UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

Background initialization for the task of video-surveillance

-TRABAJO FIN DE MÁSTER-

Diego Ortego Hernández
Septiembre 2014

Background initialization for the task of video-surveillance

Autor: Diego Ortego Hernández
Tutor: José María Martínez Sánchez
email: {diego.ortego@uam.es, joseM.Martínez@uam.es}

Video Processing and Understanding Lab
Departamento de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Septiembre 2014

Trabajo parcialmente financiado por el gobierno español bajo el proyecto TEC2011-25995 (EventVideo)

Abstract

In this work, we propose a region-wise and batch processing approach for background initialization in video-surveillance based on a spatio-temporal analysis.

First, the related work has been explored. Then, the efforts are focused on developing a new background initialization approach to outperform the literature performance. To this end, a temporal analysis and a spatial analysis are performed. In the first stage, we use a previous work techniques adding motion information to increase performance. In the second stage, a multipath iterative reconstruction scheme is performed to build the true background under the assumption of background smoothness, i.e. the empty scene is smoother than the scene with foreground regions.

Finally, the results over challenging video-surveillance sequences show the quality of the proposed approach against related work.

Keywords

Background subtraction, background initialization, stationary foreground, background visibility, dimensionality reduction, motion, clustering, smoothness, multipath.

Acknowledgements

En primer lugar, me gustaría agradecer a Chema su gran ayuda (ideas, charlas constructivas, consejos...) para realizar este trabajo a lo largo del último año, así como su comprensión en cuanto a cómo compaginar el trabajo con las asignaturas del máster que en ciertos momentos, y en contra de mis deseos, me quitaron mucho tiempo.

En segundo lugar, pero no por ello menos importante, agradecer a Juan Carlos sus ideas y consejos pues nuestro trabajo es también suyo.

También quiero agradecer a todos los compañeros del VPU (Álvaro, Luis, Rafa, Pencho y en especial a Marcos al que he dado mucho "la brasa") su gran ayuda y disponibilidad a ayudarme en todo momento. He aprendido mucho este año.

Quiero volver a mencionar, al igual que en el PFC, a mi primo Carlos que tiene gran culpa del camino que he seguido.

Acordarme también de mi familia que siempre me apoya en todo lo que hago.

Gracias a todos.

Diego Ortego Hernández

Septiembre 2014

Índice general

Índice de figuras

Índice de tablas

Chapter 1

Introduction

1.1 Motivation

Nowadays, there is a growing demand for automated surveillance systems due to heightened security concerns. Technological advances and reduced costs have led to an accelerated deployment of surveillance cameras, both in public and private facilities. These environments need to be monitored automatically by video-surveillance algorithms in order to help human security operators to identify potential threats and act accordingly. To this end, the first step in many applications is to perform a segmentation between relevant moving objects and irrelevant objects. This task is usually carried out by a Background Subtraction (BS) algorithm [6] that yields a binary mask of relevant objects —foreground (FG)— that is obtained from a comparison between the incoming frame and a model of the irrelevant objects or empty scene —background (BG) model— thus providing an essential point of departure for many video-surveillance applications —event and activity detection, object tracking, people detection— required in public and private facilities.

According to [6], a BS algorithm could be defined by the strategy followed to address four different tasks (see Figure 1.1): Background Initialization —defines the strategies to initialize the model with a True Background (TB) image free of foreground objects thus determining an appropriate point of departure for the background modelling stage— , Background Modelling —describes the nature of the model and associated statistics used to store the empty scene—, Background Maintenance —determines the procedure to adapt the model to the scene variations over time— and Foreground Detection —measures the difference between incoming frames and the BG model according to a set of features—. This work is focused in the Background Initialization (BGI) stage.

The BGI stage has been weakly investigated in comparison with the remainder stages [6]. It consists in initializing the background model computing a TB image free of foreground objects from a training sequence (see Figure 1.2). Several BS approaches in the state-of-the-art use

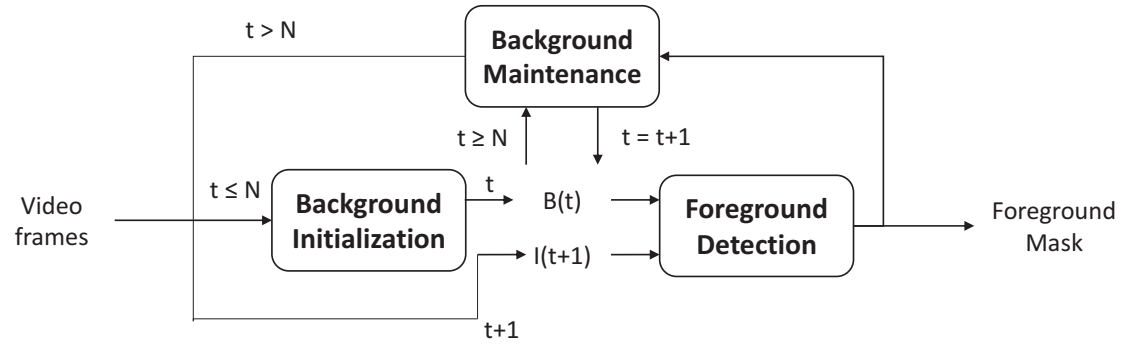

Figure 1.1: Background Subtraction process. N is the number of frames that is used for the background initialization. B(t) and I(t) are the background and the current image at time t, respectively. This image is copied from [6].

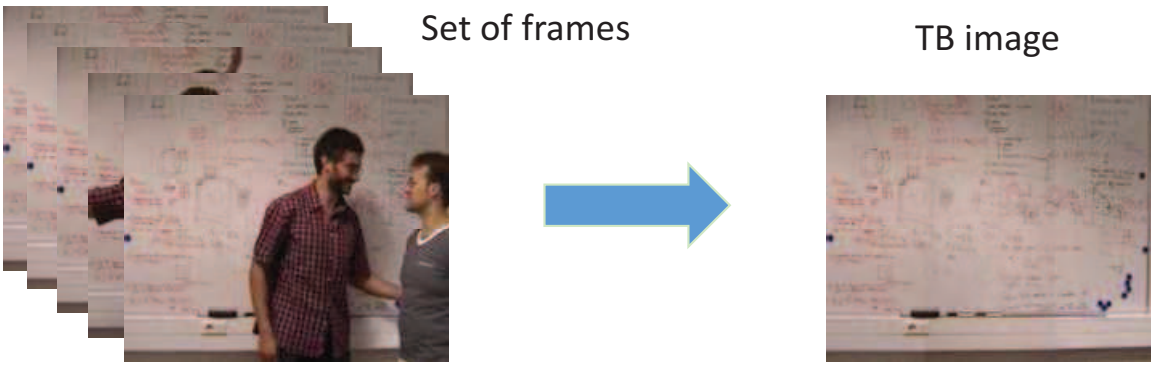

Figure 1.2: Background initialization idea. The TB of the empty scene is obtained from a set of frames —training sequence—.

an unreliable scheme to initialize the model, based on the assumption that the TB could be easily captured from the first frames of the sequence. This assumption is incorrect in many video-surveillance scenarios where there may be many foreground objects due to crowds and stationary objects. Therefore, capturing the TB in these situations is not an easy task and BGI is a suitable way to tackle it. Furthermore, BGI could be very useful to deal with high illumination changes due to the implicit capability to estimate a new background image and re-initialize the background model with it. As mentioned in [7], a BGI algorithm could be used to deal with the invisibility and ghost issues[1].

[1]Problem that arise when an object stops and becomes part of the BG model and subsequently moves again, bringing out a —ghost— false detection.

1.2 Objectives

The main objective of this Master Thesis is to develop an algorithm for the initialization of a background model improving performance achieved in the literature given a set of training frames. After a comprehensive study of the literature, including the algorithm available at VPU Lab, solutions to current issues will be proposed to increase the performance of previous approaches. The following sub-goals are defined:

- Comprehension of the key challenges of BGI: the study of the literature should provide a broad view of the current solutions and issues of the BGI task.
- Compilation of a video dataset for BGI: this set should be extensive and should cover the challenging problems studied in the revision of the literature.
- Implementation of BGI algorithm: the point of departure will be the previous approach available at VPU Lab and the enhancements will be related to a reconstruction based on spatial continuity information.
- Performance evaluation: the proposed dataset will hold a wide variety of situations to test the proposed approach in comparison with state-of-the-art approaches. Improvements will be explored taking into account the issues found.

1.3 Document structure

This document is structured as follows:

- Chapter 1: This chapter presents the motivation and objectives of this work.
- Chapter 2: This chapter describes the state-of-the-art in background initialization.
- Chapter 3: This chapter describes the developed algorithm.
- Chapter 4: This chapter presents the achieved results.
- Chapter 5: This chapter gives some conclusions and future ideas for the tasks addressed in this Master Thesis.

Chapter 2

State-of-the-art

2.1 Introduction

The aim of Background Initialization (BGI) is to yield the true —static— background (TB) of the empty scene given a set of training frames in which the background is occluded by foreground objects. As mentioned in chapter 1, BGI is different from the Background Modelling task, where the target is to deal with the challenging phenomenon of dynamic background (waving trees, moving escalators, etc) in order to supply to the Foreground Detection stage enough information to avoid false motion detections. Furthermore, a Background Modelling strategy together with a Background Maintenance scheme could be exploited to provide a TB of the scene [8]. BGI [2][5][6][9] is usually also referred as Bootstrapping [6][10][11], Background estimation [3][12], Background generation [7][13] or Background reconstruction [14].

