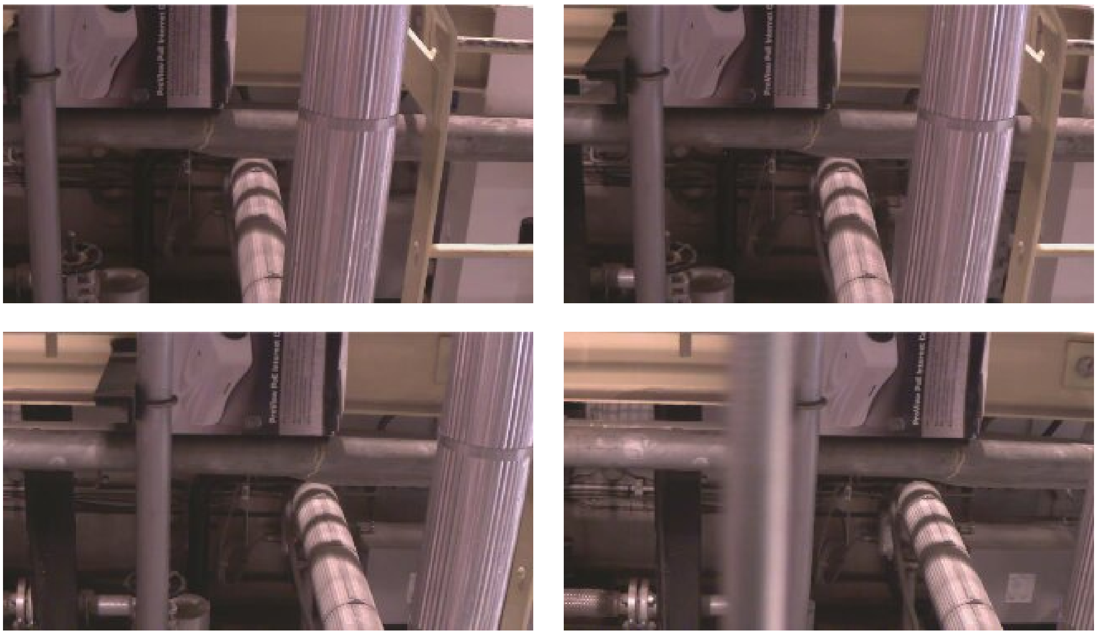


Figure 5.6: Frames of a video from the VDAO database, where the abandoned object is a camera box.

5.1 Video Marking Program

In all studies involving the detection of objects, it is important the availability of a database where the objects of interest are with their positions and dimensions mapped, so as to have a standard of comparison. Currently, it is considered that manual annotation, made frame by frame by an observer is still the most accurate

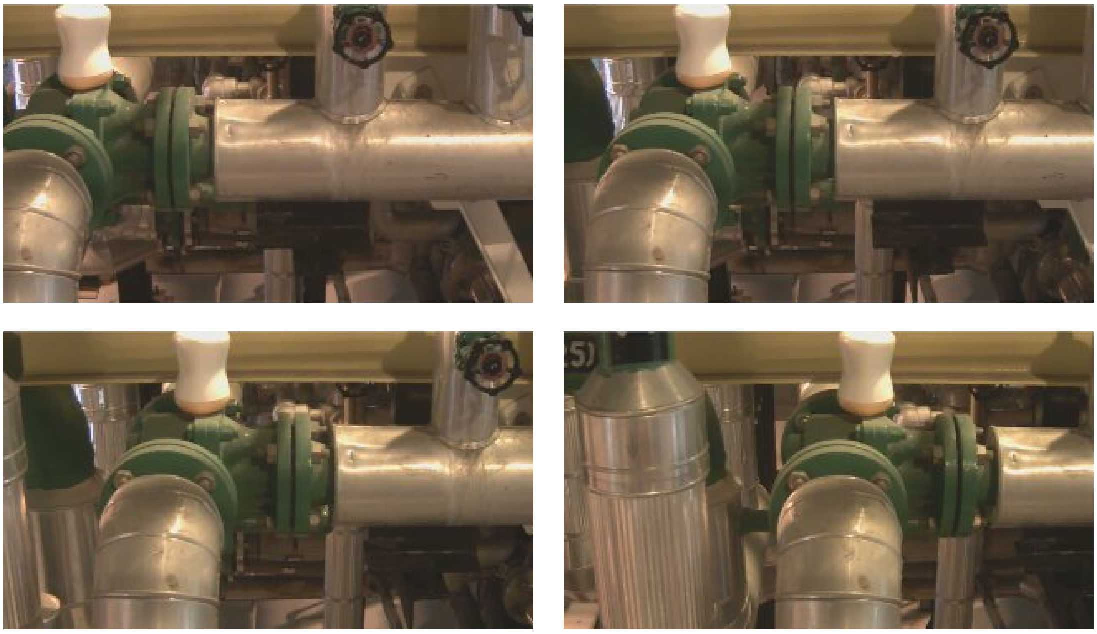


Figure 5.7: Frames of a video from the VDAO database, where the abandoned object is a bowl.

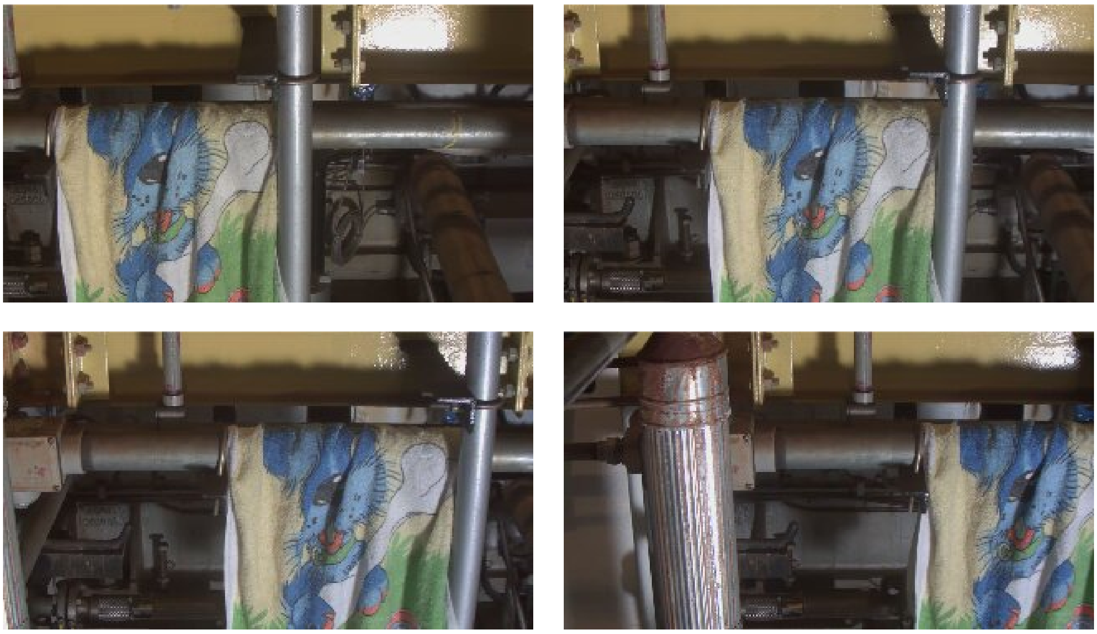


Figure 5.8: Frames of a video from the VDAO database, where the abandoned object is a towel.

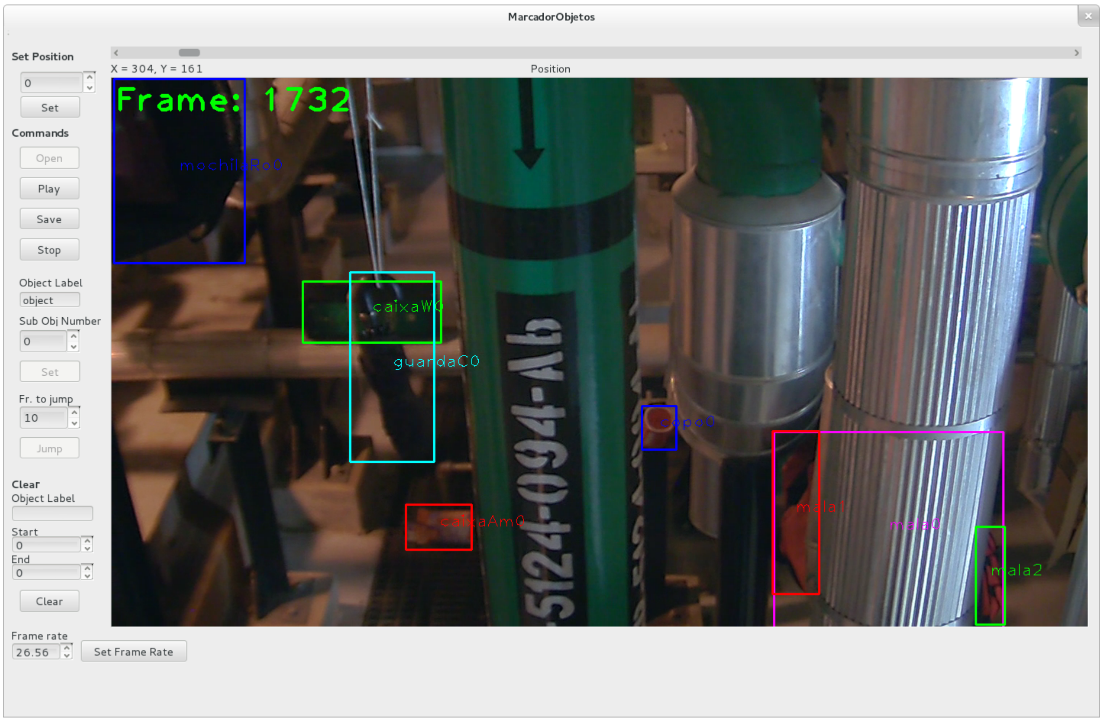
way of marking. However, this marking is a lengthy and tedious process. Only one minute of video has 1800 frames, using 30 frames per second. Thus, at the start of the work, it became clear that regardless of the algorithm to be used, a database well documented would be necessary. But instead use one of the many databases available on the Internet, it was decided to develop a software that would help in the marking task of any video of interest. That is because while there are many marked video databases on the Internet and available for download, it's hard to find one that meets exactly the unique requirements of each job or project. The available databases vary greatly in quality, quantity of tagged objects, viewing angle and duration. Besides that, standards softwares are not practical for video markup, taking too long time to mark each frame. Apart from that, with the development of an application for this purpose, it was possible familiarize with the high-performance software development using C ++. The new software was developed using the QT Creator 2.81 application, due to its friendly interface, extensive documentation available and portability between Windows and Linux (Fedora 19 and Ubuntu 14.04) operating systems. The program also used the free and multiplatform OpenCV library for video manipulations.

The basic requirements in the development were:

- Object mark consisting of an outline of easy identification, preferably a simple bounding box;
- Marks inserted quickly, via mouse, due to the large number of frames to be considered;
- Input commands via a GUI interface with a minimum number of intuitive commands;
- Ability to mark multiple objects in the same frame;
- Possibility to identify and associate several parts to a single object due to occasional partial occlusions;
- Generation of an output file with the labels and corresponding coordinates of all objects in each frame.

Fig. 5.9 shows the final release of the developed software. The main commands are:

- A slider on top of the window to set the video position (gross adjustment) or a box to insert the frame number (fine adjustment);
- "Open" button to open a window as shown in Fig. 5.10 and choose a video file to be marked;
- "Play" button to play the video and show the marked object, if a corresponding markup file exists;
- "Save" button to save the markup text file;



Set Position: 0

X = 304, Y = 161

Object Label: object

Sub Obj Number: 0

Fr. to jump: 10

Clear Object Label

Start: 0

End: 0

Frame rate: 26.56

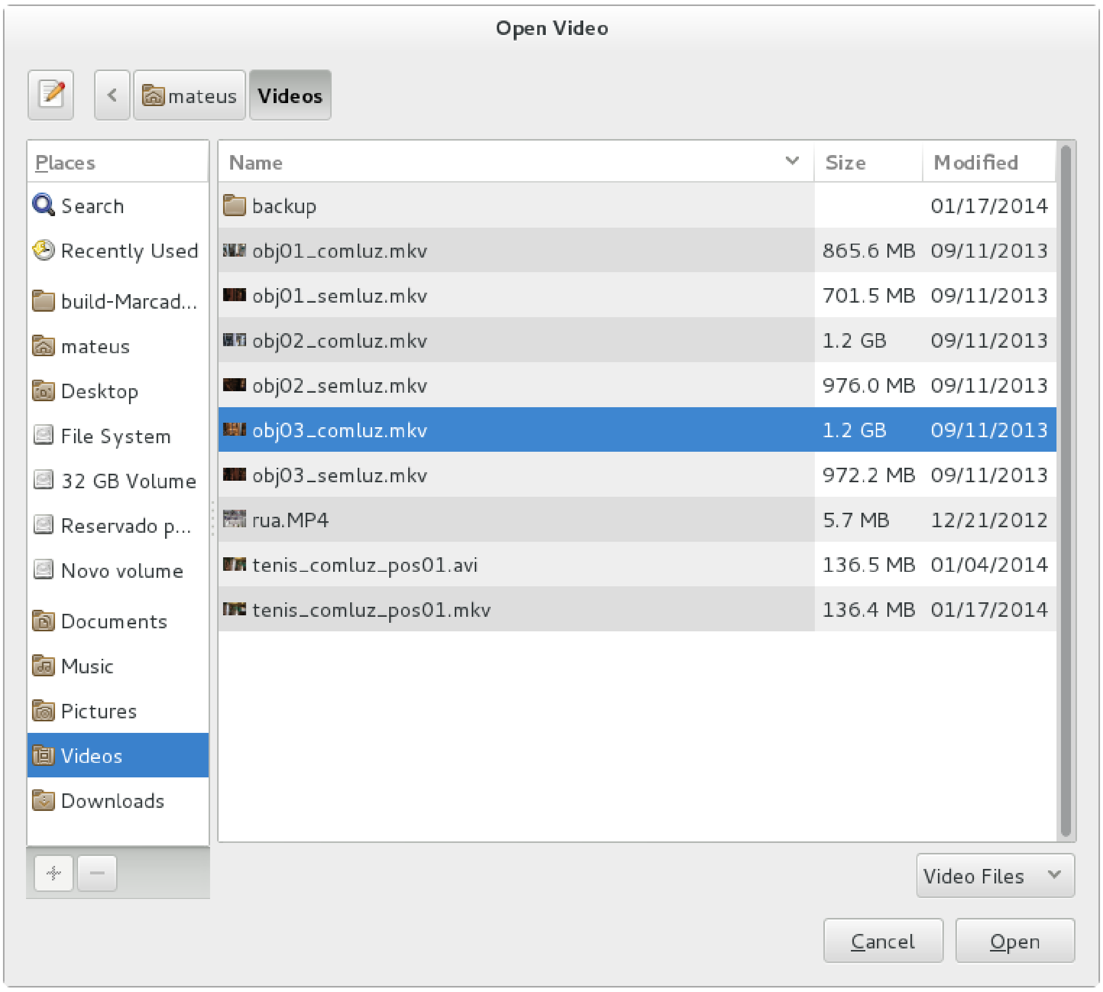
Figure 5.9: Screen of the marking software. In the left side are the commands. Some examples of marked objects are shown in the right side.

- "Set" button to set the object name and sub-index;
- "Jump" button to skip some frames without manual mark. In the skipped frames, an interpolation is performed;
- "Clear" button to clear a specific object from the output file, in a specified range of frames;
- "Frame rate" button to set the video frame rate, when the software cannot get it automatically from the video file.

The first step to mark a video is to open the target file and setting it to the desired frame position, using the slider or filling the frame number in he "Set Position" box. When first marking a video, the frame position should be set to 1; a different frame number is used when marking a video which has already been partially marked. The play video function can be used to see what video portion has already been marked.

For marking an object, a rectangular bounding box was chosen for its simplicity. The bounding box must surround all the target object. Initially, the mouse must be positioned at any bounding-box corner and dragged to the opposite corner with the left button pressed. When the left mouse button is released a box is drawn around the object, which is then validated with the “set” button.

The name of the object and its sub-index should be informed by the user. In the proposed syntax, full objects are marked with sub-index zero. In case an object



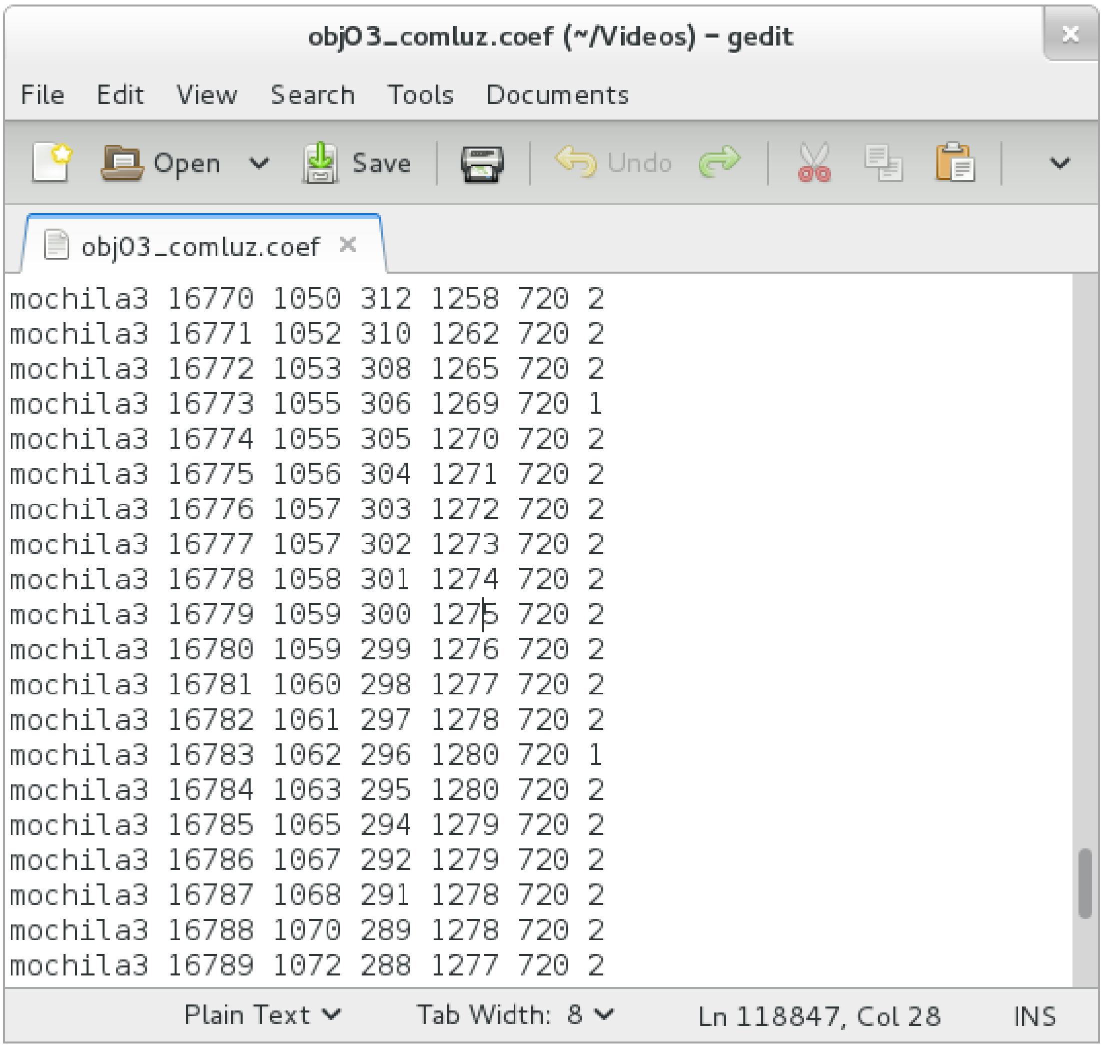
Name	Size	Modified
backup		01/17/2014
obj01_comluz.mkv	865.6 MB	09/11/2013
obj01_semluz.mkv	701.5 MB	09/11/2013
obj02_comluz.mkv	1.2 GB	09/11/2013
obj02_semluz.mkv	976.0 MB	09/11/2013
obj03_comluz.mkv	1.2 GB	09/11/2013
obj03_semluz.mkv	972.2 MB	09/11/2013
rua.MP4	5.7 MB	12/21/2012
tenis_comluz_pos01.avi	136.5 MB	01/04/2014
tenis_comluz_pos01.mkv	136.4 MB	01/17/2014

Figure 5.10: Window opened to choose the video.

is obstructed by another and it seems to be divided into several parts, each part receives a sub-index number varying from 1 to the number of parts.