BGI approaches in the literature rely on different types of data to build the mentioned TB. This data could be collected into two classes, temporal and spatial. Temporal information used could be regarded as different types of intensity variations information —optical flow [1][15], temporal stability [1][15][16], temporal smoothness [5]—. On the other hand, spatial information tries to build the TB relying on a extended idea: the spatial smoothness of the TB [2][3][4][7][9][10][12][14][15][17], thus minimizing the "transitions" energy of the selected solution with an optimal labelling scheme [4][5][12][18] or with an iterative completion [2][3][9][10][17] scheme. Temporal and spatial information could be used in a batch or an online fashion, operating at different levels, namely pixel-wise and region-wise.

The BGI task has been weakly investigated [6][8], nevertheless it is a rich source of information for video surveillance —TB is a key data for BS—, video segmentation —TB is useful to extract foreground objects—, video compression —TB represents the redundant information—, video inpainting —TB provides a solution to the hole completion of regions selected by the user—, privacy protection for videos —the target is to avoid the infringement on the privacy of people in videos mainly in sharing services— and computational photography —TB is target

image free of foreground— [8].

In the state-of-the-art is commonly used the assumption that the initial frames of the video sequence are free of foreground objects —occluders, clutter or outliers— thus being easy to obtain the TB in that instants [8]. However, this premise is erroneous in many scenarios —shopping malls, airports or train stations— where crowds are the norm, leading to an erroneous TB, thus impacting negatively in the applications performance. Therefore, capturing the TB is a key task, being the inclusion of a proper BGI stage necessary.

2.2 Classification of BGI approaches

Several paths have been explored to tackle the BGI task, however neither of them propose a sufficiently general and accurate scheme due to the initial assumptions taken into account. Among these assumptions, the one which leads to better performance results is the "Background Smoothness" in comparison with the foreground. However, this assumption is not always true and this fact should not be forgotten. Strategies followed to initialize the background have been recently compiled in the book chapter [8], in which the authors note four categories of BGI approaches:

- Methods based on temporal statistics (TS): The median as scheme to initialize the background is used in several publications [8][19], having the well-known disadvantage of considering as background every foreground object being stationary for more than the 50% of the training sequence. This category also groups the combination of Background Modelling and Background Maintenance of many Background Subtraction (BS) approaches, that can be exploited to provide a background image as in [11]. These algorithms used in BS usually have the underlying assumption of considering the most repeated representation of a location in the scene as background, which leads to numerous errors when stationary foreground objects come into play. In [7], the initial frame is used as an initial TB that is updated online for each frame depending on non-significant —system noise, gradual lighting changes and repeated background movements— or significant —sudden illumination changes, sudden background movements and foreground intrusions— variations and taking into account in the latter spatial information to promote smoothness in the TB and tackle stationary objects and ghost phenomenon.

- Methods based on subintervals of stable intensity (SSI): These approaches consist of two stages, first spatial locations are divided into temporal —and continuous— stable intervals and subsequently a technique to determine the interval representing the TB is applied (see Figure 2.1). The algorithm presented in [16] follows this scheme, computing the stable intervals at pixel level and subsequently selecting the background stable interval using the number of samples in the interval and their variance. In [1], the authors improve

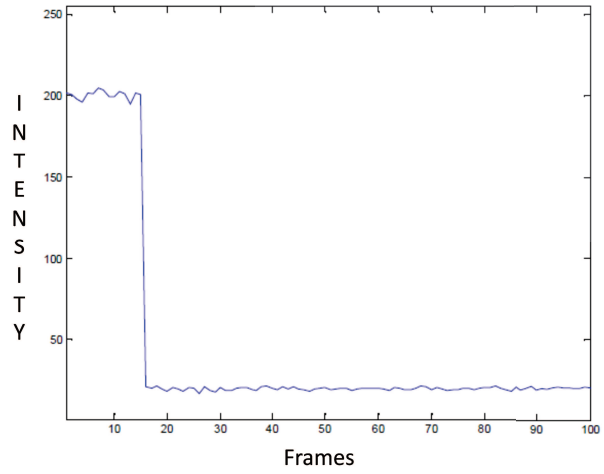

Figure 2.1: The figure represents two stable intensity intervals for a pixel, where one of them describes the TB. SSI approaches locate these intervals and subsequently identified which of them models the TB.

the algorithm Local Image Flow [15] —which in turn is motivated by [20]— equalizing the effect of objects moving at different depths thus improving the background likelihoods —computed based on stable intervals and optical flow— used to define the TB. Nevertheless, these approaches based on temporal statistics usually fail in presence of stationary objects, because they become part of the stationary scene and they will be confused with the stationary TB.

- Methods based on Iterative Model Completion (IMC): These approaches operate filling the TB spatial locations iteratively (see Figure 2.2) by selecting in each iteration a new candidate of the background to fill a hole. Before the iterative completion, usually a clustering technique is applied in each spatial location to narrow down candidates by grouping them into clusters that could include blocks from non-consecutive temporal instants (see Figure 2.2) unlike stable intervals approaches. In [17], images are processed at block level of size 16×16 and blocks in every spatial location are reduced by a clustering technique. Then the background is initialized with one or more blocks as previous step of the iterative completion based on coherence or smoothness —measured by a DCT-scheme— between one empty location and its neighbours. Algorithm presented in [9] has a scheme similar to [17], however the authors replace the DCT by the Hadamard transform —as measure of best continuity— which is faster than DCT. Furthermore, a correction of the candidate selected in each iteration is applied when the gradient in the edges between the selected candidate and its neighbours is high. In [2], an overlapping block scheme is applied and

Figure 2.2: From left to right, the iterative reconstruction process can be appreciated, where the TB is initialized and iteratively filled. Image is extracted from [3].

blocks along the training sequence are clustered by a single linkage agglomerative clustering approach. The iterative completion is based on a spatial continuity whose measure relies on a chi-square index and colour difference in overlapping and non-overlapping areas respectively between the candidate and the already fixed background locations. Recently, an evolution of [17] by the original authors was presented [3], improving the iterative completion by modelling this task as a Markov Random Fields (MRF) problem in which the energy —measured by a DCT-scheme more complete than [17]— is minimized. In [10], the background is iteratively built at block level in an online fashion —thus allowing updating— by filling the empty TB locations when there is no motion in a block and it is also classified as static.

- Methods based on optimal labelling (OL): These algorithms reconstruct the TB by determining the best label —frame that represents the background— in each location via energy —which usually has a data term and a smoothness term— minimization. Once the optimal labelling is performed, the background is obtained by Image Domain Composition (IDC) or Gradient Domain Composition (GDC). The differences between methods consist of different energy functions, optimal labelling algorithms —Loopy Belief Propagation (LBP) and Graph cuts (GC)— and the already mentioned scheme for the TB reconstruction using the optimal labelling computed. In [4], under the assumption of the background smoothness condition, an energy —comprising a first term to measure smoothness and a second term to speed-up the minimization— minimization approach using LBP and GDC is presented. In [12], an energy —comprising a pixel colour term, a predicted spatial term from stable regions and a smoothness term— is minimized by GC. In [14], the authors solve the BGI task together with motion detection —BS— by optimizing an energy —discriminative term for penalizing or favouring the presence of motion at a point, the reconstruction term involved in either in the estimation of the background intensity and in the

Reference	Operation	Spatial-level	Concepts managed
[1]	Batch	Pixel	SSI, Optical flow, Motion at different depths
[2]	Batch	Region	IMC, Smoothness
[3]	Batch	Region	IMC, Smoothness
[4]	Batch	Hybrid	OL, Smoothness
[5]	Online	Hybrid	OL, Smoothness
[7]	Online	Region	Types of intensity variations, Smoothness
[9]	Batch	Region	IMC, Smoothness
[10]	Online	Region	IMC, motion information, block correlations
[11]	Online	Hybrid	TS, BF from BS approach
[12]	Batch	Hybrid	OL, Smoothness, Inpainting
[14]	Online	Hybrid	Smoothness, Motion Information
[15]	Batch	Hybrid	SSI, Optical Flow, Smoothness
[16]	Batch	Pixel	SSI
[17]	Batch	Region	IMC, Smoothness
[18]	Batch	Hybrid	OL, avoid over-smoothing
[19]	Batch	Pixel	TS, Temporal Median

Table 2.1: Details of some relevant state-of-the-art approaches. The operation of the algorithm with the training sequence, the spatial-level used and some general concepts of the approaches are shown.

BS task and the regularization terms related with the smoothing— under the framework of Conditional Mixed-State MRF, where spatio-temporal interactions between symbolic —BS— and numeric —BGI— processes are taken into account and modelled together. The approach proposed in [18], presents a technique to deal with non-smooth backgrounds —over-smoothing problem— by obtaining the local optimal paths for the message-passing in LBP by correlation matching —local maximum weight matching— and computing the information of pixels; then the pixels with minimum information are selected as background. Recently in [5], where background adaptation is seen as a unified problem with background initialization and stationary object detection, they reconstruct the TB under an energy minimization —pixel-wise and region-wise minimization with spatio-temporal reliabilities between sites— scheme in a dynamic MRF framework.

A brief of the properties of the most relevant approaches in the literature is presented in Table 2.1, showing three fields to describe the approach: Operation —meaning if the algorithm operates in a batch or an online fashion—, Spatial-level —meaning if the method uses a pixel-wise or region-wise reconstruction— and Concepts managed —alluding the strategies followed to perform the BGI task—.

Figure 2.3: Example of objects moving at different depths. The figure shows that different foreground resolutions caused the asymmetric distribution of optical flow. This image is extracted from [1].

2.3 Selected BGI algorithms

This section deepens some interesting approaches proposed in the literature in order to summarize in detail the different strategies developed specifically to tackle de BGI task. These methods are selected due to be —in our opinion— the most interesting and recent approaches of each category presented in section 2.2.

2.3.1 SSI and motion at different depths for BGI [1]

In the batch processing and pixel level approach proposed in [1], stable intervals together with a technique to equalize the effect of objects moving at different depths is applied. The assumption made is that the TB will be disclosed for at least a short period of time in each pixel location.

First stage in this algorithm is to compute non-overlapping and continuous stable intensity intervals —the lowest interval length is set to 5—, simply by requiring a variation under a fixed threshold in each computed interval. Next step is to compute the optical flow between each pair of frames in the training sequence, for which the search radius is selected depending on the foreground activity, i.e. depending on the maximum foreground movement at different scales. Once the optical flow is computed, there are many flow vectors with their heads and tails. These heads and tails in a pixel neighbourhood provide useful information to know if a pixel location has to increase or reduce its background likelihood, according to the idea that:

- Head means an object covering the pixel.
- Tail means an object uncovering the TB.

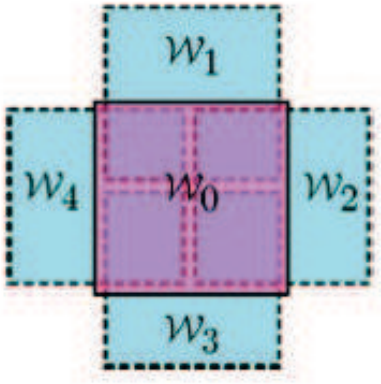

Figure 2.4: Example of overlapping block. Each block W_0 has four different fixed neighbours —W_1, W_2, W_3 and W_4—. The purple area is where Seamlessness condition —W_0 has to be similar to the overlapping part of fixed neighbours— is check it. Moreover, the blue areas represent the locations where best continuation is measured. This image is extracted from [2].

Unlike [15], local contribution of heads and tails to a pixel locations are weighted in order to equalize the effect of objects moving at different depths (see Figure 2.3) that induce uneven flow vectors, i.e. affects the local flow vector density. The main drawback of this approach is the lack of any technique to tackle the problem of stationary objects.

2.3.2 Overlapping block-based BGI with an IMC scheme [2]

In [2], a block level and batch processing approach minimizing colour transitions and taking into account overlapping similarity by an IMC scheme is proposed. It introduces one assumption that should not be forgotten: TB is revealed at least once in each block location along the training sequence.

The first stage in this method is to divide the image frame in overlapping blocks of size NxN (see Figure 2.4). Subsequently, an agglomerative clustering approach is applied to merge similar temporal blocks in each spatial location when the Sum of Square Distances (SSD) between two blocks —computed taking into account the noise in the scene— is lower than a threshold. It is important to mention that clusters with one member are rejected as it is considered that they represent blocks with motion and also that the block representing the cluster is computed averaging the pixels of the blocks. The last stage is the TB reconstruction in an IMC scheme taking into account the computed clusters —which are the candidates in each location— and choosing the best continuation candidate in one location in each iteration. This process begins by initializing the TB with the cluster(s) that compiles more blocks. Subsequently, in each iteration a new TB block is set taking into account two requirements (see Figure 2.4):

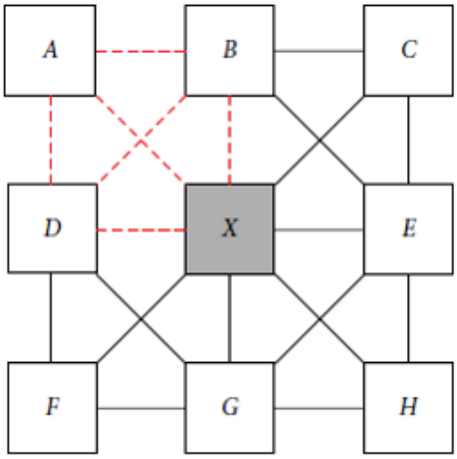

Figure 2.5: Local neighbourhood system to measure continuities. The local neighbourhood system of empty location X and its four cliques —ABDX, BCXE, DXFG and XEGH—. To determine the optimal candidate to represent the empty X location, the energy of the DCT coefficients form the four cliques is obtained, selecting the candidate with the smoothest response. This image is extracted from [3].

- Seamlessness: The overlapping area between the fixed block(s) and the candidate has(have) to be similar. This is check it by measuring SSD between these areas and thresholding the that value.
- Best continuation: In the non-overlapping area, the continuation between fixed block(s) and candidates has(have) to be the best one. To this end, a spatial GC segmentation is applied.

The main drawback of this approach is the trade-off between computational complexity and block size. It is necessary to use a small block size to assure the assumption aforementioned for this approach. Hence, the smaller the block size the higher the computational cost.

2.3.3 Block-based BGI under MRF framework [3]

In [3], an online approach based on a MRF-IMC energy minimization framework is proposed. The assumption introduced is that the TB is revealed at least in a short time interval during the training sequence.

This approach is similar to the previous approach [17] of the authors in its early stages. First the image is divided in non-overlapping blocks of size NxN. Subsequently, for each block spatial location an online clustering technique —measuring correlations and distances between already created clusters and the new candidates— is applied to narrow down the TB candidates. Then, the TB is initialized with the longer —in matter of number of blocks grouped—

cluster —represented averaging all grouped blocks— . In this point, modifications are made in comparison with [17] by introducing a MRF energy minimization framework where MRF-node is equivalent to block. Empty blocks are filled based on the scheme showed in Figure 2.5, where smoothness of the candidates in the empty location with the already fixed neighbourhood block is checked via energy of DCT macro-blocks —cliques— coefficients. From the energy computed a TB probability is computed where also the number of blocks compiled by the candidate is taken into account. The reconstruction is made under an IMC scheme. Once a complete initial reconstruction is performed, an iterative conditional mode technique is applied to avoid errors propagation that occurred in [17]. This iterative conditional scheme checks if changing a candidate increments the smoothness. In subsequent iterations the 8 neighbours of a changed position are also checked. The iterative conditional mode is repeated until convergence —no change— is obtained.

2.3.4 Online pixel labelling via LBP for BGI [4]

This method is proposed in [4] and is focused on solving the BGI task under an energy minimization framework via LBP in an online pixel-labelling scheme. Unlike other approaches where the TB has to be disclosed for a short time interval [1][15][16], in this approach the TB could be disclosed only once during the training sequence and be able to reconstruct it. Another assumption made is the aforementioned background smoothness.

The optimal pixel-labelling configuration is the one that minimizes an energy function that comprises two terms: one that encodes visual smoothness by taking into account neighbourhood information for each pixel and a second one that seeks to speed-up the posterior minimization by LBP by favouring to label pixels as background when there are pixels with the same colour in the same location —thus extending the condition of regarding the TB just once to twice— in different frames there are pixels with the same colour. The latter is weighted by a λ parameter, whose value really impact the performance when the TB is seen in a pixel location just once, thus been necessary to set $\lambda = 0$ to avoid the second energy term that makes impossible the initial assumption. The last step in this approach is to compose the TB once the optimal labelling is obtained via LBP, i.e. once the label or frame from which the TB should be selected is obtained. If the label is directly taken to compose the TB, the process used is known as IDC as mentioned in section 2.2. However, they decided to use GDC —where the gradient of each pixel in every frame is computed to compose a gradient combination field which, together with the LBP labelling, serves to decide as TB pixels those ones with closest similarity between their own gradient and the gradient combination frame—. GDC introduces two advantages:

- Takes care of small variations in the illumination among frames.
- Helps to preserve edges in the TB as this information is provided by the gradient.

The main drawback of this approach is that, although the Background smoothness assumption is in general true, there are many situations of flat foreground —people with one colour clothes or flat objects stationary or with slow-motion— that could lead to local errors, thus decreasing the performance approach.

2.3.5 Online pixel labelling BGI: a unified problem with stationary foreground detection and BGI [5]

This approach is proposed in [5] and it has an online processing —dealing with cluttered background, i.e. stationary objects and high-traffic foreground— and a two-step operation with pixel-wise and region-wise adaptation, where foreground and stationary object detection is performed respectively in a region-wise energy minimization framework via dynamic MRF.