To make the process faster, the user can skip a given number of frames and have the annotation tool to generate bounding boxes at interpolated positions (supposing they are at a constant speed). Although such interpolation process can lead to some marking error, at high frame rates (e.g., 30 frames/s), the variation between close frames (about 10 frames apart) is quite small and so is the inserted error.

The entire annotation process is summarized in a text file containing, in each line, the label of the object, the frame number and the coordinates of the upper-left and bottom-right corners of the bounding box for each object or sub-object, as shown in Fig. 5.11. The last number is a flag to indicate if the bounding box was generated by hand or interpolated. The number one indicates that the bounding box was generated by hand, and the number two that the bounding box was interpolated.



```
mochila3 16770 1050 312 1258 720 2
mochila3 16771 1052 310 1262 720 2
mochila3 16772 1053 308 1265 720 2
mochila3 16773 1055 306 1269 720 1
mochila3 16774 1055 305 1270 720 2
mochila3 16775 1056 304 1271 720 2
mochila3 16776 1057 303 1272 720 2
mochila3 16777 1057 302 1273 720 2
mochila3 16778 1058 301 1274 720 2
mochila3 16779 1059 300 1275 720 2
mochila3 16780 1059 299 1276 720 2
mochila3 16781 1060 298 1277 720 2
mochila3 16782 1061 297 1278 720 2
mochila3 16783 1062 296 1280 720 1
mochila3 16784 1063 295 1280 720 2
mochila3 16785 1065 294 1279 720 2
mochila3 16786 1067 292 1279 720 2
mochila3 16787 1068 291 1278 720 2
mochila3 16788 1070 289 1278 720 2
mochila3 16789 1072 288 1277 720 2
```

Figure 5.11: Coordinates file. Each row has the object label, the frame number, the coordinates of the upper-left corner and the bottom-right corner. The last number is a flag to indicate if the bounding box was generated by hand or interpolated.

Chapter 6

Optimization of the Parameters of the STC-mc Algorithm

This chapter describes the proposed methodology employed in the configuration of the STC-mc algorithm. This is achieved by using the VDAO database to determine the best parameter settings. Initially, the range of variation of each parameter is defined based on visual inspection of the results obtained in preliminary tests. Then the ground truth of the location of the abandoned objects provided with the VDAO database is employed to automatically compute metrics for the success of the anomaly detection operation. Using these metrics, the values of the parameters are tuned.

The ground truth of the VDAO database provides information such as the one depicted in Fig. 6.1. The area surrounding the abandoned object is marked with a blue box. We refer to a set of contiguous anomalous pixels as a blob. A true positive is an abandoned object with at least one pixel marked as anomalous, that is, a blob with non-monotonic empty intersection with a blue box. A false positive is a blob with no intersection with a blue box. A blue box with no intersection with any blob is considered a false negative. A true negative occurs when no blob has non-empty intersection with a blue box.

After the analysis of the output of the STC-mc algorithm, the number of true positives (TP) and false positives (FP) are used to plot a point of a FP× TP plane, similar to a ROC curve [54]. Each configuration of parameters generates a point on the curve. In our analysis we consider the best operating point the one closest to the point (0,1), which is the point with 0% of false positives and 100% of true positives.

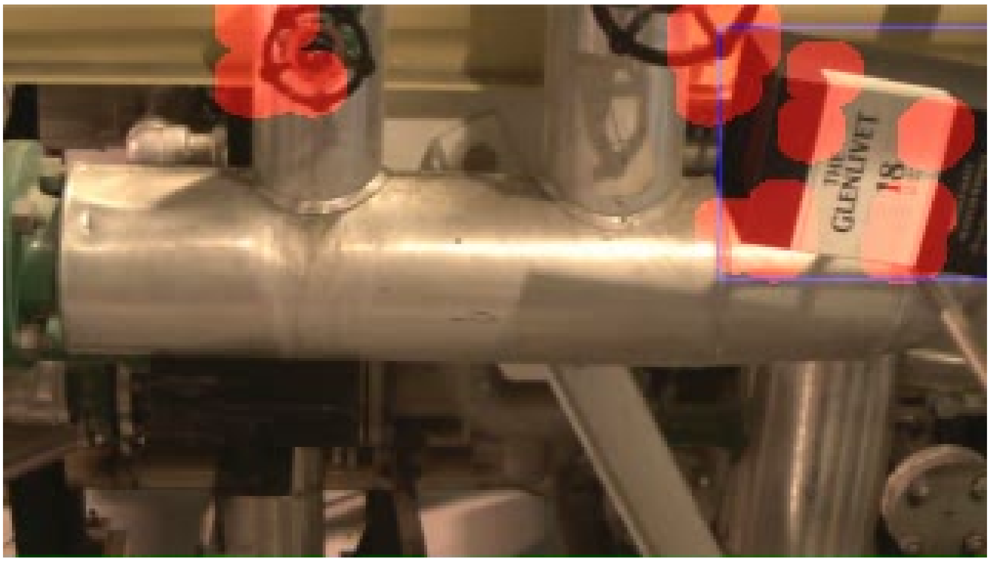


Figure 6.1: Example of a marked video and the criterion used for evaluation the detections. The area surrounding the abandoned object is marked with a blue box. If a blob contains at least one point inside the blue box, it is considered a true positive. A false negative occurs when there is no intersection between any blob and the interior of the box. A blob with all its points outside the blue box is considered a false positive. A true negative occurs if there are no blobs with all its points outside the blue box.

6.1 Parameters of the Descriptor and of the Codebook from the First Stage

There are three main parameters affecting the computation of the spatio-temporal descriptor used in STC-mc: (i) the standard deviation σ of the Gaussian temporal smoothing filter (see Section 4.5); (ii) the weight of the time derivative λ (Eq. (4.5)); (iii) the maximum distance ε_1 above which two codewords are considered to be different (algorithm in Fig. 3.4).

Table 6.1: Kernel coefficients of the Gaussian filter.

σ	-2	-1	0	1	2
0.8	0.021930	0.228512	0.499116	0.228512	0.021930
1.0	0.054489	0.244201	0.402620	0.244201	0.054489
2.0	0.085629	0.242668	0.343406	0.242668	0.085629

The set of values of σ used in the tests was {0.8,1.0,1.2}. Fig. 6.2 shows the plot of the Eq. (4.6) for these values of σ, and the coefficients of the kernel of the Gaussian filter is shown in the Table 6.1. The weight λ, that is affected by the frame rate of the videos and their resolution, and is used to adjust the sensitivity of the system to the temporal derivative assumed the values {2.5,2.0,1.5}. Lastly,

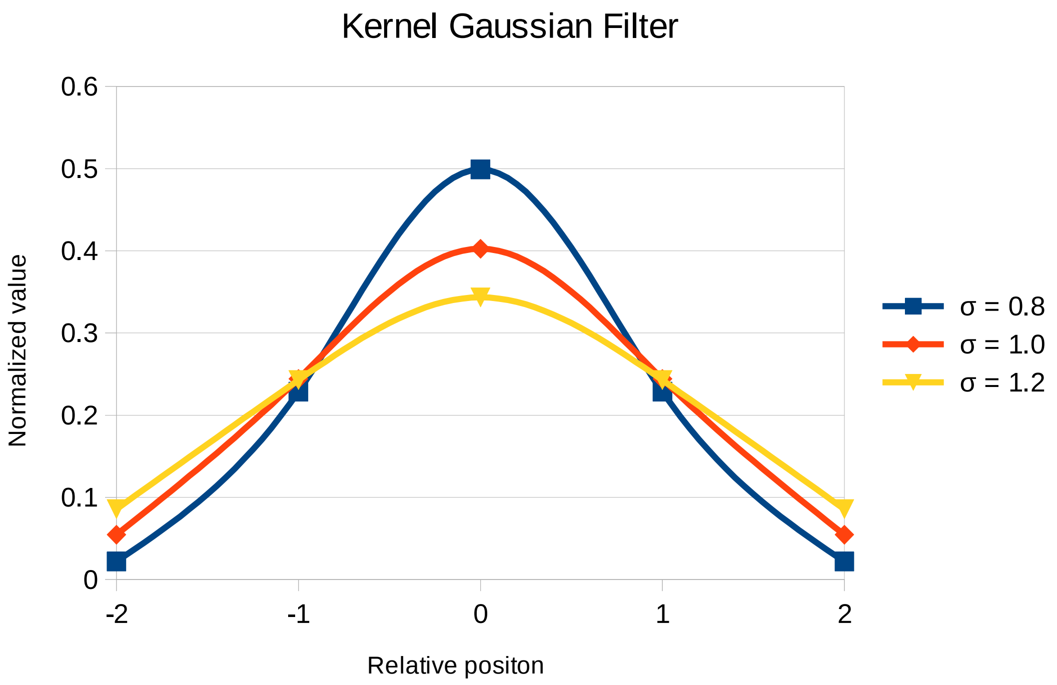


Figure 6.2: Gaussian filter curves generated from the Eq. (4.6).

the maximum distance ε_1 is used during the creation of the first codebook. It is the limit to consider a descriptor as represented by a codeword that is already in the codebook or that a new codeword has to be inserted in the codebook to represent it. The range of variation tested for ε_1 was {600, 700, 800}. As the three parameters are interdependent, all their $3 \times 3 \times 3 = 27$ combinations were evaluated, generating a cloud of points instead of a classic ROC curve. In this type of plot, the best operating points are the ones with smallest distance to the point (0,1) in the plane FP×TP. Figs. 6.3 to 6.5 show examples of the videos utilized in the test. The abandoned objects were a whisky box, a sneaker and a towel, respectively. The plot of the results obtained with these videos are shown in Figs. 6.6 to 6.8, and the numerical results are presented in the Tables 6.2 to 6.4. The criteria used to choose the best setting was the sum of the distances to the point (0,1) in the ROC curve, for the three videos. The best setting is the one with the smallest sum. Table 6.5 shows the result of this sum for the three videos. The parameter set λ, σ and ε_1 that provide the best results, for these three objects overall is $\{2.0, 1.0, 700\}$. For this configuration, the TP×FP values in the Figs. 6.6, 6.7 and 6.8 are {0.99×0.20, 0.98×0.01, 1.00×0.00}, respectively.

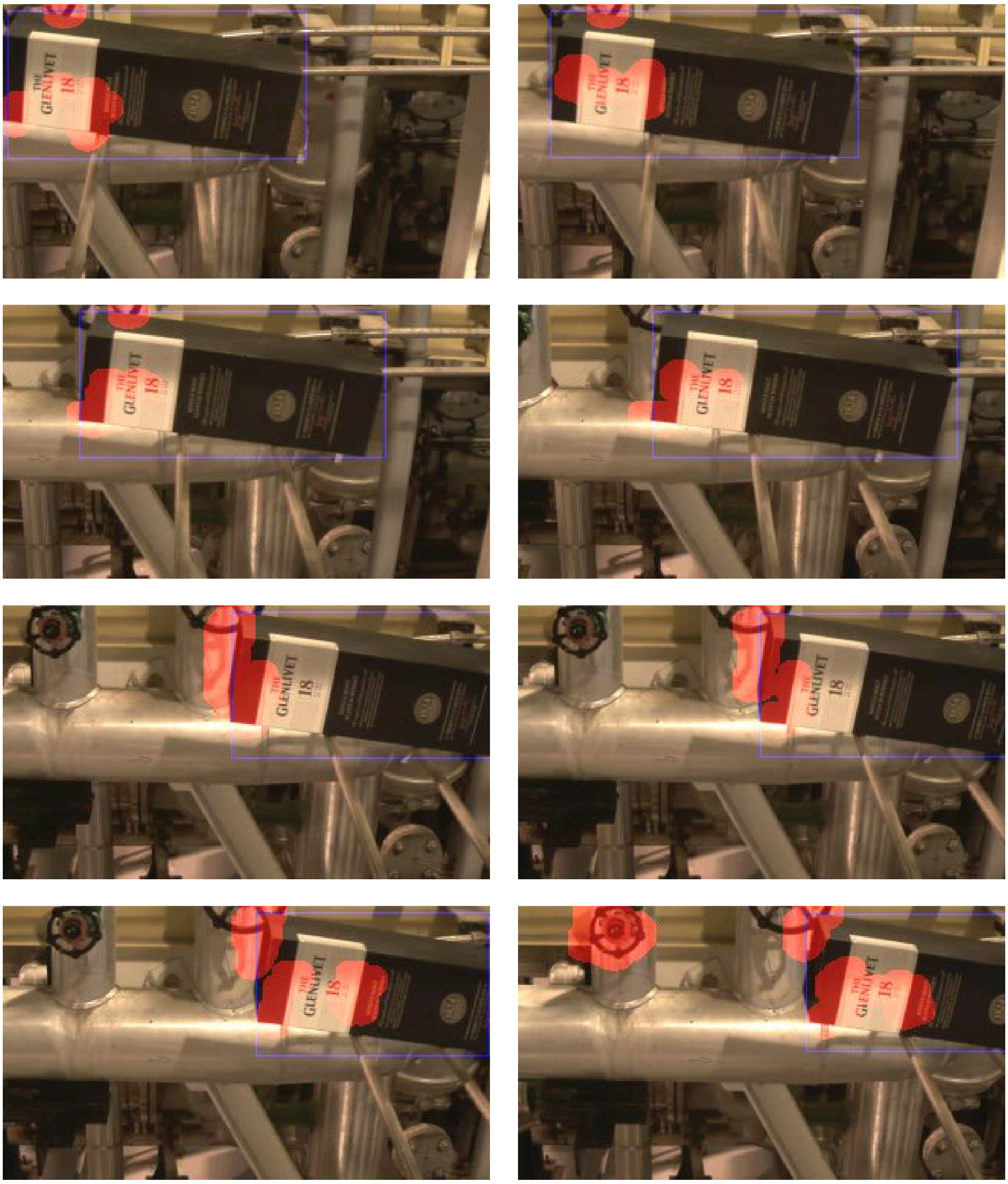


Figure 6.3: Example of one of the tests when the abandoned object is a whisky box. A valve is always detected, generation a false positive.

6.2 Second Codebook Parameters

To create the codebook from the second stage, it is necessary to determine the value of the maximum distance ε_2, the distance weight μ, and the probability weight ν (see the algorithm in Figs. 4.4 and 4.12). Like ε_1 for the first codebook, ε_2 is the distance above which a descriptor is considered as not represented by a codeword existent in the second codebook. The set of values utilized in the tests for ε_2 was $\{500, 600, 700\}$.