First the background likelihood and the energy under the MRF-MAP (Maximum A Posterior) at pixel level are defined. The energy function is composed by a likelihood term —in which longer observation of a pixel value means lower energy, i.e. more likely to be background— and a prior term comprising a spatial term —higher spatial smoothness means lower energy, i.e. detecting boundaries means it is a false adapted background— and a temporal term —higher temporal smoothness means lower energy, i.e. keeping previous background values when foreground occludes the TB for a long time—, all of them weighted dynamically by three different weights. Once the energy is defined at pixel-level it is minimized efficiently at region level to be sensitive to object-wise changes. To this end, the three terms of the energy are restricted to the use of one of them —setting the weights to 0 or 1— depending on the case of foreground traffic:

- Likelihood term is used for low foreground traffic as the background could be modelled as the most repeated .
- Spatial term is used when stationary foreground objects and ghosts take place in order to study the smoothness required to be background.
- Temporal term is used when the foreground traffic is high in order to maintain the older background.

Subsequently and the last task, is to update the unreliable background in a region-wise manner. When an object stops —stationary foreground object— or moves when it was part of the background —ghost—, a set of suboptimal labels is created where the background possibilities increase with spatio-temporal coherency. Therefore, the labels are clustered by a general connected components algorithm to obtain coherent regions. Those regions will be analysed between the region formed by the old background and the new candidate, being the latter accepted or neglected depending on the spatio-temporal coherency. If the spatial discontinuity —measured in the boundaries— is low, the new candidate is rejected.

2.4 Evaluation measures

In the literature, the BGI approaches have several experimental results criteria. A group of algorithms designed specifically for BGI [3][9][15][16][17], use a ground-truth (GT) of the TB to compute some of the following measures:

- Number of error pixels (NE): Distance between ground-truth and algorithm result (BG) at pixel level. An error appears when the distance is higher than a threshold α —typically $\alpha = 20$ [3][9][15][16][17]—:

$$Error(x,y) = \begin{cases} 1 & if \quad |BG(x,y) - GT(x,y)| > \alpha \\ 0 & otherwise \end{cases}, \tag{2.1}$$

$$NE = \sum_{\forall(x,y)} Error(x,y), \tag{2.2}$$

- Average of error pixels (AE): This measure is computed as the average gray-level error:

$$AE = \frac{NE}{k}, \tag{2.3}$$

  where k is the total amount of pixel locations.

- Number of clustered error pixels (NC): This error occurs when the four neighbours connected with an error pixel are also erroneous. The purpose of this measure is to take into account errors due to objects in the TB obtained by the BGI approach:

$$ErrorCluster(x,y) = \begin{cases} 1 & if \quad |BG(x,y) - GT(x,y)| > \alpha \ \ \forall \ \ (x,y) \in \{neigbourhood\} \\ 0 & otherwise \end{cases}, \tag{2.4}$$

$$NC = \sum_{\forall(x,y)} ErrorCluster(x,y), \tag{2.5}$$

  where *neighbourhood* is defined as the 4-connected neighbourhood, i.e. top, bottom, left and right pixel neighbours.

The aforementioned quality measures could be extended to the colour space by averaging the different channels. Other approaches [4][12][13][18] just show a visual comparison of the generated TB from their approach and other methods, instead of a more rigorous evaluation. Furthermore, in [10] an online comparison of the generated TB at different temporal instants is shown. Other approaches more focused on background subtraction results [5][14] do not evaluate the generated TB.

2.5 Conclusions: BGI issues and strategies to solve them

In the previous sections of this chapter, the state-of-the-art algorithms and their features have been studied to understand the issues and singularities of the BGI task. At this point, we can expose the key limitations that should be tackled in a BGI algorithm:

- Background visibility: When a pixel or region location from the TB is disclosed for a short time interval during the training sequence, the majority of the candidates to be TB belong to foreground objects. This situation has different cases:

  1) Stationary: This problem is also known as sleeping person and could lead, in some approaches, to the ghost phenomenon. It consists on having in a spatial location an object occluding the TB during a considerable amount of time of the training sequence, thus difficulting the reconstruction in that area. In some algorithms, where there is an online operation and the stationary object is part of the TB, when it leaves the algorithm must update the TB, thus dealing with the ghost situation. To deal with stationary foreground objects or ghost removal, is typical to use the background smoothness assumption in order to set as TB those candidates that do not introduce discontinuities.

  2) Low visibility due to high foreground traffic: These are situations where although the TB is visible in few cases, there is no a temporal dominant foreground. Therefore, it is easier to estimate the TB than in the stationary object case. In the case of low visibility caused by high traffic, a necessary premise is made: the TB is revealed at least once. Nevertheless, this condition could be extended to a disclosure of the TB in a short interval when temporal pixel variations are used as information, i.e. motion information.

- Photometric factors: This category compiles shadows and illumination changes and camouflages, both aspects affecting the BGI task:

  1) Shadows and illumination changes: Both the interactions between the light sources and the objects in the scene and the variations in the global illumination, bring out variations in the pixel values different from foreground objects, thus hindering the BGI task. In the case of shadows originated by foreground objects, they do not belong to the TB. However, there are other shadows or illumination changes inherent to the TB. To deal with these problems it is common to suppose constant illumination in the scene as it is not clear who is the TB.

  2) Camouflages: This fact comes into play when the foreground object and the covered TB have similar texture and colour. This situation could lead to errors, due to an easy

misunderstanding between the TB and the foreground object. This is a challenging issue.

After studying the issues and strategies proposed to deal with the BGI task, we can conclude that this labour is still opened.

Chapter 3

Proposed algorithm

3.1 Overview

The proposed approach analyses the training sequence making batch processing and using region-level analysis. Specifically, the training sequence is processed at block level of size NxN —$N = 16$—. Therefore, it is necessary for the TB to be revealed in each block location at least in one frame to allow a correct background initialization. The partition into non-overlapping blocks is made in order to analyse the spatio-temporal properties of the training sequence. As broad view, the algorithm establishes some candidates in each spatial location to compose the TB, which amount is reduced via temporal analysis. Subsequently, the TB is built iteratively in a IMC (see section 2.2) by selecting in each location the candidate with highest probability to be background according to spatial constraints.

Therefore, first a temporal analysis is made to narrow down the number of candidates. To this end, a three-step (see Figure 3.1) analysis is performed:

1) Dimensionality reduction: The data dimension is reduced via Principal Component Analysis (PCA) to speed the algorithm.

2) Motion based candidate filter: Blocks suffering motion are discarded as TB candidates.

3) Agglomerative hierarchical clustering: The optimal candidates to perform the clustering are obtained grouping the available ones without using any threshold.

Secondly, an spatial analysis is made to build the TB using spatial constraints via multiple paths in an iterative two-step scheme:

1) Seeds selection: An already set block location is set.

2) Multipath reconstruction: The neighbourhood of the seed is set combining different reconstructions. The process is repeated until the entire TB is filled (see Figure 3.1). Moreover, a seed location could be discarded if some conditions are reached.

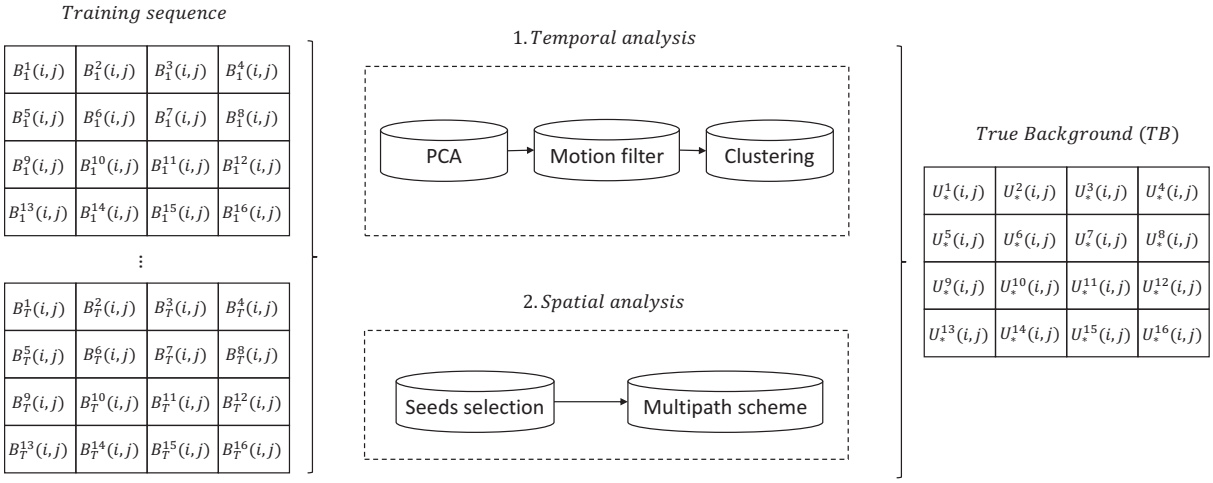

Figure 3.1: Overview of the proposed algorithm. A temporal analysis and an iterative spatial analysis are performed to compute the TB from the training sequence. The training sequence of length T is divided into non-overlapping blocks $B_t^s(i,j)$. Subscripts denote temporal indexes, superscripts denote spatial indexes and i and j denote the row and column block index respectively. Each block location of the TB is filled with the target temporal candidate $U_*^s(i,j)$. In the figure, the image is divided into 16 block locations, nevertheless is just an example.

For clarification, stages 1 and 3 from the temporal analysis are previous work and the rest of the stages is developed in this work.

3.2 Temporal analysis

The purpose of the temporal analysis is to reduce the T —length of the training sequence— candidates to compose the TB in each spatial location. To that end, the training sequence is analysed at block level, as the best state-of-the-art approach known [3], setting the block size NxN with $N = 16$. The temporal analysis is strongly similar to the one proposed in the Master Thesis [21].

Once the sequence is divided into non-overlapping blocks (see Figure 3.1), the aim is to group those blocks into similar clusters —groups— in each spatial location, in order to narrow down the number of candidates to represent the TB in each location (see Figure 3.2). To that end, a three-step method is applied: Dimensionality reduction, Candidate filtering and Agglomerative hierarchical clustering.

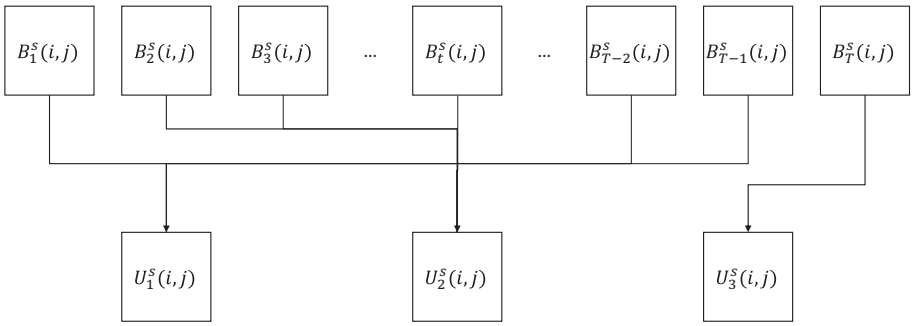

Figure 3.2: Example of candidates reduction. Similar blocks are grouped into clusters U_p^s, where $p \in \{1...NC\}$ and NC is the number of clusters in the location s. In the figure the T candidates are reduced to only 3.

3.2.1 Dimensionality reduction

The first step done, is a reduction of the data dimensions in order to decrement the algorithm complexity as performed in [21]. To this end, Principal Component Analysis (PCA) at each block temporal location is applied. PCA is a non-supervised algorithm for data —matrix of patterns in each row and variables for each pattern in each column— dimensionality reduction. It performs a linear transformation of the data that redefines the variables, composing a new set of variables in which the variance is concentrated in the first ones —each variable has a percentage of the variance—. The concentration of the variance in the first variables —columns— allows to split the dimension of the data by selecting the desired percentage of it. This is very helpful when variance is equal to useful information, due to the ability to reduce complexity while keeping the intrinsic properties of the data.

To apply PCA, the first task is to create several matrices D^s containing the whole data of the training sequence at each s spatial location, i.e. each row of D^s is a rasterized block b_t^s of size $N \times N \times 3$ —3 colour channels— at different temporal instants in s. The algorithm is going to be applied over that matrices (see Figure 3.3). Hence, if the initial dimension of D^s is $T \times (N \times N \times 3)$ —T rows of length $N \times N \times 3$— the result of this stage is going to deliver a Q^s matrix of size $T \times M$ with $M < N \times N \times 3$. Q^s, with $b_t'^s$ blocks in each row, is the input of the next stage, Motion based candidate filtering.

The reason to apply PCA is the compliance with the requirement of information being represented by data variance to perform a useful dimensionality reduction. In our case —join different blocks into clusters— we just need information that induces variations, since the static parts of the block provide redundant information to perform the clustering task. It is important to highlight that the transformed domain created by PCA is only going to be used to speed-up

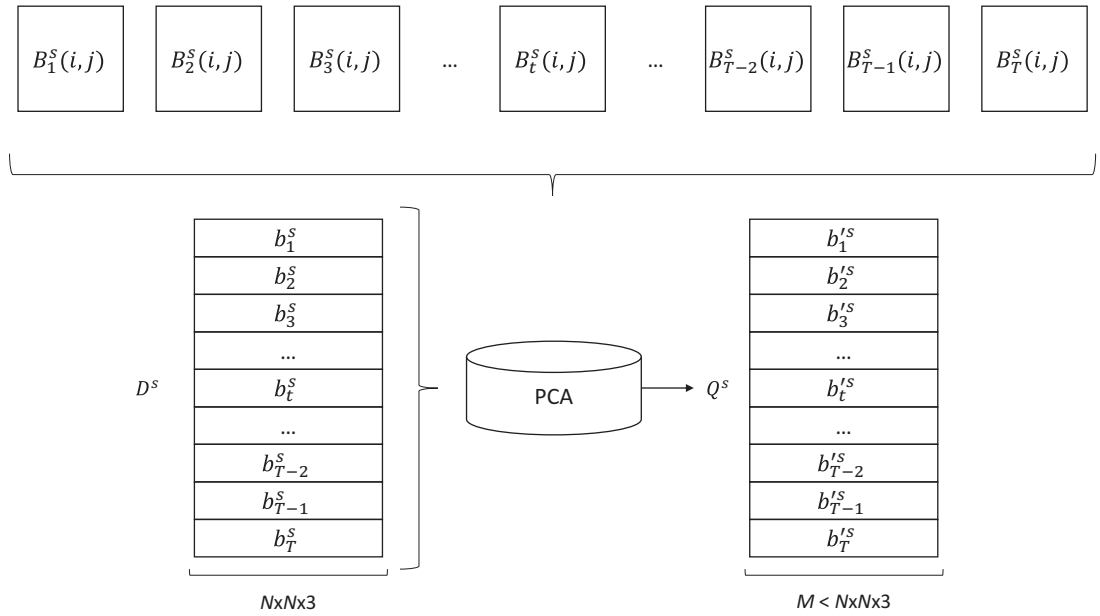

Figure 3.3: PCA process scheme for a spatial location. First blocks B_t^s are rasterized to b_t^s to compose each row of the matrix D^s. Subsequently, PCA is applied over that matrix, obtaining the new and reduced data Q^s, which serves as input for the next step in the three-step clustering.

the clustering and not as real information for the spatial analysis stages that will be performed to select the optimal candidates for the TB.

3.2.2 Motion based candidate filtering

Second step is a Motion filter, whose purpose is to discard blocks —rows from Q^s— which suffer motion activity, as they are not valid to describe the static TB. That motion information is computed from original images of the training sequence —$\{I_1...I_T\}$— and not from Q^s. The proposed strategy —not included in [21]— is simple yet effective. First, a frame difference at pixel-level is applied between each image and its $k-$separated previous image from the training sequence —k should have a small value—. Then, those differences are thresholded to determine if there is motion in each pixel of each temporal instant by obtaining a binary motion image MO_t:

$$MO_t(x,y) = \begin{cases} 1 & if \quad |I(x,y)_t - I(x,y)_{t-k}| > \eta \\ 0 & otherwise \end{cases}, \tag{3.1}$$

where x and y denote the pixel locations, I_t is the image in the temporal instant t and η is a threshold [22] to compute motion. $MO_t(x,y)$ takes value 1(0) when motion (no motion) is detected at pixel level. Subsequently, block motion information is obtained by setting as motion

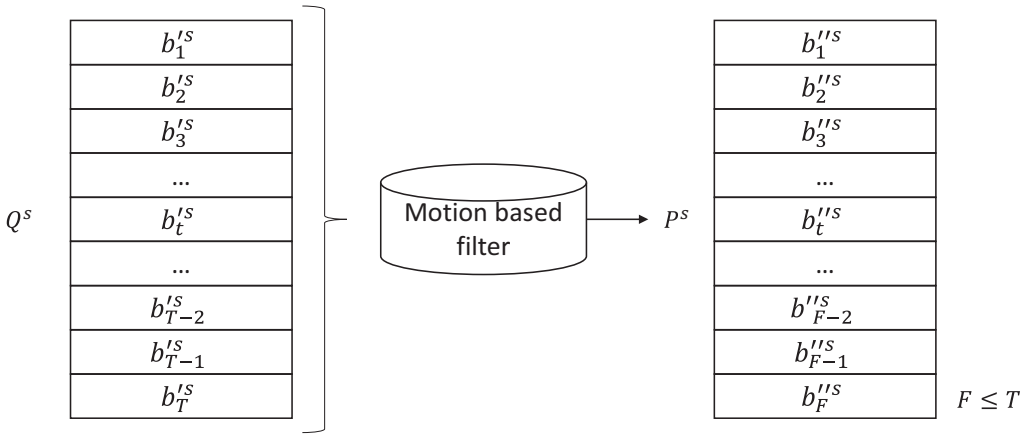

Figure 3.4: Motion based candidate filtering scheme for a spatial location. After the filter, right area of the image, the new data P^s has F rows that usually will be lower than T, however if no motion is detected along the s location, $F = T$ and no motion filtering is performed. The motion filter only uses Q^s rows to discard rows —blocks— where motion is computed.

blocks those with any pixel suffering motion. Hence, the blocks B_t^s where motion is detected are rejected as background candidates, thus performing the output matrix of this stage P^s, whose number of rows is $F \leq T$ (see Figure 3.4). The incorporation of motion information extracted from $k-$separated frames modifies the requirement of visualizing the TB in one frame to a small temporal interval —as other state-of-the-art approaches do [1][3]—. Furthermore, it is important to highlight that this stage does not use the blocks obtained in the PCA process, therefore this stage could be implemented in parallel to the dimensionality reduction stage —subsection 3.2.1—.