The parameters μ and ν are used during the analysis of a video as described in

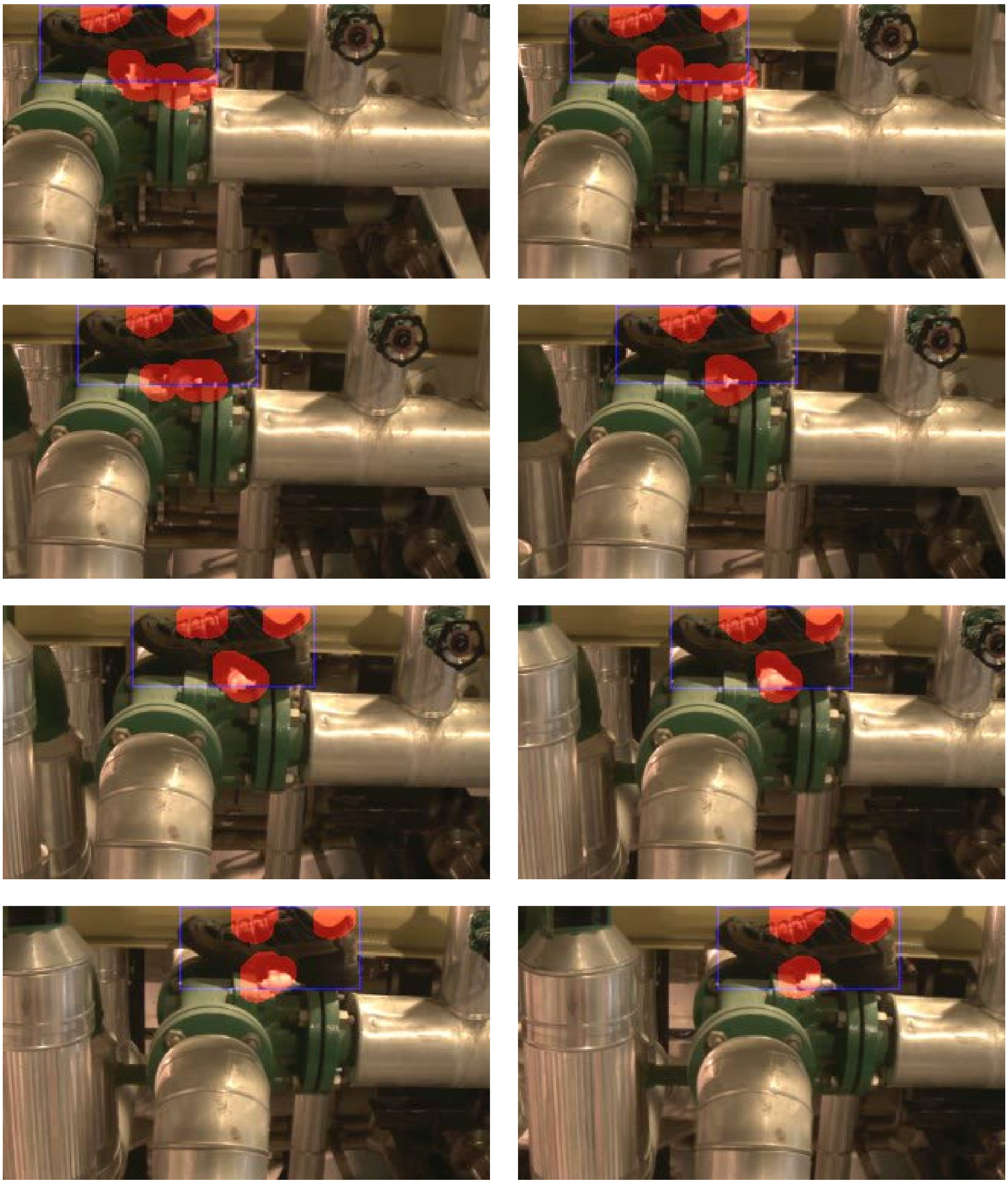


Figure 6.4: Example of one of the tests when the abandoned object is a sneaker.

Fig. 4.12. They are necessary because, in a surveillance system, although the video to be analyzed is recorded in roughly the same conditions as the reference video, often there are differences in illumination and even small differences in camera positioning. Therefore, one must introduce some tolerance both when matching a descriptor to a codeword and when applying the probability threshold. In the proposed second stage dictionary, μ and ν are the distance and probability tolerance, respectively. Initially, the set of values of μ employed in the tests was $\{1.0,\ 1.05,\ 1.10\}$. The set used for ν was $\{2, 4, 6, 8\}$. All the $3\times3\times4 = 36$ combinations were simulated. As has been performed for the first pass dictionary (Section 6.1), their results generated a

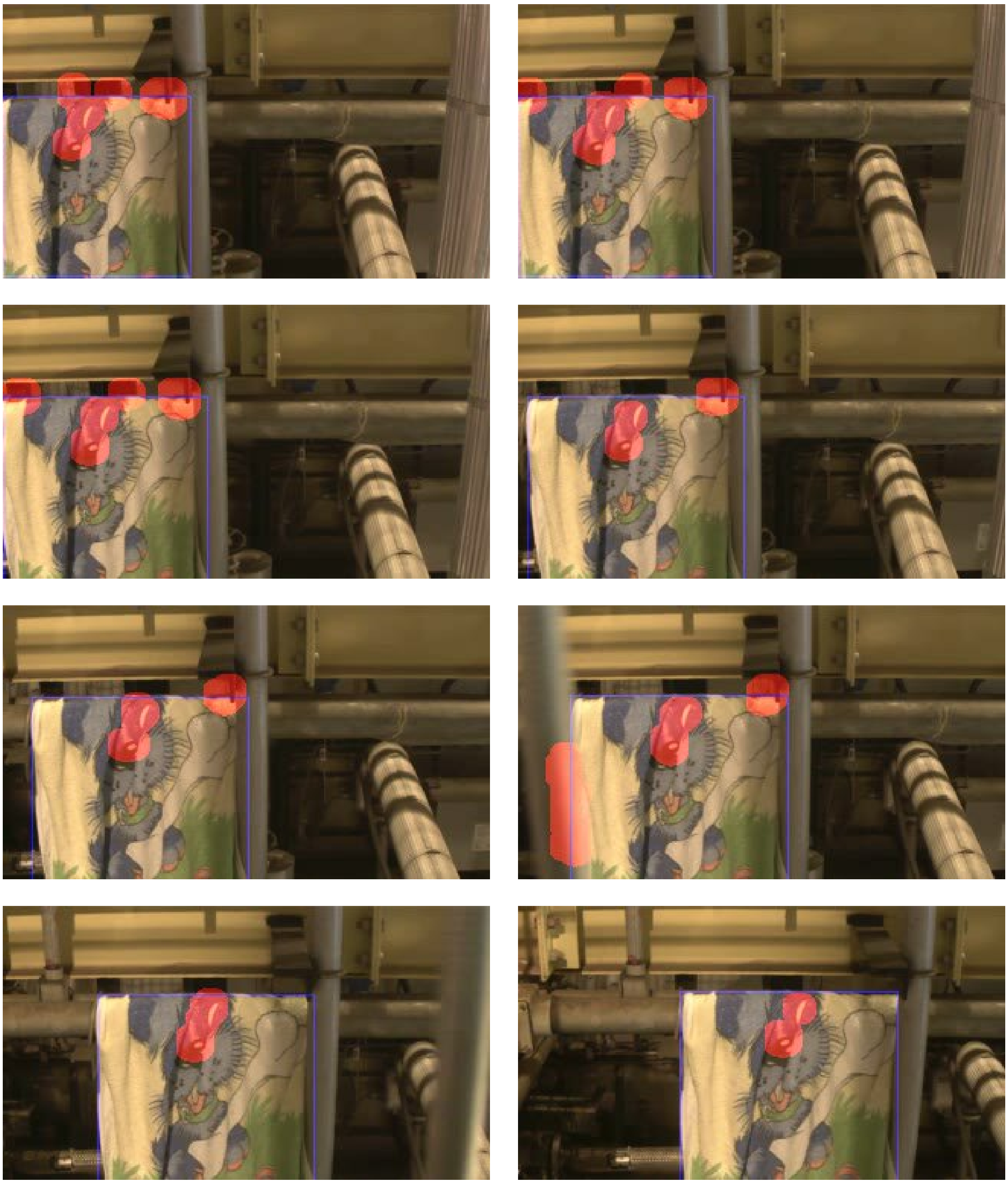


Figure 6.5: Example of one of the tests when the abandoned object is a towel.

cloud of points on the FP×TP plane. After the simulation with these set of values, the best results were obtained when $\varepsilon_2 = 600$. So, to get more accurate results, additional simulations were performed only with $\varepsilon_2 = 600$. For this case, the final μ set was $\{1.0, 1.05, 1.06, 1.08, 1.1\}$ and ν set was $\{2, 4, 6, 7, 8\}$. The total number of simulation was 49, being 12 with $\varepsilon_2 = 500$, 12 with $\varepsilon_2 = 700$ and 25 with $\varepsilon_2 = 600$. The results for the videos with whisky box, the sneaker and the towel are shown in Figs. 6.9 to 6.11, respectively. In the same order, the Tabs. 6.6 to 6.8 show the numeric results of the tests.

Again, the criteria used to choose the best setting was the sum of the distances

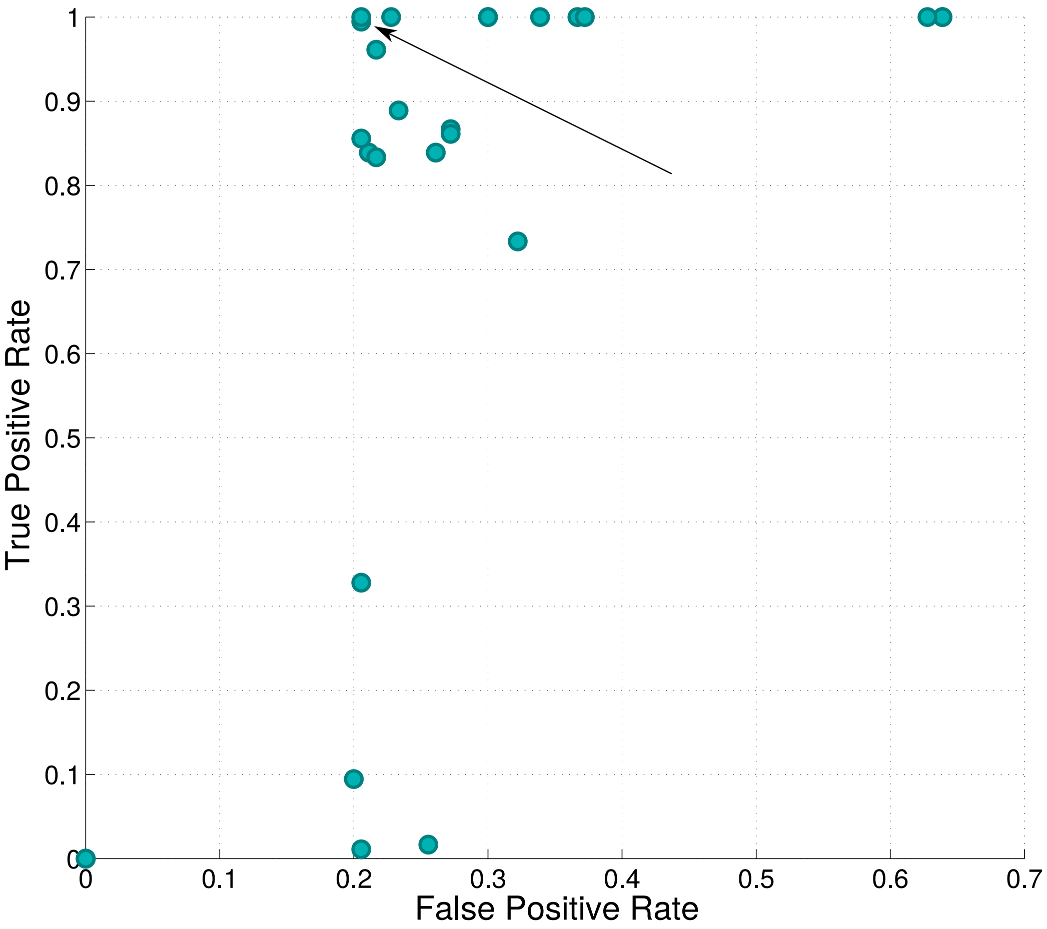


Figure 6.6: Scatter results of descriptor parameters estimation tests when the abandoned object is a whisky box. The parameters are the filter kernel standard deviation σ, the temporal derivative weight λ and the codebook formation distance ε_1. The arrow indicates the result when the values of σ, λ and ε_1 are $\{1.0, 2.0, 700\}$, respectively.

to the point (0,1) in the ROC curve, for the three videos. The best setting is the one with the smallest sum. Table 6.9 shows the result of this sum for the three videos. The set of parameters (ε_2, μ, ν) that provide the best overall results (points with smallest distances to the point (0,1) in the TP×FP plane), taking into account the three sets of points in Figs. 6.9 to 6.11, is $\{600, 1.08, 7\}$. For this configuration, the FP×TP values in the Figs. 6.9, 6.10 and 6.11 are {1.00×0.04, 0.90×0.04, 0.92×0.01}, respectively.

Tabs. 6.10 and 6.11 show a summary of the best values obtained in tests.

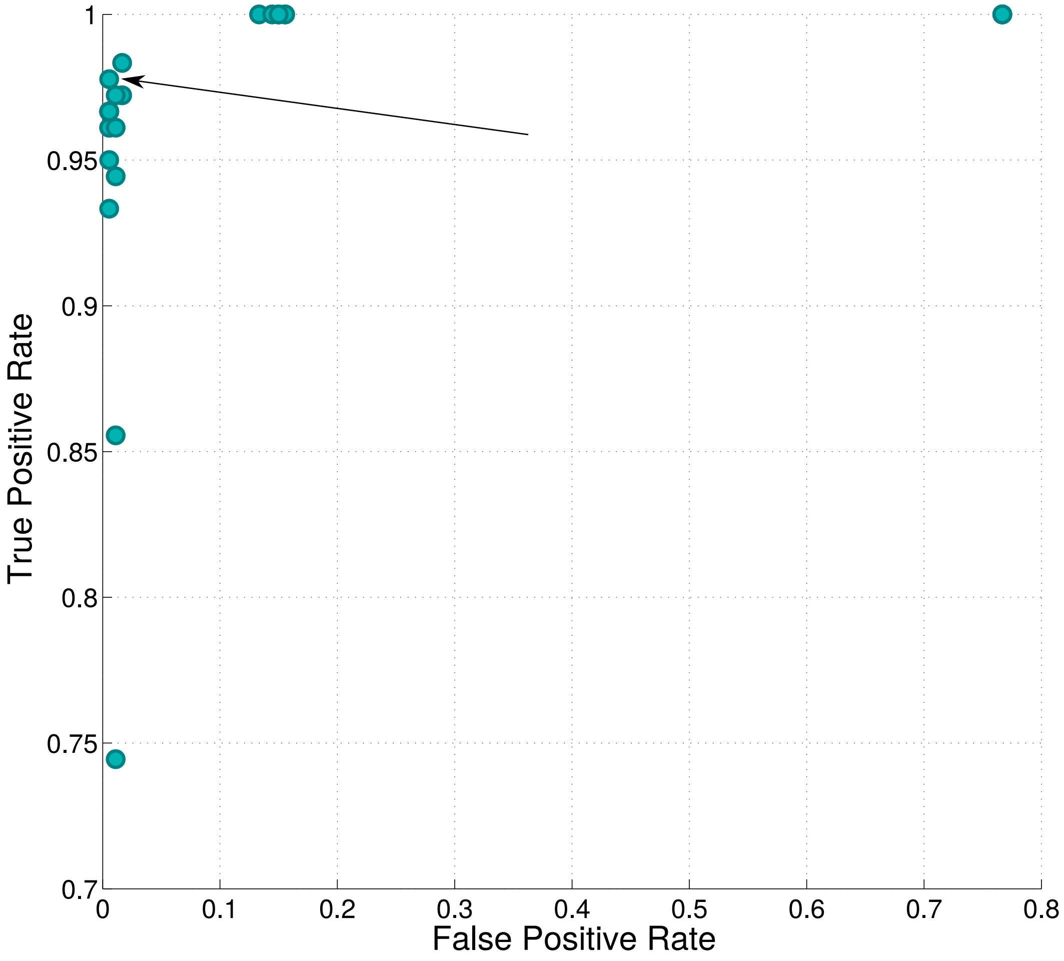


Figure 6.7: Results of descriptor parameters estimation tests when the abandoned object is a sneaker. The parameters are the filter kernel standard deviation σ, the temporal derivative weight λ and the codebook formation distance ε_1. The arrow indicates the result when the values of σ, λ and ε_1 are $\{1.0, 2.0, 700\}$, respectively.

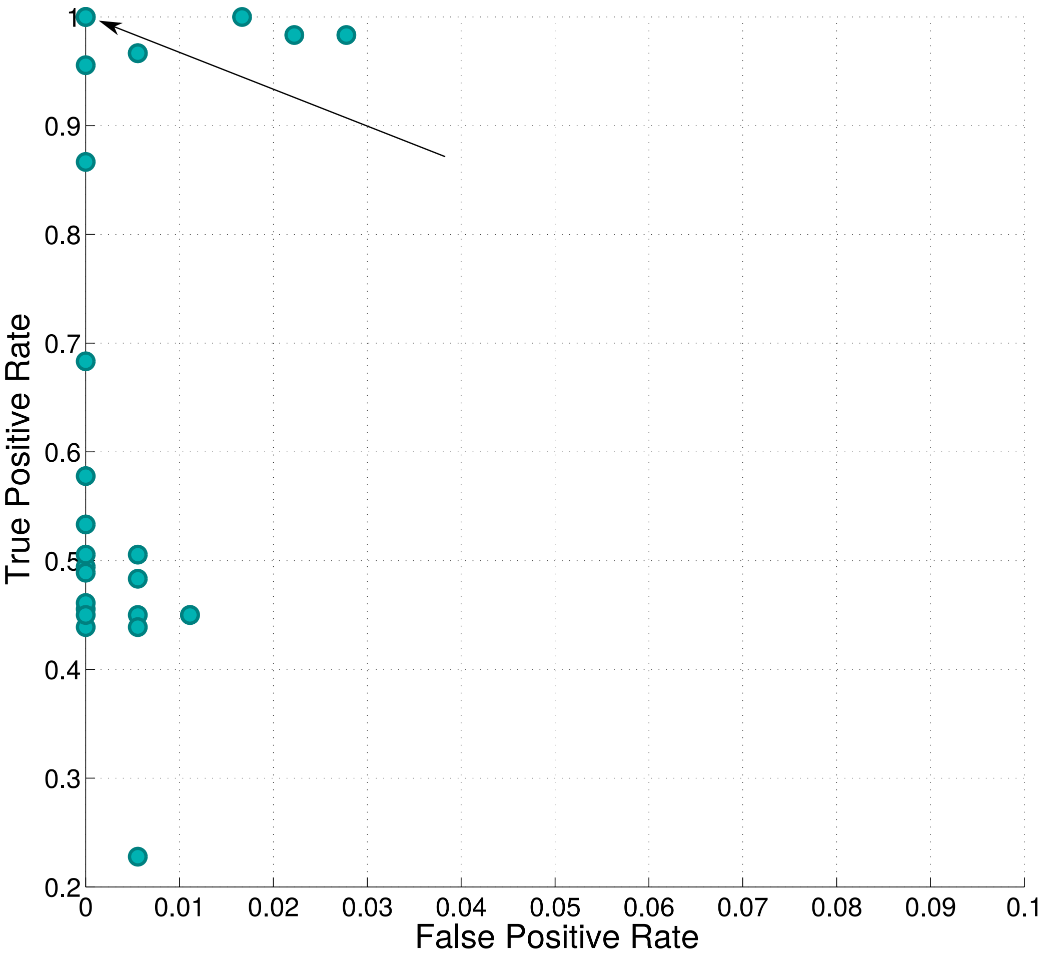


Figure 6.8: Scatter results of descriptor parameters estimation tests when the abandoned object is a towel. The parameters are the filter kernel standard deviation σ, the temporal derivative weight λ and the codebook formation distance ε_1. The arrow indicates the result when the values of σ, λ and ε_1 are $\{1.0, 2.0, 700\}$, respectively.

Table 6.2: Results of descriptor parameters estimation tests when the abandoned object is a whisky box.

Item	λ	σ	ε_1	TP	FP	Dis. (0,1)
1	2.50	0.80	700	1.00	0.64	0.64
2	2.50	0.80	800	1.00	0.64	0.64
3	2.50	0.80	900	1.00	0.63	0.63
4	2.50	1.00	700	0.73	0.32	0.42
5	2.50	1.00	800	1.00	0.30	0.30
6	2.50	1.00	900	1.00	0.30	0.30
7	2.50	1.20	700	0.87	0.27	0.30
8	2.50	1.20	800	1.00	0.23	0.23
9	2.50	1.20	900	0.86	0.27	0.31
10	2.00	0.80	700	1.00	0.37	0.37
11	2.00	0.80	800	1.00	0.37	0.37
12	2.00	0.80	900	1.00	0.34	0.34
13	2.00	1.00	700	0.99	0.21	0.21
14	2.00	1.00	800	1.00	0.21	0.21
15	2.00	1.00	900	0.96	0.22	0.22
16	2.00	1.20	700	0.84	0.21	0.27
17	2.00	1.20	800	0.86	0.21	0.25
18	2.00	1.20	900	0.00	0.00	1.00
19	1.50	0.80	700	0.89	0.23	0.26
20	1.50	0.80	800	0.83	0.22	0.27
21	1.50	0.80	900	0.02	0.26	1.02
22	1.50	1.00	700	0.84	0.26	0.31
23	1.50	1.00	800	0.09	0.20	0.93
24	1.50	1.00	900	0.00	0.00	1.00
25	1.50	1.20	700	0.33	0.21	0.70
26	1.50	1.20	800	0.01	0.21	1.01
27	1.50	1.20	900	0.00	0.00	1.00

Table 6.3: Scatter results of descriptor parameters estimation tests when the abandoned object is a sneaker.

Item	λ	σ	ε_1	TP	FP	Dis. (0,1)
1	2.50	0.80	700	1.00	0.77	0.77
2	2.50	0.80	800	1.00	0.77	0.77
3	2.50	0.80	900	1.00	0.77	0.77
4	2.50	1.00	700	1.00	0.13	0.13
5	2.50	1.00	800	1.00	0.15	0.15
6	2.50	1.00	900	1.00	0.14	0.14
7	2.50	1.20	700	0.98	0.02	0.02
8	2.50	1.20	800	0.97	0.02	0.03
9	2.50	1.20	900	0.96	0.01	0.04
10	2.00	0.80	700	1.00	0.16	0.16
11	2.00	0.80	800	1.00	0.16	0.16
12	2.00	0.80	900	1.00	0.15	0.15
13	2.00	1.00	700	0.98	0.01	0.02
14	2.00	1.00	800	0.97	0.01	0.03
15	2.00	1.00	900	0.97	0.01	0.03
16	2.00	1.20	700	0.96	0.01	0.04
17	2.00	1.20	800	0.95	0.01	0.05
18	2.00	1.20	900	0.86	0.01	0.14
19	1.50	0.80	700	0.97	0.01	0.03
20	1.50	0.80	800	0.97	0.01	0.03
21	1.50	0.80	900	0.97	0.01	0.03
22	1.50	1.00	700	0.97	0.01	0.03
23	1.50	1.00	800	0.97	0.01	0.03
24	1.50	1.00	900	0.94	0.01	0.06
25	1.50	1.20	700	0.96	0.01	0.04
26	1.50	1.20	800	0.93	0.01	0.07
27	1.50	1.20	900	0.74	0.01	0.26

Table 6.4: Results of descriptor parameters estimation tests when the abandoned object is a towel.

Item	λ	σ	ε_1	TP	FP	Dis. (0,1)
1	2.50	0.80	700	0.98	0.03	0.03
2	2.50	0.80	800	0.98	0.02	0.03
3	2.50	0.80	900	1.00	0.02	0.02
4	2.50	1.00	700	0.97	0.01	0.03
5	2.50	1.00	800	0.97	0.01	0.03
6	2.50	1.00	900	0.68	0.00	0.32
7	2.50	1.20	700	0.87	0.00	0.13
8	2.50	1.20	800	0.49	0.00	0.51
9	2.50	1.20	900	0.46	0.00	0.54
10	2.00	0.80	700	1.00	0.00	0.00
11	2.00	0.80	800	1.00	0.00	0.00
12	2.00	0.80	900	0.96	0.00	0.04
13	2.00	1.00	700	1.00	0.00	0.00
14	2.00	1.00	800	0.48	0.01	0.52
15	2.00	1.00	900	0.44	0.00	0.56
16	2.00	1.20	700	0.51	0.01	0.49
17	2.00	1.20	800	0.45	0.01	0.55
18	2.00	1.20	900	0.45	0.01	0.55
19	1.50	0.80	700	0.58	0.00	0.42
20	1.50	0.80	800	0.51	0.00	0.49
21	1.50	0.80	900	0.46	0.00	0.54
22	1.50	1.00	700	0.53	0.00	0.47
23	1.50	1.00	800	0.49	0.00	0.51
24	1.50	1.00	900	0.23	0.01	0.77
25	1.50	1.20	700	0.45	0.00	0.55
26	1.50	1.20	800	0.45	0.01	0.55
27	1.50	1.20	900	0.44	0.01	0.56

Table 6.5: The best configuration of λ, σ and ε_1 is the one with the smaller sum of the distances to the point (0,1) of the three objects, in the ROC curve.

Sum of the distance to the point (0,1) of the three objects							
Item	λ	σ	ε_1	Whisky	Sneaker	Towel	Sum
1	2.50	0.80	700	0.64	0.77	0.03	1.44
2	2.50	0.80	800	0.64	0.77	0.03	1.43
3	2.50	0.80	900	0.63	0.77	0.02	1.41
4	2.50	1.00	700	0.42	0.13	0.03	0.59
5	2.50	1.00	800	0.30	0.15	0.03	0.48
6	2.50	1.00	900	0.30	0.14	0.32	0.76
7	2.50	1.20	700	0.30	0.02	0.13	0.46
8	2.50	1.20	800	0.23	0.03	0.51	0.77
9	2.50	1.20	900	0.31	0.04	0.54	0.89
10	2.00	0.80	700	0.37	0.16	0.00	0.52
11	2.00	0.80	800	0.37	0.16	0.00	0.53
12	2.00	0.80	900	0.34	0.15	0.04	0.53
13	2.00	1.00	700	0.21	0.02	0.00	0.23
14	2.00	1.00	800	0.21	0.03	0.52	0.76
15	2.00	1.00	900	0.22	0.03	0.56	0.82
16	2.00	1.20	700	0.27	0.04	0.49	0.80
17	2.00	1.20	800	0.25	0.05	0.55	0.85
18	2.00	1.20	900	1.00	0.14	0.55	1.69
19	1.50	0.80	700	0.26	0.03	0.42	0.71
20	1.50	0.80	800	0.27	0.03	0.49	0.80
21	1.50	0.80	900	1.02	0.03	0.54	1.59
22	1.50	1.00	700	0.31	0.03	0.47	0.81
23	1.50	1.00	800	0.93	0.03	0.51	1.47
24	1.50	1.00	900	1.00	0.06	0.77	1.83
25	1.50	1.20	700	0.70	0.04	0.55	1.29
26	1.50	1.20	800	1.01	0.07	0.55	1.63
27	1.50	1.20	900	1.00	0.26	0.56	1.82

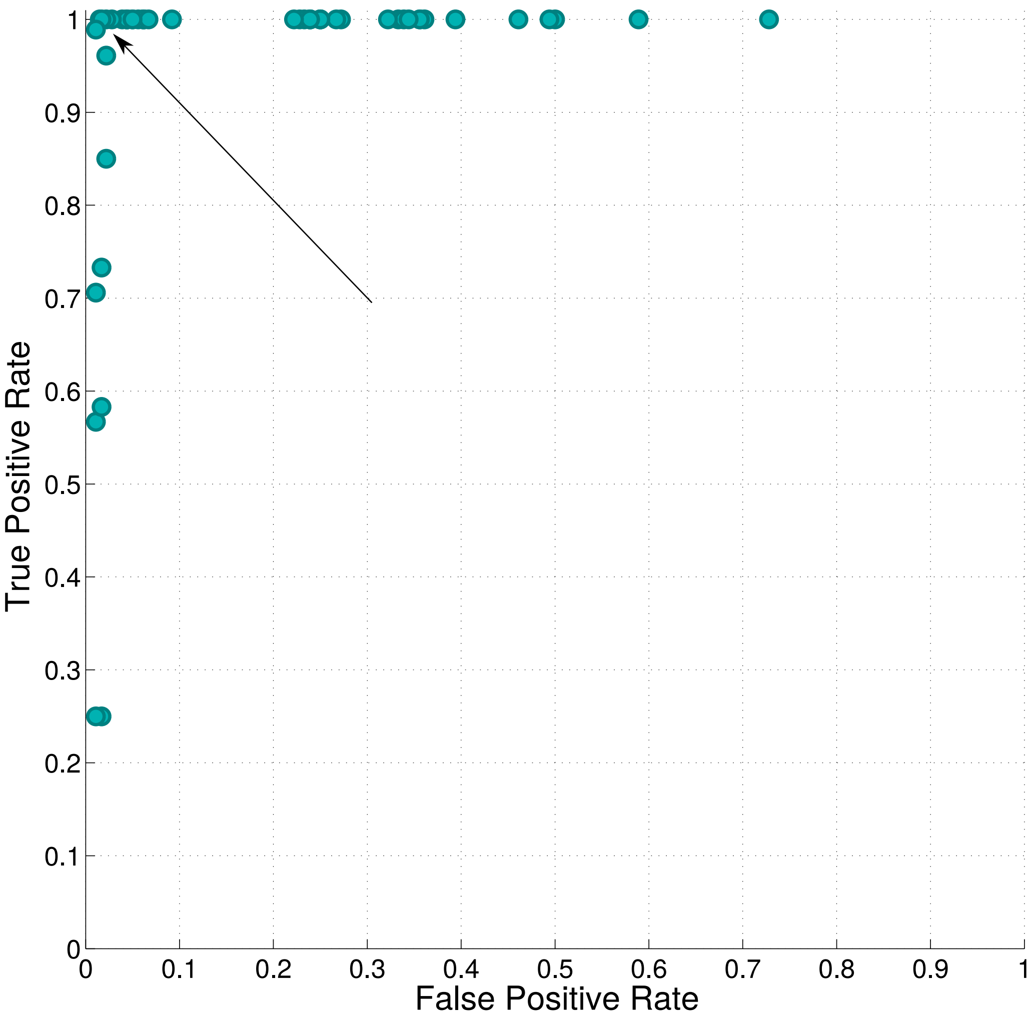


Figure 6.9: Scatter results for the parameters of the codebook of the second stage, that is, maximum distance ε_2, the distance weight μ, and the probability weight ν. The abandoned object was the whisky box. The arrow indicates the result when the values of ε_2, μ, ν are $\{600, 1.08, 7\}$, respectively.

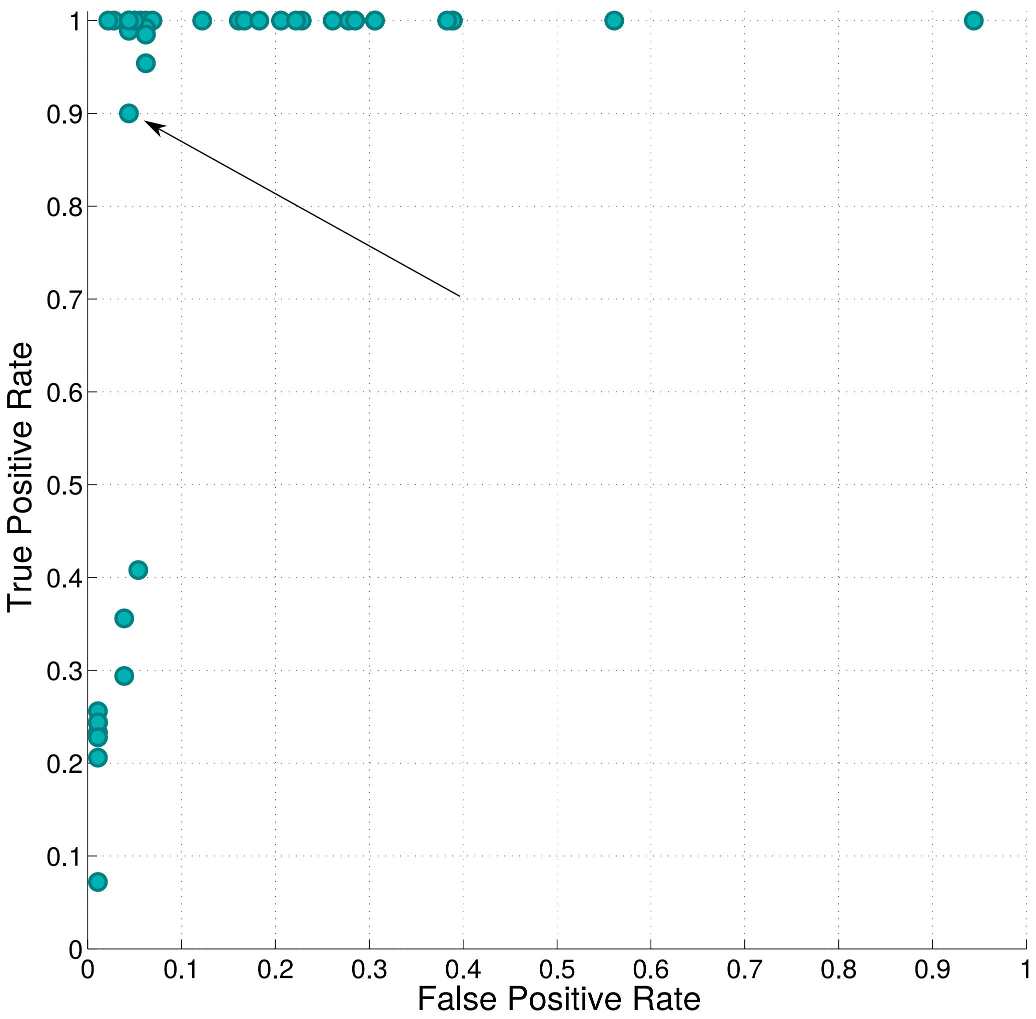


Figure 6.10: Scatter results for the parameters of the codebook of the second stage, that is, maximum distance ε_2, the distance weight μ, and the probability weight ν. The abandoned object was the sneaker. The arrow indicates the result when the values of ε_2, μ, ν are $\{600, 1.08, 7\}$, respectively.

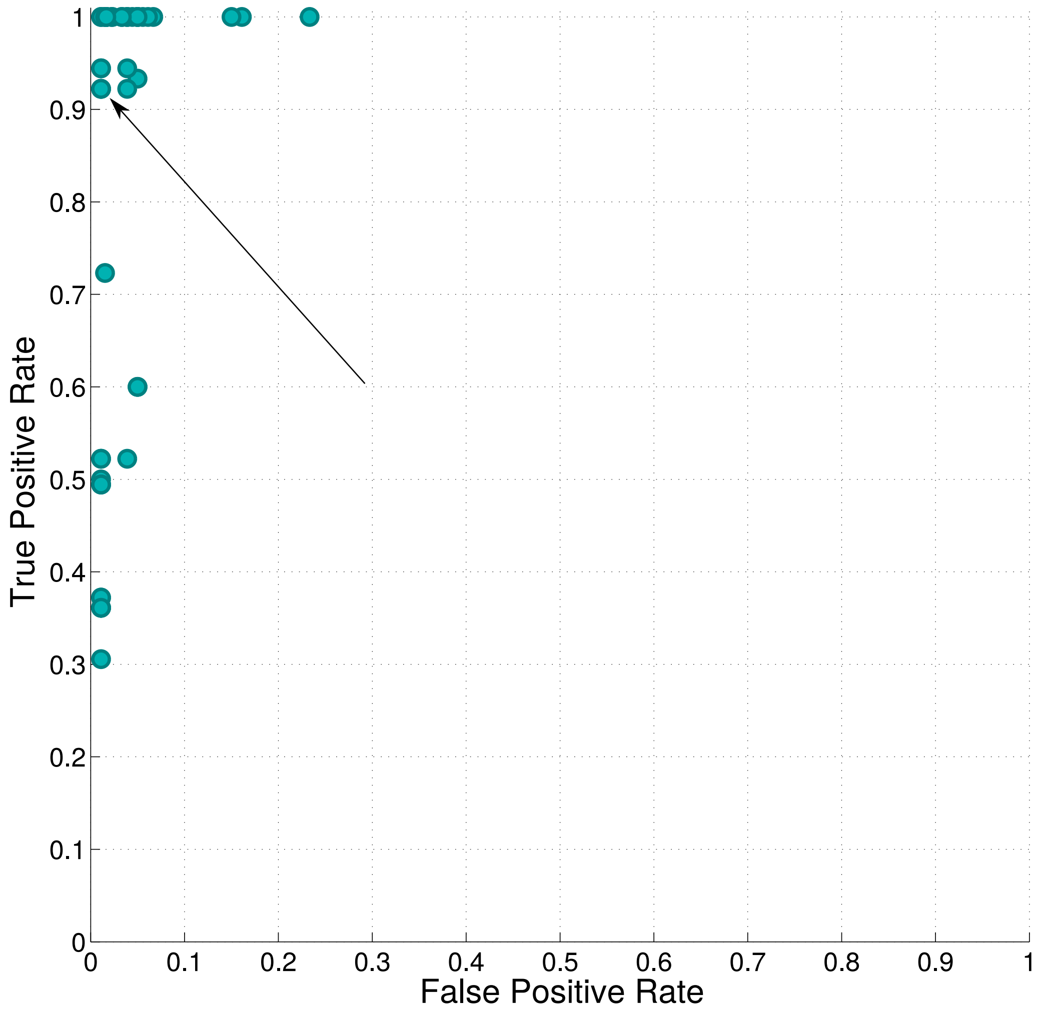


Figure 6.11: Scatter results for the parameters of the codebook of the second stage, that is, maximum distance ε_2, the distance weight μ, and the probability weight ν. The abandoned object was a towel. The arrow indicates the result when the values of ε_2, μ, ν are $\{600, 1.08, 7\}$, respectively.

Table 6.6: Results of the codebook of the second stage parameters estimation tests when the abandoned object is a whisky box.