3.2.3 Agglomerative hierarchical clustering

The target of this stage is to reduce the amount of candidates representing the TB in each location by building temporal clusters with the already reduced data from the Motion filter stage. These clusters will be created by maximizing the difference between them and minimizing the difference among their members, avoiding the use of thresholds during the process. The inputs and outputs of each spatial location B_t^s of this stage are, the P^s matrices and the U_p^s clusters respectively (see Figure 3.5). U_p^s will be the candidates to compose the TB in the location s —i.e. the candidates representing each cluster— and the best one —U_*^s— will be selected in the spatial analysis.

To perform the clustering, the classical method called agglomerative hierarchical [23] has been applied following a similar process as [21]. Roughly, it works as explained below:

- Initially, each possible candidate is treated as a possible cluster.
- Secondly, initial candidates are grouped iteratively —one in each iteration— by merging in the same cluster the two candidates with close distance. This step is repeated until only one cluster is obtained. Therefore, a tree is built with the number of levels of the iterations done.
- Finally, a cluster validation technique should be applied to determine the optimal number of clusters (NC) —level of the tree— according to some criteria.

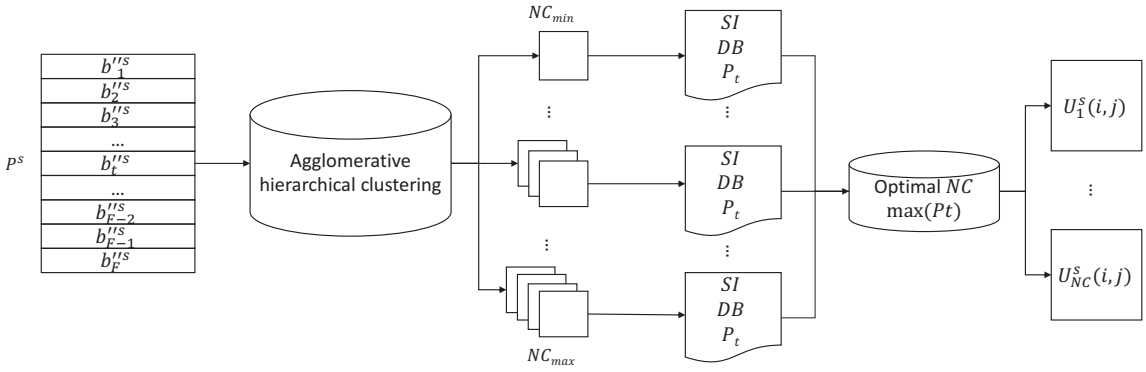

Figure 3.5: Agglomerative clustering scheme. In each location s, the P^s matrix is used to perform the agglomerative hierarchical clustering where several groupings of clusters are computed —from NC_{min} to NC_{max} clusters—. Subsequently, a probability from each clustering is obtained from the internal validation indexes SI and DB. The optimal clustering is selected as the group with the maximum probability.

In our task, each initial candidate is a rasterized and reduced block b'^s_t of the same location s and different temporal instant t, i.e. a row of the P^s matrix. Then the distance between two clusters U^s_i and U^s_j —$D(U^s_i, U^s_j)$— is defined as the highest euclidean distance —the use of the highest distance is known as complete linkage criteria— among members of U^s_i and U^s_j:

$$D(U^s_i, U^s_j) = max(d(b''^s_i, b''^s_j)) \tag{3.2}$$

where b''^s_i and b''^s_j denote the set of blocks compiled into U^s_i and U^s_j respectively. Therefore, a tree with different amount of clusters in each level is obtained. Then, a clustering validation stage is going to be applied to obtain the best clustering attending to internal similarity and external difference of the clusters.

The levels of the tree to analyse have been bounded (see Figure 3.5) between a minimum and maximum amount of clusters —NC_{min} and NC_{max} grouping—. To perform this task, an approach based on subintervals of stable intensity (SSI) has been applied (see Section 2.2). The

selected interval to sweep starts from $NC_{min} = 1$ and ends in the number of stable intervals at block level —a block is stable when every pixel inside the block does not suffer motion—. As the worst case is the one in which the number of continuous SSI is caused by different representations of the TB, this amount is equal to NC_{max}. SSI are computed attending to the motion information extracted in the stage Motion based candidate filter (subsection 3.2.2).

The purpose of the validation task, is to check each possible level of the tree to determine the optimal clustering. The input is the tree from the hierarchical clustering and the output is a splitted level of the tree containing the NC optimal clusters U_p^s where $p \in \{1...NC\}$. Several validation indexes could be used [24]. Two categories could be distinguish among them [21]:

- External indexes: This type of indexes compare two different clusters where one of them is known —ground-truth (GT)— and the other one is the output of the algorithm. Some well-known external indexes are Rand, adjusted Rand, Jaccard, and Fowlkes Mallows (FM).

- Internal indexes: This type of indexes are used to measure the goodness of a clustering structure without any external information, i.e. performing measures just with the features of the dataset. As in our task, usually no GT is available to check the optimal clustering, thus NC is computed via an internal validation index. Some well-known internal indexes are Silhouette (SI), Davies-Bouldin (DB), Calinski- Harabasz, Dunn, Hubert-Levin (C-index), Krzanowski-Lai and Hartigan ; the Root-mean-square standard deviation (RMSSTD), R-squared, Semi-partial R-squared (SPR) and Distance between two clusters (CD); the weighted inter-intra index; and the Homogeneity and Separation.

Based on previous work [21], the internal indexes used to measure each possible clustering have been SI and DB (see Figure 3.5). Silhouette index: measures the compactness and separation among clusters. A higher average value of this measure implies a better quality of the cluster. Hence, the optimal NC is the clustering with higher Silhouette index. Davies-Bouldin index: Measures the similarity between each group and its highest similar one. Small values in this index corresponds to compact clusters whose centroid is far from the rest, thus the minimum DB determines the optimal NC.

After computing both indexes, they are combined to determine the optimal NC. To this end, both indexes are normalized —100% of score is assigned to the maximum SI and minimum DB respectively— and then a final probability P_t for each group is obtained:

$$P_{SI} = \frac{SI - min(SI)}{max(SI) - min(SI)} \tag{3.3}$$

$$P_{DB} = \frac{DB - max(DB)}{max(DB) - min(DB)} \tag{3.4}$$

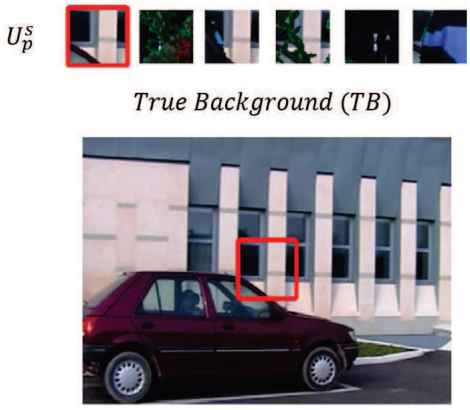

Figure 3.6: True Background candidates example. In the figure the TB of one video sequence is shown. The red square marks one spatial location whose candidates U_p^s obtained in the temporal analysis are presented. The candidate marked in red is the correct one —U_*^s— that is going to be determined by the spatial analysis.

$$P_t = P_{SI} \cdot w + (1 - w) \cdot P_{DB} \tag{3.5}$$

where P_{SI} and P_{DB} are the normalized measures and w weights the contribution of each measure —initially 0.5— to the probability P_t of each cluster. NC and therefore, the optimal clustering is selected as the one with the highest P_t. At this point, the remaining task of the temporal analysis is to compute the block representing each cluster U_p^s, which is obtained as the average of all the B_t^s composing each cluster.

3.3 Spatial analysis

The target of this stage is to obtain the TB from the candidates U_p^s (see Figure 3.6). To this end, a proposed multipath local reconstruction based on spatial constraints is performed. This spatial constraints are used to enforce the extended assumption of background smoothness that, as mentioned in Section 2.2, is the premise for this task that provides best results. The reconstruction process is divided in three stages: Seed selection, Multipath reconstruction and Reconstruction enhancements. The aim is to obtain among the U_p^s candidates in each spatial location, the one representing the TB: U_*^s.

3.3.1 Seeds selection

The aim of this stage is to obtain an initial TB —partial TB reconstruction— with the highest number of proper TB blocks (see Figure 3.7). This is a key step as an error in this point will

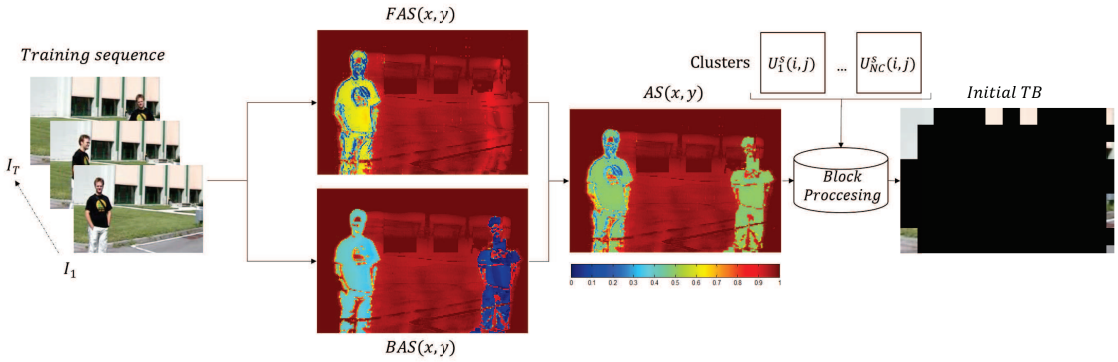

Figure 3.7: Seed selection scheme. The Figure shows a training sequence where a stationary object change its position from the beginning of the sequence to the end of the sequence. AS score, computed from FAS and BAS, allows to determine the spatial locations where a mayor cluster could not be suitable for an initialization. Subsequently, a block processing to compute AS at block-level and determine the initialized locations s with the major cluster among the available $U_p^s$is performed. The result is an initial TB, that will be fulfilled in the next stages of the spatial analysis.

lead to great errors in the estimated TB. This task is commonly performed in the state-of-the-art [2][3][9][17] by selecting as initial TB blocks —U_*^s— those locations where there is only one cluster. If there is no location with one cluster, the location having the cluster that groups more members —among all the locations— is initialized. If several locations comply this fact, all of them are initialized. Attending to the task of avoiding errors, this strategy is usually valid. However, a better analysis could be done in order to increment the amount of proper U_*^s selected and to assure avoiding locations where stationary objects could be the most repeated block —if the time interval of stationarity is enough long— and therefore, the selected cluster.