Item	ε_2	μ	ν	TP	FP	Dis. (0,1)
1	500	1.00	2.00	1.00	0.73	0.73
2	500	1.00	4.00	1.00	0.59	0.59
3	500	1.00	6.00	1.00	0.50	0.50
4	500	1.00	8.00	1.00	0.46	0.46
5	500	1.05	2.00	1.00	0.49	0.49
6	500	1.05	4.00	1.00	0.39	0.39
7	500	1.05	6.00	1.00	0.33	0.33
8	500	1.05	8.00	1.00	0.34	0.34
9	500	1.10	2.00	1.00	0.36	0.36
10	500	1.10	4.00	1.00	0.32	0.32
11	500	1.10	6.00	1.00	0.23	0.23
12	500	1.10	8.00	1.00	0.24	0.24
13	600	1.00	2.00	1.00	0.36	0.36
14	600	1.00	4.00	1.00	0.27	0.27
15	600	1.00	6.00	1.00	0.25	0.25
16	600	1.00	7.00	1.00	0.25	0.25
17	600	1.00	8.00	1.00	0.25	0.25
18	600	1.05	2.00	1.00	0.36	0.36
19	600	1.05	4.00	1.00	0.27	0.27
20	600	1.05	6.00	1.00	0.25	0.25
21	600	1.05	7.00	1.00	0.23	0.23
22	600	1.05	8.00	1.00	0.23	0.23
23	600	1.06	2.00	1.00	0.34	0.34
24	600	1.06	4.00	1.00	0.24	0.24
25	600	1.06	6.00	1.00	0.24	0.24
26	600	1.06	7.00	1.00	0.23	0.23
27	600	1.06	8.00	1.00	0.22	0.22
28	600	1.08	2.00	1.00	0.06	0.06
29	600	1.08	4.00	1.00	0.05	0.05
30	600	1.08	6.00	1.00	0.04	0.04
31	600	1.08	7.00	1.00	0.04	0.04
32	600	1.08	8.00	1.00	0.06	0.06
33	600	1.10	2.00	1.00	0.07	0.07
34	600	1.10	4.00	1.00	0.04	0.04
35	600	1.10	6.00	1.00	0.04	0.04
36	600	1.10	7.00	1.00	0.02	0.05
37	600	1.10	8.00	1.00	0.05	0.05
38	700	1.00	2.00	1.00	0.03	0.03
39	700	1.00	4.00	1.00	0.02	0.02
40	700	1.00	6.00	1.00	0.02	0.02
41	700	1.00	8.00	0.96	0.02	0.04
42	700	1.05	2.00	0.99	0.01	0.02
43	700	1.05	4.00	0.85	0.02	0.15
44	700	1.05	6.00	0.73	0.02	0.27
45	700	1.05	8.00	0.57	0.01	0.43
46	700	1.10	2.00	0.71	0.01	0.29
47	700	1.10	4.00	0.58	0.02	0.42
48	700	1.10	6.00	0.25	0.02	0.75
49	700	1.10	8.00	0.25	0.01	0.75

Table 6.7: Results of the codebook of the second stage parameters estimation tests when the abandoned object is a sneaker.

Item	ε_2	μ	ν	TP	FP	Dis. (0,1)
1	500	1.00	2.00	1.00	0.56	0.56
2	500	1.00	4.00	1.00	0.31	0.31
3	500	1.00	6.00	1.00	0.28	0.28
4	500	1.00	8.00	1.00	0.23	0.23
5	500	1.05	2.00	1.00	0.31	0.31
6	500	1.05	4.00	1.00	0.16	0.16
7	500	1.05	6.00	1.00	0.12	0.12
8	500	1.05	8.00	1.00	0.06	0.06
9	500	1.10	2.00	1.00	0.26	0.26
10	500	1.10	4.00	1.00	0.17	0.17
11	500	1.10	6.00	1.00	0.03	0.03
12	500	1.10	8.00	1.00	0.02	0.02
13	600	1.00	2.00	1.00	0.18	0.18
14	600	1.00	4.00	1.00	0.04	0.04
15	600	1.00	6.00	1.00	0.04	0.04
16	600	1.00	7.00	1.00	0.06	0.06
17	600	1.00	8.00	1.00	0.044	0.04
18	600	1.05	2.00	1.00	0.21	0.21
19	600	1.05	4.00	1.00	0.05	0.05
20	600	1.05	6.00	0.99	0.04	0.05
21	600	1.05	7.00	0.99	0.06	0.06
22	600	1.05	8.00	0.36	0.04	0.65
23	600	1.06	2.00	1.00	0.29	0.29
24	600	1.06	4.00	1.00	0.07	0.07
25	600	1.06	6.00	0.99	0.06	0.06
26	600	1.06	7.00	0.99	0.06	0.06
27	600	1.06	8.00	0.41	0.05	0.60
28	600	1.08	2.00	1.00	0.94	0.94
29	600	1.08	4.00	1.00	0.39	0.39
30	600	1.08	6.00	1.00	0.39	0.39
31	600	1.08	7.00	0.90	0.04	0.10
32	600	1.08	8.00	1.00	0.38	0.38
33	600	1.10	2.00	1.00	0.22	0.22
34	600	1.10	4.00	1.00	0.05	0.05
35	600	1.10	6.00	1.00	0.04	0.04
36	600	1.10	7.00	0.95	0.06	0.08
37	600	1.10	8.00	0.29	0.04	0.71
38	700	1.00	2.00	0.26	0.01	0.75
39	700	1.00	4.00	0.24	0.01	0.76
40	700	1.00	6.00	0.24	0.01	0.76
41	700	1.00	8.00	0.24	0.01	0.76
42	700	1.05	2.00	0.24	0.01	0.76
43	700	1.05	4.00	0.23	0.01	0.77
44	700	1.05	6.00	0.23	0.01	0.77
45	700	1.05	8.00	0.21	0.01	0.80
46	700	1.10	2.00	0.24	0.01	0.76
47	700	1.10	4.00	0.23	0.01	0.77
48	700	1.10	6.00	0.23	0.01	0.77
49	700	1.10	8.00	0.07	0.01	0.93

Table 6.8: Results of the codebook of the second stage parameters estimation tests when the abandoned object is a towel.

Item	ε_2	μ	ν	TP	FP	Dis. (0,1)
1	500	1.00	2.00	1.00	0.23	0.23
2	500	1.00	4.00	1.00	0.07	0.07
3	500	1.00	6.00	1.00	0.05	0.05
4	500	1.00	8.00	1.00	0.04	0.04
5	500	1.05	2.00	1.00	0.16	0.16
6	500	1.05	4.00	1.00	0.05	0.05
7	500	1.05	6.00	1.00	0.04	0.04
8	500	1.05	8.00	1.00	0.03	0.03
9	500	1.10	2.00	1.00	0.15	0.15
10	500	1.10	4.00	1.00	0.06	0.06
11	500	1.10	6.00	1.00	0.02	0.02
12	500	1.10	8.00	1.00	0.02	0.02
13	600	1.00	2.00	1.00	0.06	0.23
14	600	1.00	4.00	1.00	0.04	0.04
15	600	1.00	6.00	1.00	0.02	0.02v
16	600	1.00	7.00	1.00	0.02	0.02
17	600	1.00	8.00	1.00	0.02	0.02
18	600	1.05	2.00	1.00	0.05	0.16
19	600	1.05	4.00	1.00	0.03	0.03
20	600	1.05	6.00	1.00	0.01	0.01
21	600	1.05	7.00	1.00	0.02	0.02
22	600	1.05	8.00	1.00	0.01	0.01
23	600	1.06	2.00	0.93	0.05	0.08
24	600	1.06	4.00	0.94	0.04	0.07
25	600	1.06	6.00	0.94	0.01	0.06
26	600	1.06	7.00	0.94	0.02	0.06
27	600	1.06	8.00	0.94	0.01	0.06
28	600	1.08	2.00	0.60	0.05	0.40
29	600	1.08	4.00	0.92	0.04	0.09
30	600	1.08	6.00	0.92	0.01	0.08
31	600	1.08	7.00	0.92	0.01	0.08
32	600	1.08	8.00	0.92	0.01	0.08
33	600	1.10	2.00	0.60	0.05	0.15
34	600	1.10	4.00	0.52	0.04	0.48
35	600	1.10	6.00	0.52	0.01	0.48
36	600	1.10	7.00	0.52	0.02	0.48
37	600	1.10	8.00	0.52	0.01	0.48
38	700	1.00	2.00	0.50	0.01	0.50
39	700	1.00	4.00	0.49	0.01	0.51
40	700	1.00	6.00	0.49	0.01	0.51
41	700	1.00	8.00	0.49	0.01	0.51
42	700	1.05	2.00	0.37	0.01	0.63
43	700	1.05	4.00	0.36	0.01	0.64
44	700	1.05	6.00	0.36	0.01	0.64
45	700	1.05	8.00	0.36	0.01	0.64
46	700	1.10	2.00	0.37	0.01	0.63
47	700	1.10	4.00	0.36	0.01	0.64
48	700	1.10	6.00	0.31	0.01	0.69
49	700	1.10	8.00	1.00	0.02	0.02

Table 6.9: The best configuration of ε_2, μ and ν is the one with the smaller sum of the distances to the point (0,1) of the three objects, in the ROC curve of the second codebook parameters.

Sum of the distance to the point (0,1) of the 3 objects							
Item	ε_2	μ	ν	Whisky	Sneaker	Towel	Sum
1	500	1.00	2.00	0.73	0.56	0.23	1.52
2	500	1.00	4.00	0.59	0.31	0.07	0.96
3	500	1.00	6.00	0.50	0.28	0.05	0.83
4	500	1.00	8.00	0.46	0.23	0.04	0.73
5	500	1.05	2.00	0.49	0.31	0.16	0.96
6	500	1.05	4.00	0.39	0.16	0.05	0.61
7	500	1.05	6.00	0.33	0.12	0.04	0.49
8	500	1.05	8.00	0.34	0.06	0.03	0.43
9	500	1.10	2.00	0.36	0.26	0.15	0.77
10	500	1.10	4.00	0.32	0.17	0.06	0.54
11	500	1.10	6.00	0.23	0.03	0.02	0.28
12	500	1.10	8.00	0.24	0.02	0.02	0.28
13	600	1.00	2.00	0.36	0.18	0.23	0.77
14	600	1.00	4.00	0.27	0.04	0.04	0.36
15	600	1.00	6.00	0.25	0.04	0.02	0.31
16	600	1.00	7.00	0.25	0.06	0.02	0.33
17	600	1.00	8.00	0.25	0.04	0.02	0.31
18	600	1.05	2.00	0.36	0.21	0.16	0.72
19	600	1.05	4.00	0.27	0.05	0.03	0.35
20	600	1.05	6.00	0.25	0.05	0.01	0.31
21	600	1.05	7.00	0.23	0.06	0.02	0.31
22	600	1.05	8.00	0.23	0.65	0.01	0.88
23	600	1.06	2.00	0.34	0.21	0.08	0.64
24	600	1.06	4.00	0.24	0.07	0.07	0.38
25	600	1.06	6.00	0.24	0.06	0.06	0.36
26	600	1.06	7.00	0.23	0.06	0.06	0.35
27	600	1.06	8.00	0.22	0.59	0.06	0.87
28	600	1.08	2.00	0.06	0.94	0.40	1.41
29	600	1.08	4.00	0.05	0.39	0.09	0.53
30	600	1.08	6.00	0.04	0.39	0.08	0.51
31	600	1.08	7.00	0.04	0.11	0.08	0.23
32	600	1.08	8.00	0.06	0.38	0.08	0.52
33	600	1.10	2.00	0.07	0.22	0.15	0.44
34	600	1.10	4.00	0.04	0.05	0.48	0.57
35	600	1.10	6.00	0.04	0.04	0.48	0.57
36	600	1.10	7.00	0.05	0.08	0.48	0.60
37	600	1.10	8.00	0.05	0.71	0.48	1.23
38	700	1.00	2.00	0.03	0.74	0.50	1.27
39	700	1.00	4.00	0.02	0.76	0.51	1.28
40	700	1.00	6.00	0.02	0.76	0.51	1.28
41	700	1.00	8.00	0.04	0.76	0.51	1.31
42	700	1.05	2.00	0.02	0.76	0.63	1.40
43	700	1.05	4.00	0.15	0.77	0.64	1.56
44	700	1.05	6.00	0.27	0.77	0.64	1.67
45	700	1.05	8.00	0.43	0.79	0.64	1.87
46	700	1.10	2.00	0.29	0.76	0.63	1.68
47	700	1.10	4.00	0.42	0.77	0.64	1.83
48	700	1.10	6.00	0.75	0.77	0.69	2.22
49	700	1.10	8.00	0.75	0.93	0.02	1.69

Table 6.10: Results obtained when the parameters σ, λ and ε_1 are set to $\{1.0, 2.0, 700\}$.

Object	TP	FP	Dis. to (0,1)
Whisky box	0.99	0.20	0.20
Sneaker	0.98	0.01	0.02
Towel	1.00	0.00	0.00

Table 6.11: Results obtained when the parameters ε_2, μ, ν are set to $\{600, 1.08, 7\}$.

Object	TP	FP	Dis. to (0,1)
Whisky box	1.00	0.04	0.04
Sneaker	0.90	0.04	0.11
Towel	0.92	0.01	0.08

Chapter 7

Experimental Results

In this Chapter, we assess the results of the methodology employed in the configuration of parameters of the STC-mc algorithm (see Chapter 6). After configuring these parameters, several simulations were performed using other videos of the VDAO database. This way we could assess both the performance of the STC-mc algorithm and the robustness of the choice of the parameters of the algorithm by the methodology described in Chapter 6.

The Table 7.1 shows the results of the simulations for seven videos from the VDAO database with abandoned objects. In each video there was an abandoned object: a whisky box (two different positions), a towel, a sneaker, a bottle, a camera box and a bowl. An implementation of the BOV was used too, to compare the results. This implementation uses only a dense sampling and a dictionary of BOV to analyze the video. The STC method uses only the threshold γ to detect the anomalies, and it was not possible to find a threshold in which only true positives were present in every frame of the video. Also from Table 1 we note that the original STC method always generates (FP,TP) points that are distant from ideal point (0,1) for the detection of static objects with a moving camera.

The parameter set was $\{\lambda, \varepsilon_1, \sigma, \gamma, \varepsilon_2, \mu, \nu\}$ configured to $\{2, 700, 1, 1 \times 10^{-7}, 600, 1.08, 7\}$. Figs. 7.1 to 7.7 show the results obtained. In these figures the abandoned objects are identified and painted red. Even though not all parts of the abandoned objects were identified, in almost all the cases at least one pixel of the object were identified as anomalous. In a real application where the main objective is to identify the presence of an anomaly, the red spot will be shown in the screen and it will be enough to call the attention of an operator.

The proposed STC-mc algorithm has a good performance detecting the abandoned object in five of the seven videos, with a TP rate greater than 90% and a FP rate lower than 13%. In the videos where the object was not so well detected, the object was very similar to the background and the system was not able to distinguish it. In Fig. 7.6, STC-mc was not able distinguish the whisky box from the

Table 7.1: Comparison of the results obtained. TP is the true positive rate, FP is the false positive rate and DIS is the distance to the point (0,1) in the FP×TP plane, which is the best possible point. The STC-mc algorithm either outperforms or perform very close to the BOV implementation. The STC implementation has the worse results and is not suitable to this kind of application.

Object	STC-mc			BOV			STC		
	TP	FP	DIS	TP	FP	DIS	TP	FP	DIS
Whisky 1	1.00	0.04	0.04	1.00	0.22	0.22	0.92	0.82	0.83
Towel	0.92	0.01	0.08	1.00	0.01	0.01	0.93	0.52	0.53
Sneaker	0.90	0.04	0.11	1.00	0.04	0.04	0.92	0.94	0.53
Bottle	0.99	0.13	0.13	0.99	0.27	0.27	0.00	1.00	1.41
Camera Box	1.00	0.03	0.03	1.00	0.01	0.01	1.00	0.79	0.79
Whisky 2	0.37	0.42	0.76	0.48	0.64	0.83	0.58	1.00	1.08
Bowl	0.29	0.64	0.96	0.01	0.69	1.21	0.85	1.00	1.01

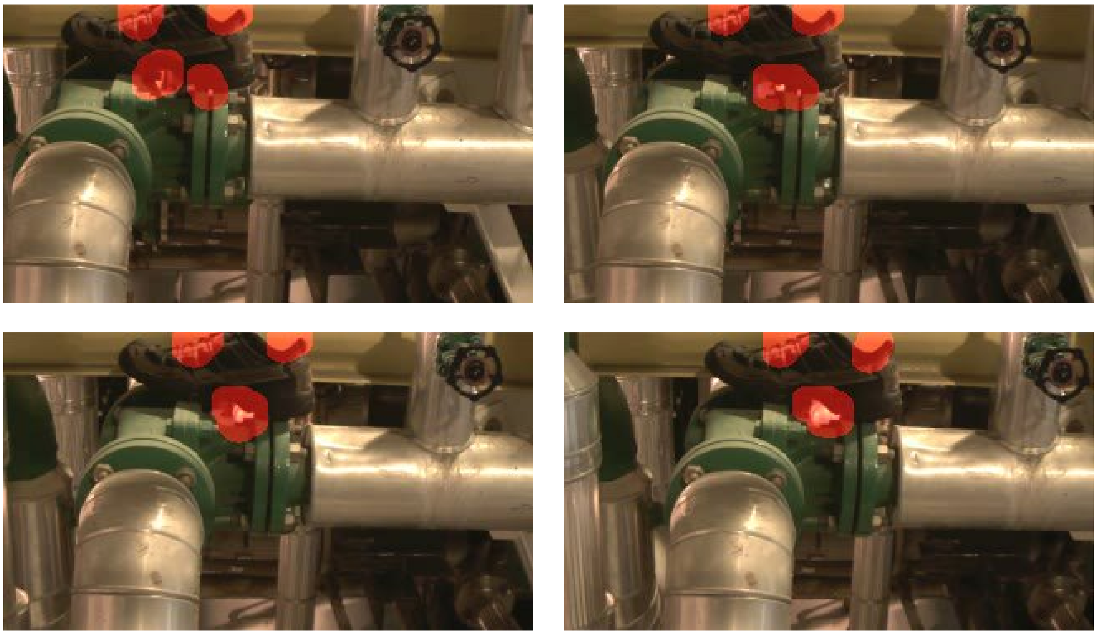


Figure 7.1: In this simulation the sneaker was the abandoned object and it was detected as an anomaly. The points of the object detected as anomalous are painted in red.

background in about 60% of the frames, and in the Fig. 7.7 the false negative rate was about 70%.

As can be seen in Table 7.1, when the STC-mc is compared with the BOV implementation, in three of the videos the BOV approach performs slightly better than STC-mc (less than 7%). However, the results of the STC-mc algorithm in these cases are also very good, with a high TP rate and a low FP rate. On the other hand, in four of the seven simulations the STC-mc performs much better result than BOV.

Some examples of the results of the comparison of the three methods obtained in Table 7.1 are shown in Figs. 7.8 and 7.9.

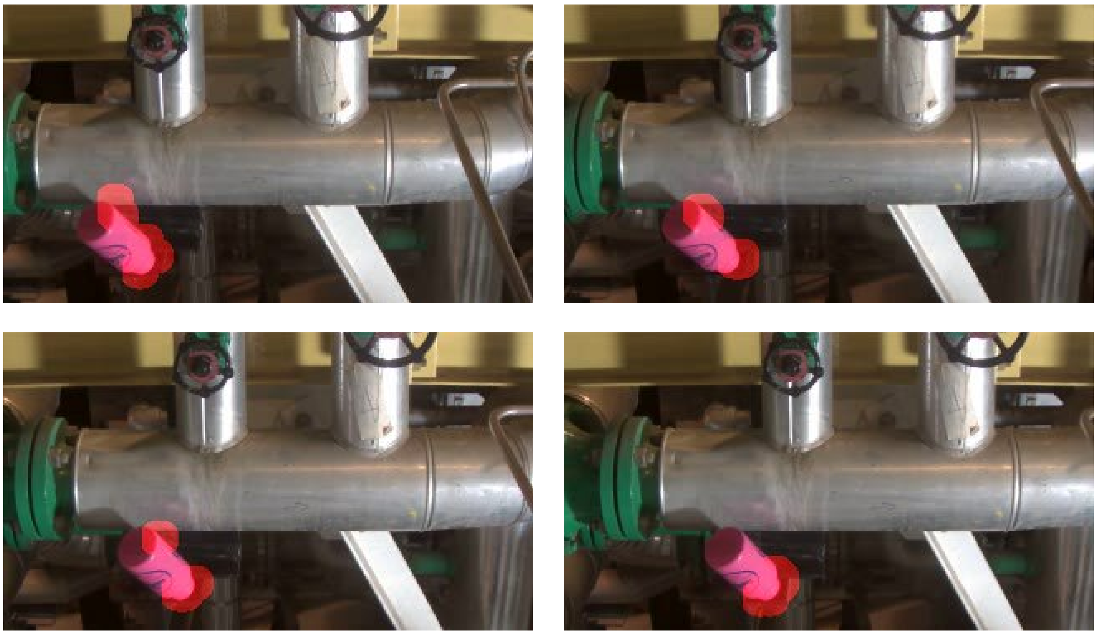


Figure 7.2: The bottle was the abandoned object and it was detected as anomaly. The points of the object detected as anomalous are painted in red.

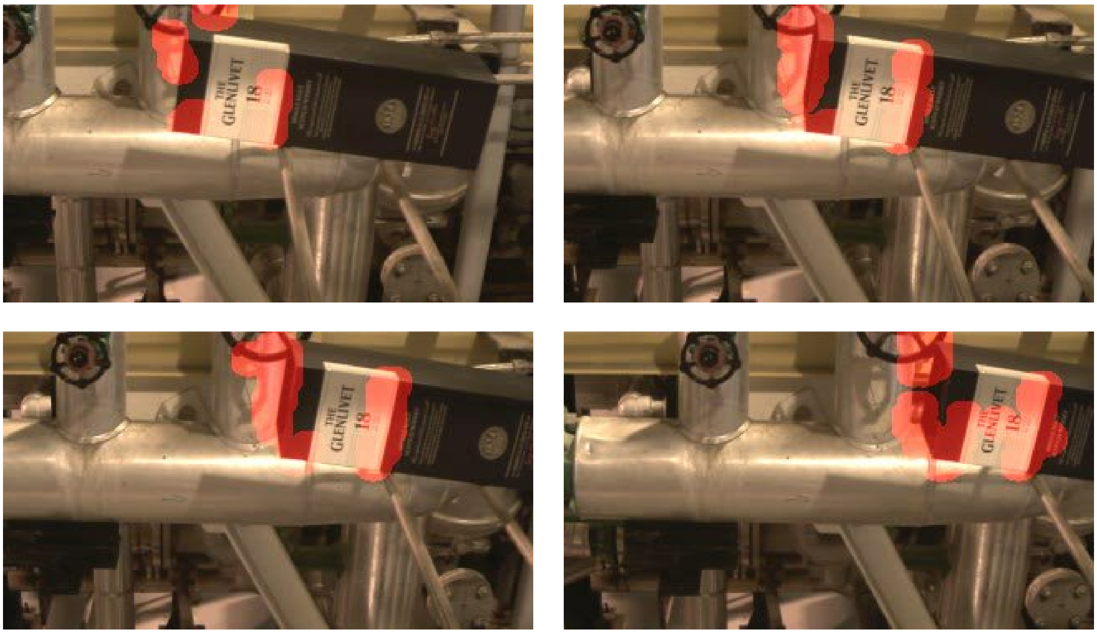


Figure 7.3: In this simulation the whisky box was the abandoned object and it was detected as an anomaly. The points of the object detected as anomalous are painted in red.

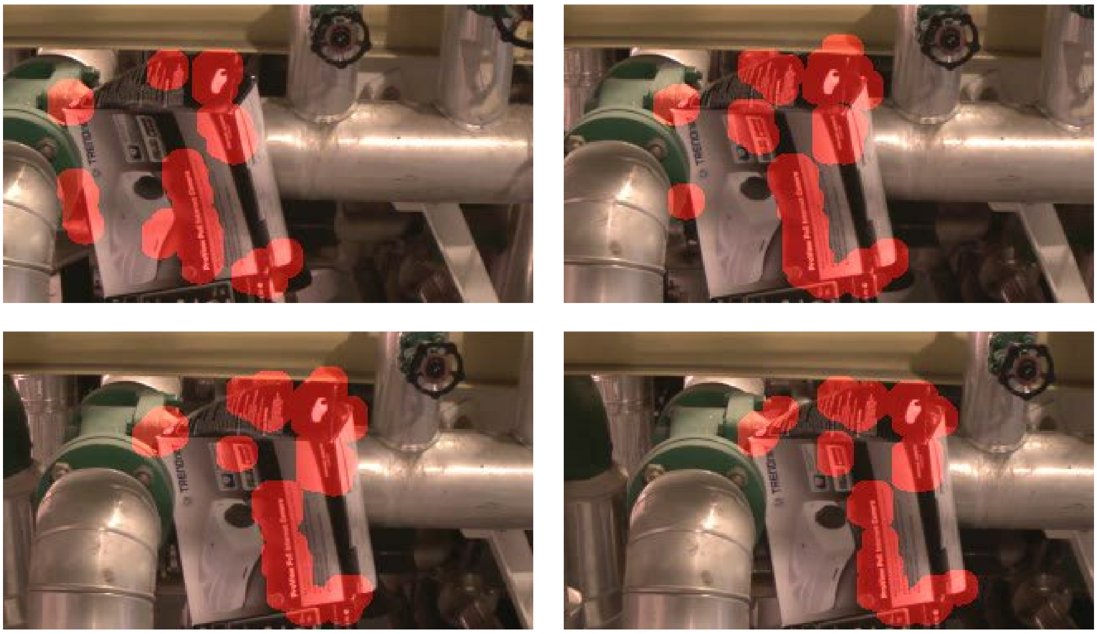


Figure 7.4: The camera box was the abandoned object and it was detected as anomaly. The points of the object detected as anomalous are painted in red.

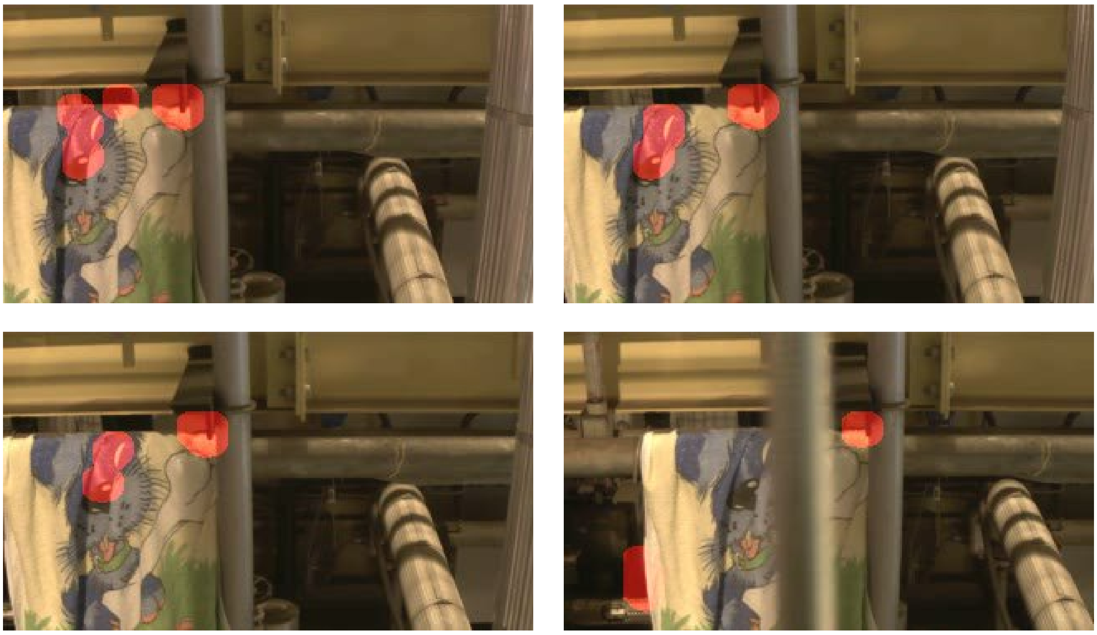


Figure 7.5: The towel was the abandoned object and it was detected as anomaly. The points of the object detected as anomalous are painted in red.

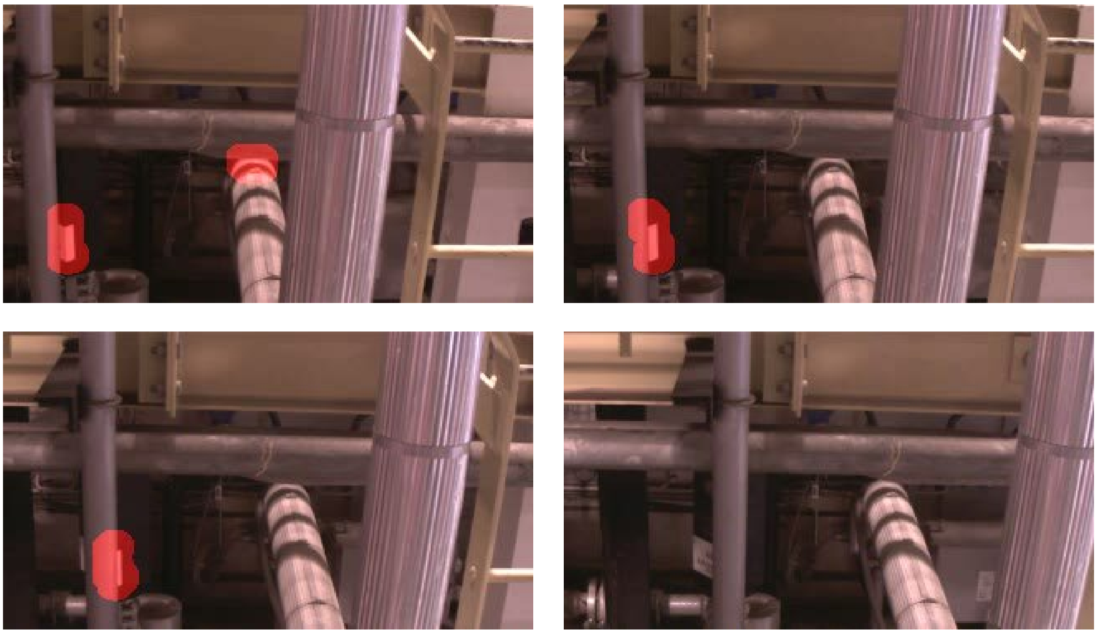


Figure 7.6: The whisky box was the abandoned object, but in some frames it is not detected as an anomaly. The points of the object detected as anomalous are painted in red.

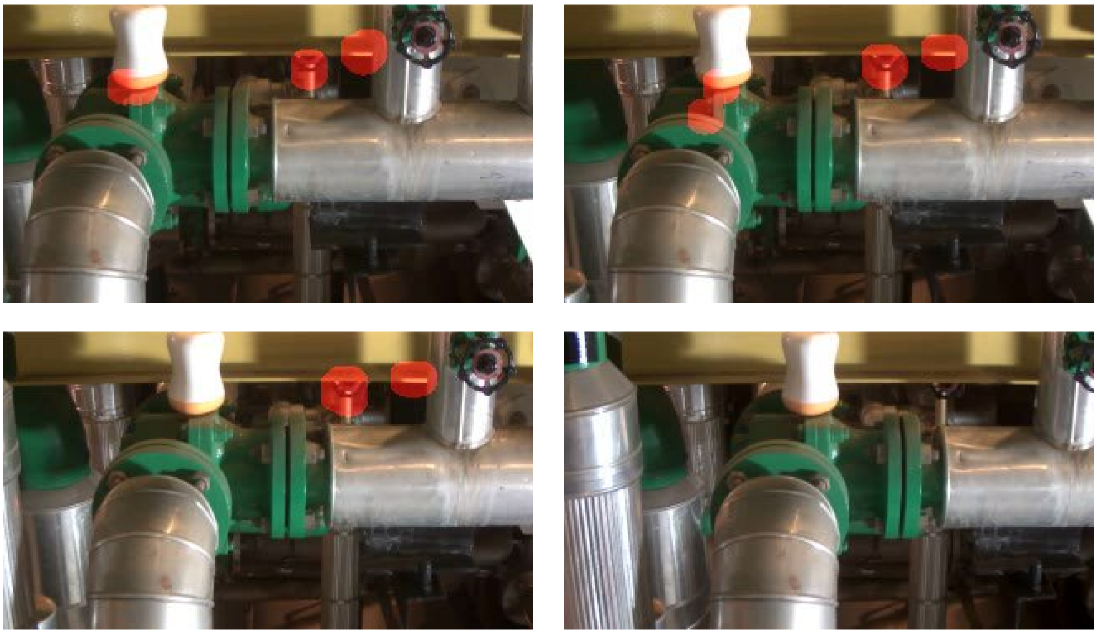
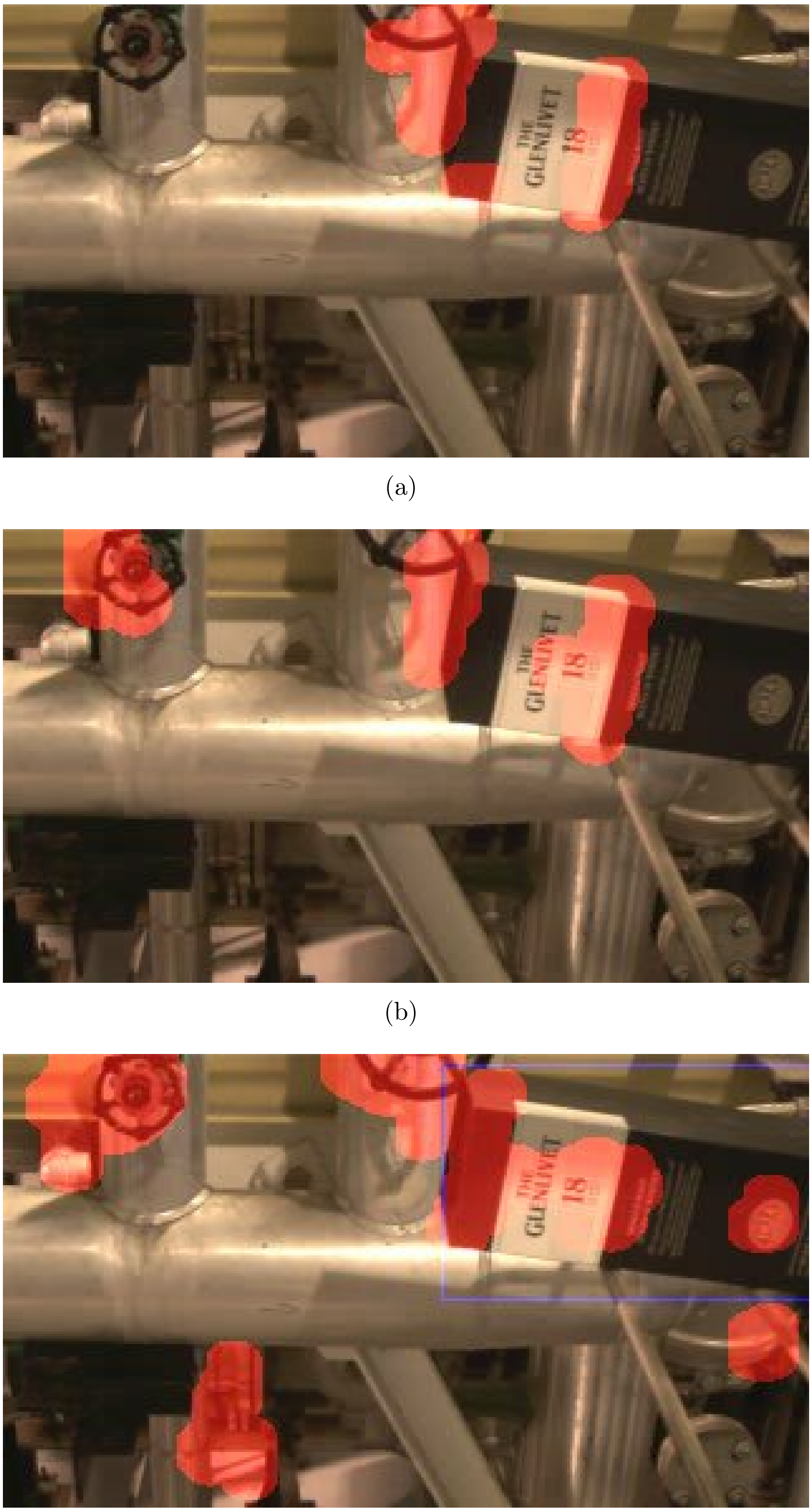


Figure 7.7: The bowl was the abandoned object, but in some frames it is not detected as an anomaly. The points of the object detected as anomalous are painted in red.