We propose to include motion information to obtain those block locations s that suffer none or few motion along the training sequence and initialize them with the major cluster[1]. To avoid those locations where there is a stationary object an easy assumption is taken: if a stationary object is occluding the TB in the first frame of the training sequence it is not going to remain there in the last frame. This assumption is logic, as an object been stationary for the whole training sequence is considered as part of the static TB and not as a stationary object. Once this assumption is formulated the way of computing the initial locations could be understood easily:

- First, frame differences at pixel-level between the first frame and the rest and between the last frame and the rest are performed. Those differences are thresholded (τ and σ) [22], accumulated and averaged over time —dividing by $T-1$, i.e. the number of differences—

[1]The major cluster of a spatial location is the cluster grouping more members in that location

to obtain Forward and Backward pixel-level activity scores,FAS and BAS. The pixel level activity score (AS) is obtained averaging FAS and BAS:

$$FAS(x,y) = \left(\sum_k \begin{cases} 1\, if & |I_1 - I_k| > \tau \\ 0 & otherwise \end{cases} \right) /(T-1), \tag{3.6}$$

$$BAS(x,y) = \left(\sum_l \begin{cases} 1\, if & |I_T - I_l| > \sigma \\ 0 & otherwise \end{cases} \right) /(T-1), \tag{3.7}$$

$$AS(x,y) = \frac{FAS + BAS}{2}, \tag{3.8}$$

where $k \in \{2...T\}$ and $l \in \{T-1...1\}$, are the indexes to sweep the images I of the training sequence. The reason why forward and backward motion is computed is to assure that no location with stationary object is selected.

- Then, activity score at block-level, AS^s, is computed by labelling each block with the maximum $AS(x,y)$:

$$AS^s = max(AS(x,y)), \tag{3.9}$$

where pixel locations $(x,y) \in s$.

- Finally, locations with the minimum AS^s among the whole TB are selected to be initialized with the "major cluster" obtained in the temporal analysis (see Figure 3.7). Hence, in the locations obtained, U_*^s is set with the major cluster.

It is necessary to justify why the already extracted motion information from subsection 3.2.2 is not used again. That motion between $k-$separated frames is not useful to obtain the activity in a spatial location for the desired task: compute the activity in an spatial location along the training sequence. For example, if in the scene there was a statiorary foreground during some time and subsequently that static foreground leaves, the accumulated motion during the whole training sequence would be very low —as only motion when the foreground leaves could be obtained—, thus obtaining a low activity in a location where de major cluster could belong to stationary foreground.

3.3.2 Multipath reconstruction

This subsection proposes an initial multipath scheme to initialize the background from the initial TB built in the seeds selection stage. The idea is to perform iterative local reconstructions of the seeds neighbourhoods under certain background smoothness constraints, until the TB is entirely built.

In each seed neighbourhood, a reconstruction is performed via combination of multiple local estimations. The multipath idea is introduced here: several reconstructions for the seed neighbourhoods are estimated using spatial constraints and in different orders —different paths—. The Iterative Model Completion (IMC) process is explained below:

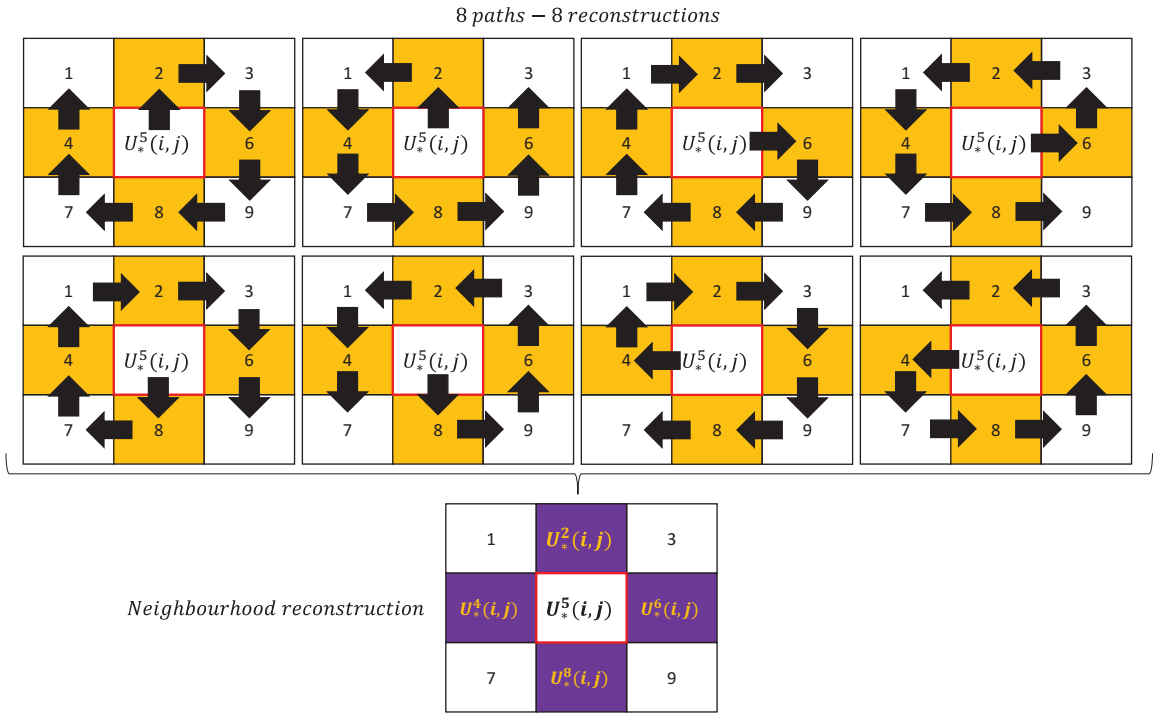

Figure 3.8: Multipath reconstruction scheme. Location $s = 5$ is selected as seed and —starting from that location— 8 paths are followed to perform 8 reconstructions of the local neighbourhood —yellow ones—. Finally, a unique reconstruction —purple one— is computed from the 8 reconstructions —yellow ones—.

- A seed position is chosen according to the amount of neighbours already set. A position with more neighbours is more suitable to be reconstructed as more information is already available in the paths.

- The estimation of the neighbourhood is performed going over 8 paths —4 neighbours with two directions each one—, thus 8 reconstructions are built. In Figure 3.8 an example of this scheme is shown. In the figure, the location $s = 5$ is selected as seed —where U_*^5 is already set— and the other locations are empty; subsequently, each empty location —$s = \{1, 2, 3, 4, 6, 7, 8, 9\}$— is filled starting from $s = 5$ and following the aforementioned 8 paths —black arrows—; therefore, 8 reconstructions of the local neighbourhood —yellow ones—

are computed. Each path reconstruction is based on colour continuity (CC) between the candidates to set in each empty location and its adjacent already set blocks, i.e. each path location is filled according to colour continuity with the already fixed neighbours. Hence, here the background smoothness is understood as introducing the lower colour discontinuity between adjacent (ad) already set neighbours and the empty member (em) that has to be set in the path (see Figure 3.9):

$$EC = \left(\sum_{edge} |U_*^{em}(x_i, y_i) - U_*^{ad}(x_{i\prime}, y_{i\prime})| \right) /N \tag{3.10}$$

$$CC = [\sum_{k} EC]/k \tag{3.11}$$

where k is the number of adjacent already set neighbours and Edge Continuity (EC) is the averaged difference of the adjacent edges —(x_i, y_i) and $(x_{i\prime}, y_{i\prime})$ are adjacent pixels in the edge— of the empty member (em) and the adjacent (ad) one —one of the four neighbours—. Among the 8 neighbours only 4 are used as reconstruction —up, down, left and right— as the diagonal ones are built with CC of less neighbours.