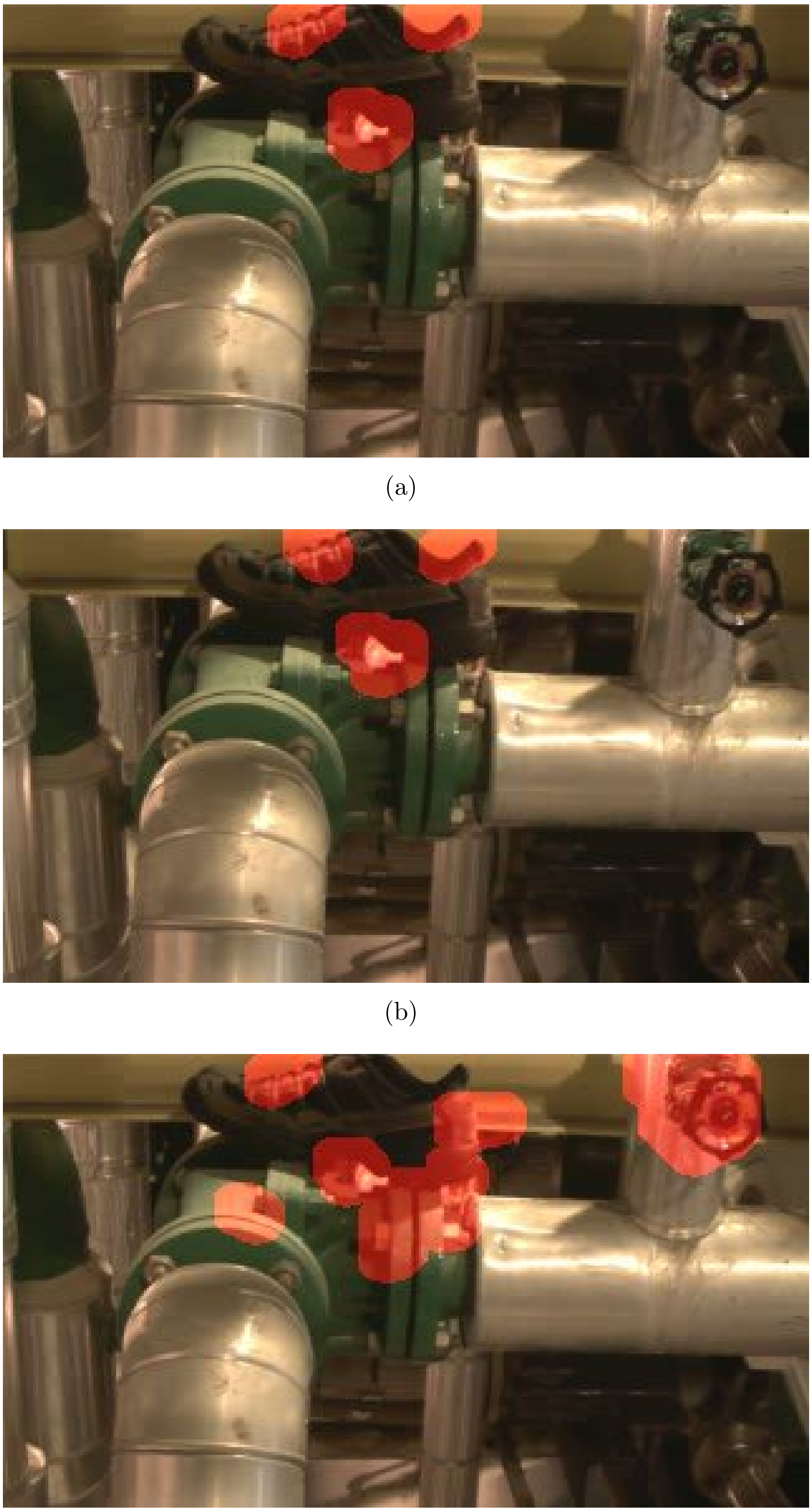


(a)

(b)

(c)

Figure 7.8: Example of the results obtained using: a) STC-mc, b) BOV and c) STC, when the abandoned object is a whisky box. In this case, the BOV and STC have more false positives, as shown in Table 7.1.



(a)

(b)

(c)

Figure 7.9: Example of the results obtained using: a) STC-mc, b) BOV and c) STC, when the abandoned object is a sneaker. In this case, the result is very similar, but the STC-mc has a little more false negatives, as shown in Table 7.1. STC has several false positives.

The STC-mc can also be configured to detect anomalous in the static camera case with a moving background. The results obtained with the parameters set $\{\lambda, \varepsilon_1, \sigma, \gamma, \varepsilon_2, \mu, \nu\}$ configured to $\{10, 1.3 \times 10^3, 0.4, 5 \times 10^{-12}, 600, 1.08, 7\}$ are shown in Fig. 7.10. Comparing these with the ones obtained with the original STC in Fig. 3.10 we can see that STC-mc has a performance as good as the one of the original STC in this case.

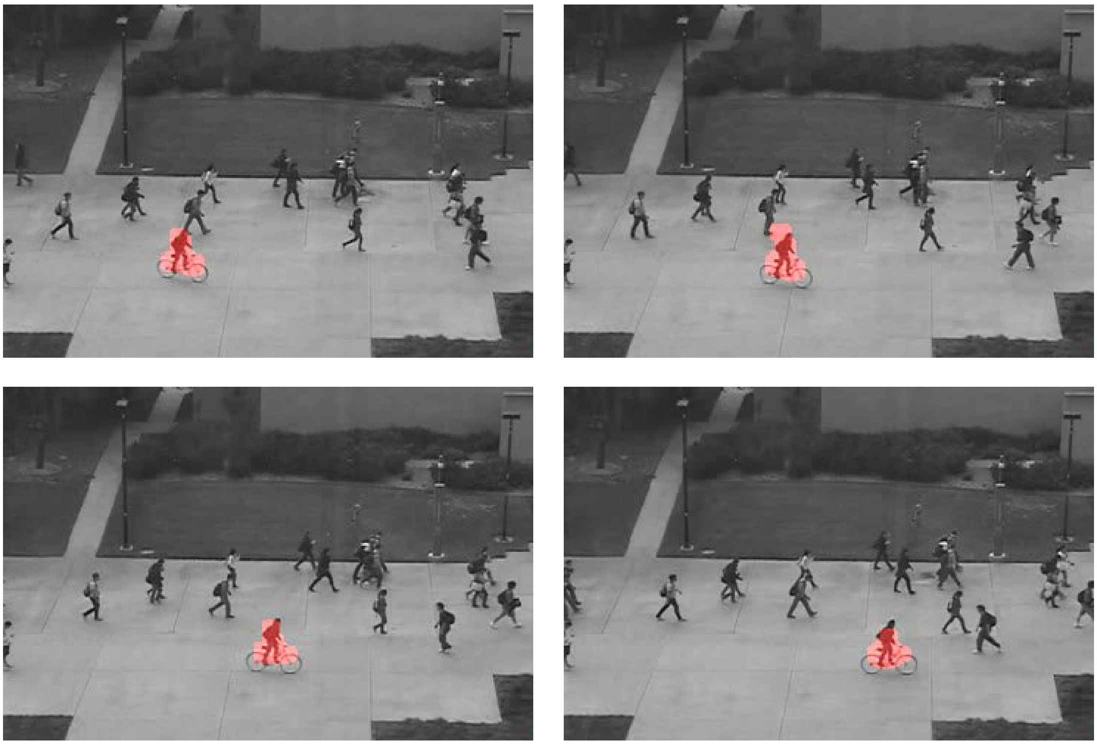


Figure 7.10: STC-mc applied to one video of the UCSD database. One can see that STC-mc is able to detect anomalous events in the case of a static camera with a moving background.

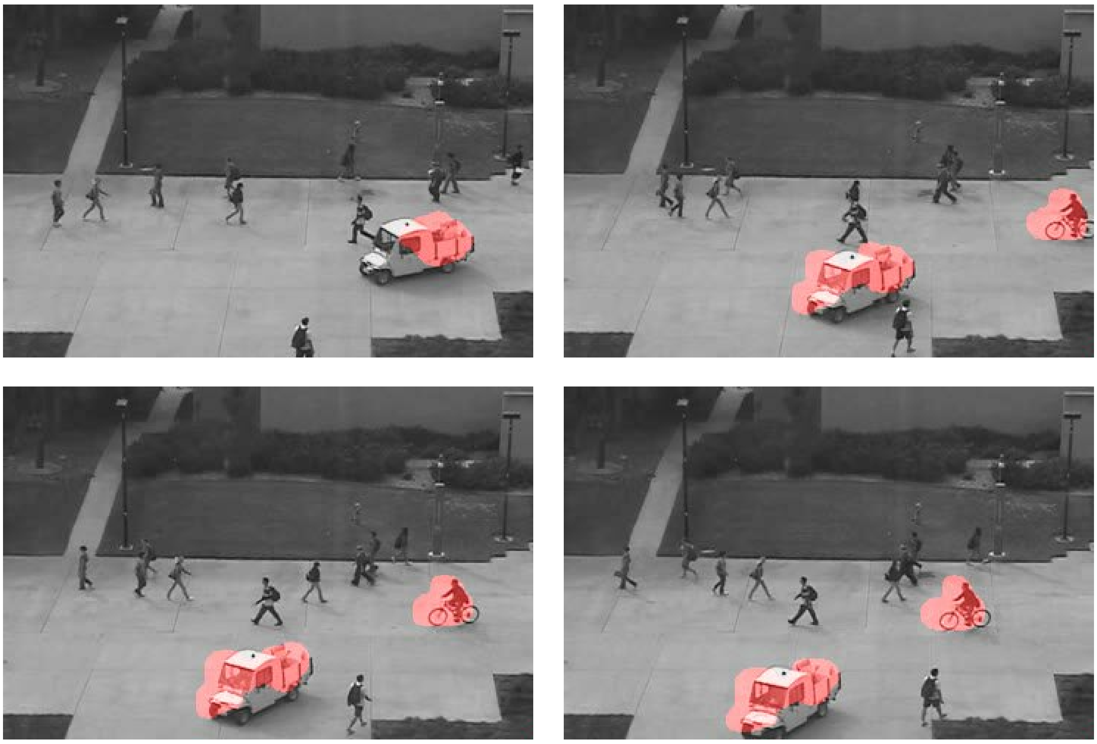


Figure 7.11: STC-mc applied to one video of the UCSD database. The cyclist and the cart are well detected.

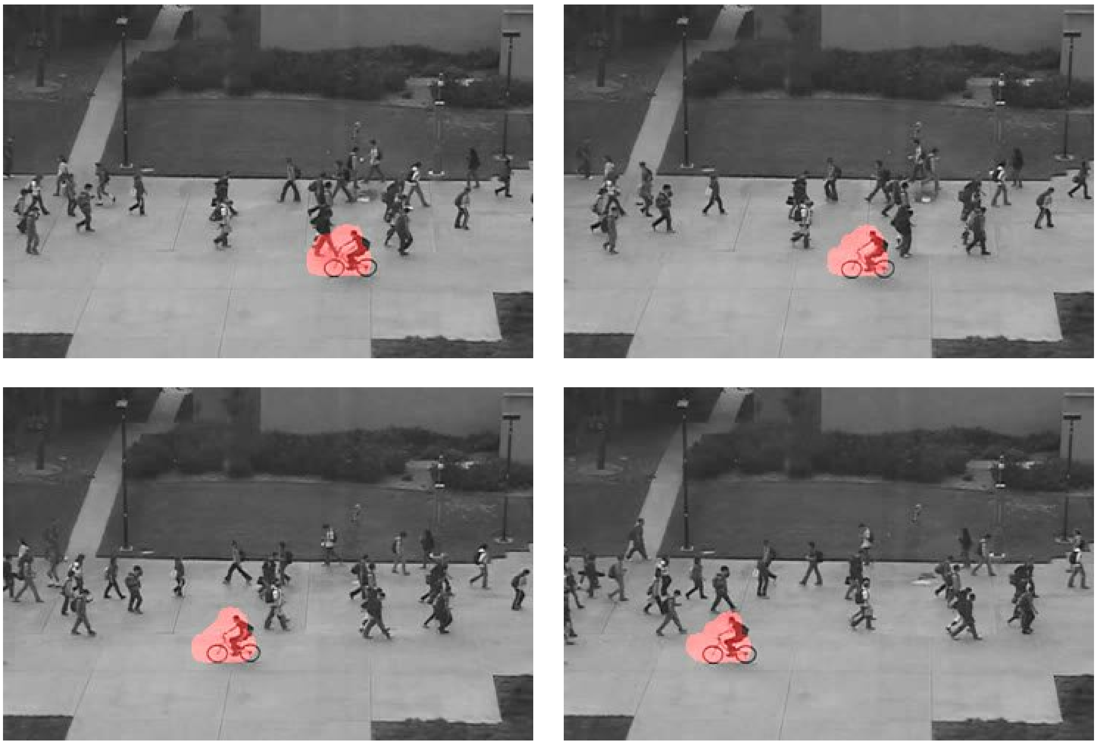


Figure 7.12: STC-mc applied to one video of the UCSD database. Only the cyclist is detected as an anomaly.

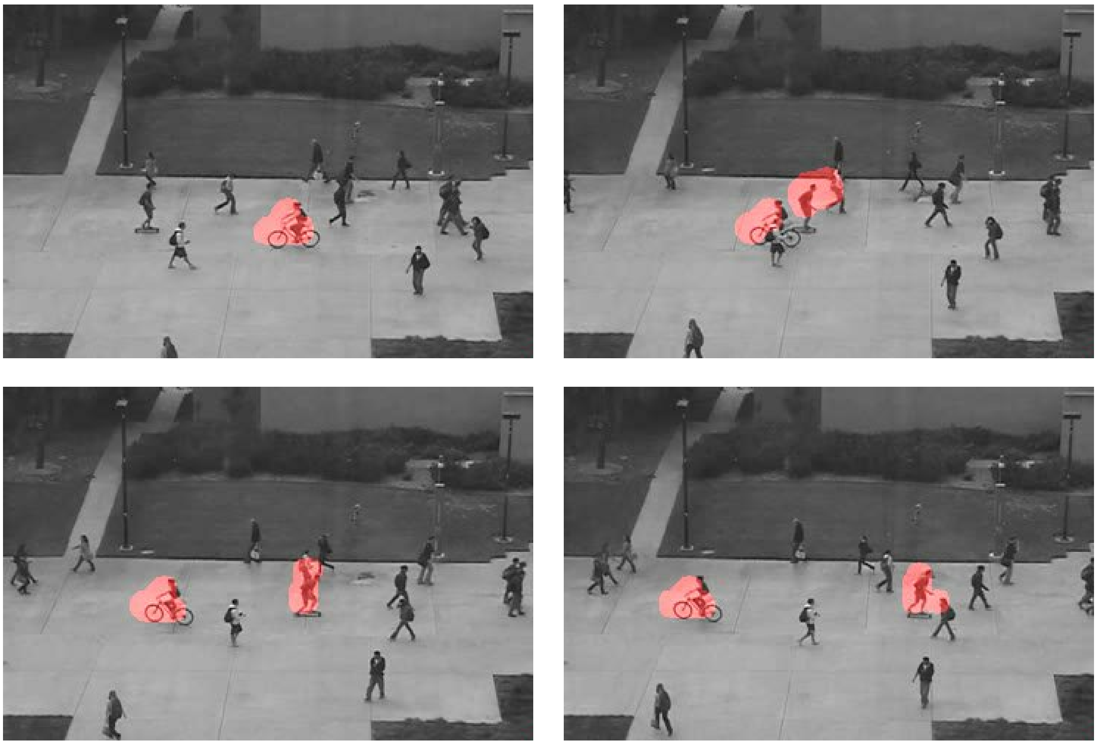


Figure 7.13: STC-mc applied to one video of the UCSD database. At the beginning, the skateboarder is very similar to a pedestrian, so he is not detected. But when he picks up speed he is detected too.

Chapter 8

Conclusions

This thesis has proposed the STC-mc algorithm, a new approach to detect abandoned objects in a cluttered environment, from videos obtained from a camera mounted on a moving platform. Although numerous works deal with the detection of abandoned objects, most of them are suitable only to the case of static cameras, and few of them have good results with a moving camera. The proposed STC-mc is based on the same principles as the STC method from [7], which uses dense sampling to break the video in small 3D volumes and calculates the probability of the spatio-temporal arranges of these volumes. In the STC a codebook is created to eliminate redundancies, representing similar volumes with the same codeword, and calculating the probability of occurrence of these codewords. The probability of the occurrence of the arrangements of the codewords in the image are calculated too. Therefore, the codebook is formed by the codewords and the probabilities of the arrangements. In the analysis of a video, the codebook is used to create a reconstruction of the target video replacing the volumes by the codewords from the codebook and calculating the probability of the arrangements of these codewords. Points with probability below a given threshold are considered as anomalous. However, the STC algorithm a good performance only when the video to be analyzed is obtained from a static camera.

Our STC-mc algorithm has enhancements that allow it to perform well in the case where the anomalies are abandoned objects and the video is obtained in a cluttered environment, a real industrial plant, using a camera mounted on a moving robot. This type of environment makes it very difficult to identify an abandoned object, because in the background there are objects with different shapes and sizes, and variations in the illumination. The robot movement also brings additional challenges, such as shaking and speed variations.

This work has three main contributions. The first and crucial contribution was the use of a second codebook that is generated by performing the training in two stages. The first codebook is used to determine what is common in the video, representing similar volume by a codeword from the codebook and the probability density function (pdf) of the arrangement of these videos volumes are calculated, as in the STC method. In the analysis, points with probability below a given threshold are candidates to be anomalous, but if the abandoned object in this video is similar to parts of the background that are very common, several false detections can occur. The introduction of the second codebook contributes to reduce the number of false detections significantly. It does so by containing codewords representing spatio-temporal compositions that have probabilities in the reference video that are lower than ones commonly associated to anomalies. Therefore, in the analysis phase, any point with a probability of occurrence below the threshold but with a descriptor similar to some codeword presents in the second codebook is discarded, not being considered as anomalous.

The second is the use of a new descriptor based on both spatial and temporal gradients. The use of only a temporal descriptor is not able to detect all the anomalies with a good performance, especially in the case of abandoned objects. Some anomalies are better detected when the spatial derivative is included, probably because the speed change does not affect the spatial derivative, since this depends only on points present in the same frame. The introduction of the spatial derivative in the descriptor has made the detection more stable, and improved the performance. However, only the spatial derivative has a poor performance too, because some differences in the training video and in the target video only can be detected by the parallax shift caused by the movement of the robot. So, the best detection result is obtained with the proposed spatio-temporal descriptor.

The third contribution is the introduction of a Gaussian filtering to deal with misalignments caused by camera shaking. Although the introduction of the spatial derivative in the descriptor helped to minimize the effect of the shaking, a more specific solution had to be developed. If the temporal derivative is too smoothed, some variations in the scene may be lost, and the identification of anomalies could be compromised. So the parameters tuning step was crucial to improve the performance of the method.

In the Chapter 6, the parameters used to configure the STC-mc algorithm were adjusted by performing several simulations, generating a cloud of points instead of a classic ROC curve. This was necessary because some parameters are correlated, and it was necessary to simulate several times, changing all the parameters, to determine the best set of parameters. To determine the best set of parameter for the first codebook, 27 combinations of parameters were simulated, with three different videos, totaling 81 simulations. For the second codebook, 49 combinations of parameters were simulated, with three different videos, totaling 147 simulations.

The proposed STC-mc algorithm was evaluated by processing seven videos of the

VDAO database, each of than with an abandoned object. In most of the cases the STC-mc algorithm was able to detect the abandoned objects in the VDAO database with low false positive and high true positive rates. It also managed to solve the false detections problem of the original STC algorithm. In fact, the STC-mc method had a good performance in five of the seven videos, with a TP rate greater than 90% and a FP rate lower than 13%. However, when the object to be detected was very similar to the background, the detection did not have a good performance. In two simulations where this occurred the TP rate was about 40% and 30%.

The STC-mc also performs better than a BOV (Bag of Video words) algorithm implementation. When compared to a BOV implementation, the STC-mc had better performance in four of the seven videos, and when the BOV approach was better, the difference was small, about 7%, and both the algorithm had a high TP rate and a low FP rate. A comparison with the STC method was also performed, but this method uses only the threshold to configure the algorithm, and was not possible to find a threshold where only TP were present in the frames. So, the STC-mc is better than the STC in all the seven videos.

Some visual tests were performed using the STC-mc in the case where the camera was static and several people were walking in the background. The STC-mc was able to detect anomalies like when a cyclist or a cart appears in the video, without generate false detections. So, the STC-mc performs as well as the STC method in the case of a static camera and a moving background.

In brief, although the performance of the STC-mc algorithm can be improved, especially in the case where the abandoned object is very similar to the background, the enhancements proposed to the STC method, that resulted in the STC-mc algorithm, succeeded in the identification of abandoned objects, without background subtraction, motion estimation or tracking. A prior knowledge of the type of event was not necessary, even in a cluttered environment.

8.1 Future Work

This thesis demonstrates that the STC-mc algorithm can be used to detect abandoned objects in a cluttered environment, from videos obtained from a moving camera. Besides that, it can also be used to detect anomalies in a tumultuous background, in the case of a static camera. However, several improvements can be made to have a better performance in the detection of anomalous events. Fig. 8.1 shows two situations when the STC-mc did not have a good performance. In many frames the objects were not detected, probably because they were confused with the background. Therefore, further works can be carried out to improve the performance of the STC-mc in these situations.

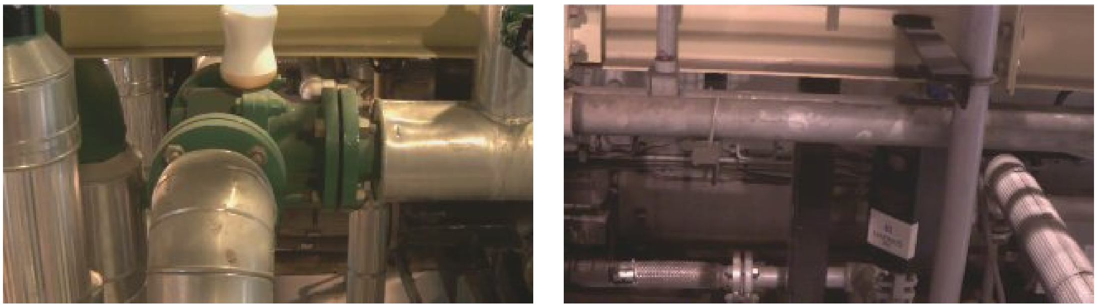


Figure 8.1: The STC-mc did not have a good performance detecting these abandoned objects. In many frames, the bowl and the whisky box were confused with the background.

Some attempts to improve the performance of the STC-mc can be made with the study of the performance of other descriptors. A more extensive study of the influence of the diverse types of descriptor can have a significant impact in the performance of the algorithm. Although our tests with one implementation of the HOG descriptor did not perform well, other types of implementations could have better results. Descriptor that use colors can also have good results in industrial plants, because it is common to use color to identify areas of risk and the colors of the pipe are used to indicate the kind of product that it is carrying. So, the use of the color can improve the identification of what is common in a scene. Besides that, the color of the uniform of the operators can help to identify when a person gets in an area of interest.

Although this thesis uses dense sampling to get the points of interest, based in the work of [38], the use of other interest points selection method can bring better results. Methods like SIFT or SURF can improve the performance and decrease the demand of memory and the computational complexity, once the number of interest point in the video tends to be smaller.

In the creation of the codebooks, a different algorithm can be used. The algorithm used in this work is very simple, and more sophisticated clustering algorithms can bring better results. For example, k-means or mutual information algorithms [57]. The expectation maximization (EM) algorithm can also be used to create these codebooks. A special care should be taken with the number of codewords in the codebook, because there is a trade-off between codebook size and time consumption. In this thesis, the codebook was split in pages, to speed up the search in the codebook. The drawback is that the frames of the reference and target video must be synchronized. An analysis could be performed to determine the best codebook size.

The use of a multi-resolution approach as the one proposed by [8] could bring improvements in detection, mainly when there are many objects of different sizes in the same video. Small video resolutions can be used to detect the bigger objects, and higher resolutions can be used to detect small objects. The multi-resolution scheme can reduce the computational complexity too. If the analysis of the video starts with the low resolution, and a object is detected in an area of the video, this area does not need to be analyzed with the other higher resolution. Furthermore, the size of the space-time volume may influence the accuracy of the STC-mc method.

The lighting can influence the detection capability of the STC-mc method. Especially when there is a great difference of lighting between the training video and the video to be analyzed. Therefore, a future work could perform tests to check how the STC-mc method is sensitive to lighting variations.

A further investigation could be performed to the case when in an environment like the observed in the VDAO database, there is a moving object, as a rotating machine, always present in the background. Possibly the STC-mc algorithm could learn that the rotating machine is part of the background and if an anomaly, like a person walking, happens in the scene, the system could detect only the person walking. In some industrial plants the presence of water vapor is common too, and the STC-mc can be trained to ignore this kind of steam, since this will always be present in the image and STC-mc has the characteristic of learn that a constant movement is part of the background.

As a final note, we should like to say that the main objective of this work was to prove the feasibility of the use of the STC approach to detect abandoned objects in cluttered environments using a moving camera. After some improvements, this objective was achieved, resulting in the STC-mc method. However, many other opportunities for the improvement of this method can exist, since it is a new application for this type of methodology, and this work did not intend to do a exhaustive search of the best solutions to all problems in the implementation of the STC-mc method.

Bibliography

[1] HAERING, N., VENETIANER, P. L., LIPTON, A. "The Evolution of Video Surveillance: An Overview", *Machine Vision and Applications*, , n. 5, pp. 279–290, June 2008.

[2] DEE, H. M., VELASTIN, S. A. "How Close Are We to Solving the Problem of Automated Visual Surveillance?" *Machine Vision and Applications*, v. 19, n. 5-6, pp. 329–343, June 2008.

[3] ADAM, A., RIVLIN, E., SHIMSHONI, I., et al. "Robust Real-time Unusual Event Detection Using Multiple Fixed-location Monitors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 30, n. 3, pp. 555–560, March 2008.

[4] BOIMAN, O., IRANI, M. "Detecting Irregularities in Images and in Video", *International Journal of Computer Vision*, v. 74, n. 1, pp. 17–31, January 2007.

[5] LAZEBNIK, S., SCHMID, C., PONCE, J. "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories", *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2169–2178, June 2006.

[6] SCHWARTZ, O., HSU, A., DAYAN, P. "Space and Time in Visual Context", *Nature Reviews Neuroscience*, v. 8, n. 7, pp. 522–535, November 2007.

[7] ROSHTKHARI, M. J., LEVINE, M. D. "An On-line, Real-Time Learning Method for Detecting Anomalies in Videos Using Spatio-Temporal Compositions", *Computer Vision and Image Understanding*, v. 117, n. 10, pp. 1436–1452, July 2013.

[8] CARVALHO, G. H. F. D. *Automatic Detection of Abandoned Objects With a Moving Camera Using Multiscale Video Analysis*. D.Sc. thesis, Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil, 2015.

[9] BAY, H., ESS, A., TUYTELAARS, T., et al. "Speeded-Up Robust Features (SURF)", *Computer Vision and Image Understanding*, v. 110, n. 3, pp. 346–359, 2008.

[10] FISCHLER, M. A., BOLLES, R. C. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", *Communications of the ACM*, v. 24, n. 6, pp. 381–395, June 1981.

[11] KONG, H., AUDIBERT, J., PONCE, J. "Detecting Abandoned Objects With a Moving Camera", *IEEE Transactions on Image Processing*, v. 19, n. 8, pp. 2201–2210, August 2010.

[12] SILVA, A. F., THOMAZ, L. A., CARVALHO, G. H. F., et al. "An Annotated Video Database for Abandoned-object Detection in a Cluttered Environment". In: *2014 International Telecommunications Symposium*, São Paulo, Brazil, August 2014.

[13] "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries", *IEEE Std 610*, pp. 1–217, Jan 1991.

[14] ANTONAKAKI, P., KOSMOPOULOS, D., PERANTONIS, S. "Detecting Abnormal Human Behaviour Using Multiple Cameras", *Signal Processing*, v. 89, n. 9, pp. 1723–1738, April 2009.

[15] YU, Y., ZHOU, C., HUANG, L., et al. "A Moving Target Detection Algorithm Based on the Dynamic Background". In: *International Conference on Computational Intelligence and Software Engineering*, pp. 1–5, Wuhan, China, December 2009.

[16] PEIJIANG, C. "Moving Object Detection Based on Background Extraction". In: *International Symposium on Computer Network and Multimedia Technology*, pp. 1–4, Wuhan, China, January 2009.

[17] LIPTON, A. J., FUJIYOSHI, H., PATIL, R. S. "Moving Target Classification and Tracking from Real-time Video". In: *Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision*, pp. 8–14, Princeton, USA, October 1998.

[18] ZHOU, D., ZHANG, H. "Modified GMM Background Modeling and Optical Flow for Detection of Moving Objects". In: *IEEE International Conference on Systems, Man and Cybernetics*, pp. 2224–2229, Waikoloa, USA, October 2005.

[19] FELZENSZWALB, P. F., GIRSHICK, R. B., MCALLESTER, D., et al. "Object Detection with Discriminatively Trained Part-Based Models", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 32, n. 9, pp. 1627–1645, Sept 2010.

[20] KIM, C., HWANG, J. N. "Fast and Automatic Video Object Segmentation and Tracking for Content-Based Applications", *IEEE Transactions on Circuits and Systems for Video Technology*, v. 12, n. 2, pp. 122–129, February 2002.

[21] POPOOLA, O., WANG, K. "Video-Based Abnormal Human Behavior Recognition – A Review", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, v. 42, n. 6, pp. 1–14, November 2012.

[22] RIDDER, C., MUNKELT, O., KIRCHNER, H. "Adaptive Background Estimation and Foreground Detection using Kalman-Filtering". In: *Proceedings of Intenational Conference on recent Advances in Mechatronics*, pp. 193–199, 1995.

[23] ARULAMPALAM, M. S., MASKELL, S., GORDON, N., et al. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Transactions on Signal Processing*, v. 50, n. 2, pp. 174–188, February 2002.

[24] YANG, H., SHAO, L., ZHENG, F., et al. "Recent Advances and Trends in Visual Tracking: A Review", *Neurocomputing*, v. 74, n. 18, pp. 3823–3831, November 2011.

[25] SENIOR, A. "Tracking People with Probabilistic Appearance Models". In: *IEEE workshop on Performance Evaluation of Tracking and Surveillance*, p. 48–55, June 2002.

[26] HARD, M., MACCORMICK, J. "BraMBLe: a Bayesian Multiple-blob Tracker". In: *Proceedings of International Conference on Computer Vision*, p. 34–41, 2001.

[27] ZHANG, J., CHEN, C. H. "Moving Objects Detection and Segmentation In Dynamic Video Backgrounds". In: *IEEE Conference on Technologies for Homeland Security*, pp. 64–69, Woburn, USA, June 2007.

[28] SCHOLKOPF, B., PLATT, J. C., SHAWE-TAYLOR, J., et al. "Estimating the Support of a High-dimensional Distribution", *Neural Computation*, v. 13, n. 7, pp. 1443–1471, June 2001.

[29] HEIJDEN, F. V. D., DUIN, R. P. W., DE RIDDER, D., et al. *Classification, Parameter Estimation and State Estimation.* 1st ed. West Sussex, UK, Wiley, 2004.

[30] REMAGNINO, P., TAN, T., BAKER, K. "Agent Orientated annotation in Model Based Visual Surveillance". In: *Proceedings of International Conference on Computer Vision*, p. 857–862, Bombay, India, 1998.

[31] BILMES, J. *A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models.* Relatório técnico, International Computer Science Institute and Computer Science Division, University of California at Berkeley, 1998.

[32] CAVALLARO, A., MAGGIO, E. *Video Tracking : Theory and Practice.* 2nd ed. Vicon, UK, Wiley, 2011.

[33] KRATZ, L., NISHINO, K. "Anomaly detection in Extremely Crowded Scenes Using Spatio-Temporal Motion Pattern Models". In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1975—-1981, Miami, USA, June 2009.

[34] MAHADEVAN, V., WEIXIN, L., BHALODIA, V., et al. "Anomaly Detection in Crowded Scenes". In: *2010 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1975–1981, San Francisco, USA, June 2010.

[35] BERTINI, M., DEL BIMBO, A., SEIDENARI, L. "Multi-Scale and Real-Time Non-Parametric Approach for Anomaly Detection and Localization", *Computer Vision and Image Understanding*, v. 116, n. 3, pp. 320–829, March 2012.

[36] KLASER, A., MARSZAŁEK, M., SCHMID, C. "A Spatio-Temporal Descriptor Based on 3d-Gradients". In: *BMVC 2008-19th British Machine Vision Conference*, pp. 1–10, September 2008.

[37] ADAM, A., RIVLIN, E., SHIMSHONI, I., et al. "Robust Real-Time Unusual Event Detection Using multiple Fixed-location Monitors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 30, n. 3, pp. 555–560, March 2008.

[38] WANG, H., ULLAH, M. M., KLASER, A., et al. "Evaluation of Local Spatio-temporal Features for Action Recognition". In: *British Machine Vision Conference*, pp. 124.1–124.11, London, UK, September 2009.

[39] TOMIOKA, Y., TAKARA, A., KITAZAWA, H. "Generation of an Optimum Patrol Course for Mobile Surveillance Camera", *IEEE Transactions on Circuits and Systems for Video Technology*, v. 22, n. 2, pp. 216–224, February 2012.

[40] SUHR, J. K., JUNG, H. G., LI, G., et al. "Background Compensation for Pan-Tilt-Zoom Cameras Using 1-D Feature Matching and Outlier Rejection", *IEEE Transactions on Circuits and Systems for Video Technology*, v. 21, n. 3, pp. 371–377, March 2011.

[41] DAVIS, J. W., MORISON, A. M., WOODS, D. D. "An Adaptive Focus-of-Attention Model for Video Surveillance and Monitoring", *Machine Vision and Applications*, v. 18, n. 1, pp. 41–64, October 2007.

[42] BRADSKI, G. R., DAVIS, J. W. "Motion Segmentation and Pose Recognition with Motion History Gradients", *Machine Vision and Applications*, v. 13, n. 3, pp. 174–184, August 2001.

[43] WU, X., GONG, H., CHEN, P., et al. "Surveillance Robot utilizing video and audio information", *Journal of Intelligent & Robotic Systems*, v. 55, n. 4-5, pp. 403–421, August 2009.

[44] ZHOU, D., WANG, L., CAI, X., et al. "Detection of Moving Targets with a Moving Camera". In: *IEEE International Conference on Robotics and Biomimetics*, pp. 677–681, Guilin, China, December 2009.

[45] LOWE, D. G. "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, v. 60, n. 2, pp. 91–110, November 2004.

[46] VDAO. "VDAO - Video Database of Abandoned Objects in a Cluttered Industrial Environment". [Online], 2016. Available at `http://www.smt.ufrj.br/~tvdigital/database/objects`.

[47] RAPANTZIKOS, K., AVRITHIS, Y., KOLLIAS, S. "Dense Saliency-Based Spatiotemporal Feature Points for Action Recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1454–1461, Miami, USA, June 2009.

[48] ZHONG, H., SHI, J., VISONTAI, H. "Detecting Unusual Activity in Video". In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 819–826, Washington, USA, June 2004.

[49] UCSD. “UCSD Anomaly Detection Dataset”. [Online], 2014. Available at `http://www.svcl.ucsd.edu/projects/anomaly`.

[50] QT. “QT Project”. [Online], 2014. Available at `http://www.qt-project.org`.

[51] DEITEL, P. J., DEITEL, H. M. *C++ Como Programar*. 3rd ed. Porto Alegre, Brazil, Bookman, 2001.

[52] OPENCV. “Opencv Library”. [Online], 2014. Available at `http://www.opencv.org`.

[53] LAGANIÈRE, R. *OpenCV 2 Computer Vision Application Programming Cookbook*. Birmingham UK, Packt, 2011.

[54] FAWCETT, T. “An introduction to ROC analysis”, *Pattern Recognition Letters*, v. 27, n. 8, pp. 861–874, June 2006.

[55] GONZALEZ, R. C., WOODS, R. E. *Digital Image Processing*. 3 ed. New Jersey, Pearson Prentice Hall, 2008.

[56] “IRobot Roomba Vacuum Cleaning Robot”. [Online]. `http://www.irobot.com/For-the-Home/Vacuum-Cleaning/Roomba.aspx`.

[57] LIU, J., SHAH, M. “Learning Human Actions Via Information Maximization”. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008.