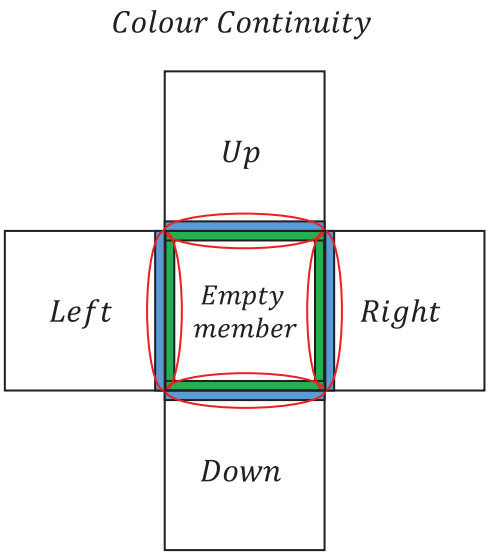

Figure 3.9: Colour continuity scheme. The CC is measured between the edges —green pixel row or column— of the empty member and the adjacent —blue— edges of its neighbours. The four possible areas to measure CC are marked in red. Nevertheless, only already initialized blocks are used in the measure.

- Finally, the Neighbourhood reconstruction is performed, selecting the best candidate U_*^s for each 4 neighbouring locations among the 8 possible reconstructions. The best candidates

are selected in matter of best —minimum— CC obtained. This is shown in Figure 3.8, where a unique reconstruction —purple one— is computed from the 8 reconstructions —yellow ones—.

- Steps 1,2 and 3 are repeated until the TB is built.

The aim of a multipath scheme is to address erroneous reconstructions of adjacent blocks induced by arriving at a block to reconstruct in a certain way in which the continuity measure can fail. Hence, arriving at the block by different paths can handle local errors of the CC measure.

With the aforementioned multipath scheme, an important issue was detected due to the limited knowledge of the neighbours of an empty member when a path is reconstructed, i.e. the CC measure of an empty member is commonly computed with less than 4 neighbours. Empirically, a situation causing the errors was observed: sometimes several candidates in a location have very close CC and the best one is not the TB. However, observing the erroneous selected candidate, a non homogeneity inside the block is appreciated (see Figure 3.10). Although, the multipath idea is able to deal with an erroneous reconstruction due to a possible erroneous way of arriving to the block, the external edges of the neighbourhood are not analysed and discontinuities could arise there. Furthermore, errors due to a non-compliance of the fact that the best candidate is the one with best CC also appear (see Figure 3.10). The next section shows the enhancements proposed to tackle the commented limitations —external edge and non-compliance of CC—.

3.3.3 Reconstruction enhancements

This subsection shows the modified multipath scheme, where the external edge problem and the non-compliance of CC measure are tackled. The enhancements proposed are related with the enforcement of the smoothness by measuring the internal homogeneity of the block, the colour similarity among adjacent blocks and the re-election of the seed location.

As mentioned in section 3.3.2, when the external edge is not analysed or simply if the adjacent edges between two blocks do not comply the CC measure for the TB candidate, problems could come into play. Hence, the initial proposed solution is to detect when a problem could be occurring and avoid the reconstruction of that neighbourhood by selecting a new seed location —seed re-selection with the same criteria explained in the previous section but avoiding the rejected locations—. The process is as follows:

- When a candidate is selected in the path reconstruction based on CC, it is also checked if there is any candidate with a close CC to the selected candidate —difference lower or equal than a threshold ρ—:

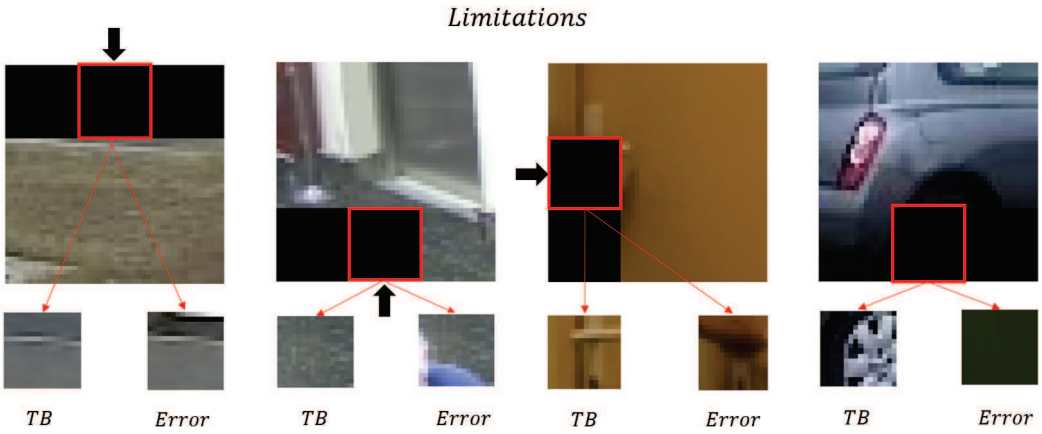

Figure 3.10: Multipath scheme limitations. From left to right: 3 problems caused by the external edge and another one caused by the non-compliance of CC. In every example, the position on the neighbourhood to be filled is marked in red and the correct candidate —TB— and the erroneous candidate selected —Error— are displayed. Examples 1 to 3 have a black arrow that is marking the external edge that is never analysed in the explained multipath scheme. In such external edge a non-measured discontinuity appear in the selected erroneous block. On the other hand, the TB blocks for the three examples have internal homogeneity or colour similarity with the surroundings, fact that is exploited in the enhancements done that are explained in section 3.3.3. In the last example, an erroneous block is selected due to having better CC than the TB one. Colour similarity can be exploited for this cases.

- In affirmative case: intra-block Homogeneity (H) and inter-block Colour Similarity (CS) are checked to address external edge and non-compliance of CC measure respectively. H is computed via energy of the DCT coefficients —with the continuous set to 0— of the U_p^{em} candidates in the empty member (em) location. CS is computed via pixel by pixel cosine distance between em and the adjacent already set neighbours (ad):

$$H = \sum_{coef} coef\{DCT\{U_p^s\}\}^2 \tag{3.12}$$

$$CS = [\sum_{k} cos(\measuredangle\{U_p^{em}(x,y), U_*^{ad}(x,y)\}) \quad]/k \tag{3.13}$$

  where k is the amount of adjacent already set neighbours. If the alternative block has lower H —higher homogeneity— or lower CS —higher colour similarity—, a critical situation is found due to low smoothness and the seed location is rejected.

- In negative case: The reconstruction continues as explained in 3.3.2.

Moreover, if all the seed locations are rejected, a mandatory reconstruction of a critical location has to be done. In this case, the first seed location —that was initially rejected— is used to

reconstruct the neighbourhood via Spatial Continuity (SC) computed averaging the obtained smoothness measures:

$$SC = \frac{CC + H + CS}{3} \tag{3.14}$$

where each individual measure is previously normalized between 0 and 1 to allow the combination. The normalization is computed with the data of the 8 possible reconstructions. The modified process is repeated until the entire TB is built.

Chapter 4

Experimental work

4.1 Dataset

This section describes the dataset compiled to evaluate the proposed approach against the state-of-the-art methods. Public sequences used are selected from:

- AVSS2007 (www.eecs.qmul.ac.uk/~andrea/avss2007.html): this dataset compiles sequences with increasing complexity for two scenarios: abandonment of objects and illegal parking.
- PBI[2] (www.diegm.uniud.it/fusiello/demo/bkg/): this dataset compiles several video sequences for stationary camera and moving camera.
- LIRIS2012 (liris.cnrs.fr/voir/activities-dataset/videoframes.html): this dataset compiles several video sequences of human activities.
- PETS2009 (www.cvg.rdg.ac.uk/PETS2009/index.html): this dataset includes multi-sensor sequences containing different crowd activities.
- LIMU (limu.ait.kyushu-u.ac.jp/dataset/en/): this dataset includes several videos for the detection of moving objects.
- TRECVID (trecvid.nist.gov/trecvid.data.html): this dataset includes several sequences for video-surveillance event detection.
- Wallflower (research.microsoft.com/en-us/um/people/jckrumm/wallflower/testimages.htm): this dataset compiles several video sequences for the task of Background Subtraction.

Among the mentioned datasets, 25 sequences have been selected. Furthermore, 6 sequences from a faculty hall have been included. Therefore, a dataset of 31 sequences covering different scenarios (see Figure 4.1) and complexities is built.

Figure 4.1: Dataset scenarios. The different environments included in the dataset are shown in the figure.

It is important to highlight that sequences used are composed by some parts of the original ones —details in the Appendix A—. The reason for bounding some original sequences is the selection of challenging intervals for the BGI task. The Table 4.1 shows the properties of each video sequence, including complexity features —Stationarity, Visibility and Shadows—. It is necessary to explain how some of the complexity levels are defined:

- Stationarity —complexity of stationary foreground in matter of size and time—:
  - Low: Few stationary regions of small size and/or less of 50% —percentage of the training sequence length— of stopped time of the regions.
  - Medium: Several regions of small size with stopped time from 50% to 70%.
  - High: Multiple stationary regions with stopped time higher than 70% or few ones of great size with stopped time higher than 50%.

- Visibility —complexity of background visualization—:
  - Low: Most of the TB is disclosed along the training sequence and there are not areas with low visibility.
  - Medium: There are areas where the TB is disclosed few times, however many other ones are revealed.
  - High: Low revelation of the TB along the training sequence in every area.

Video (Dataset) - Number	Frames	Scene	Stationarity	Visibility	Shadows
AB_E (AVSS2007) - 1	400	Indoor	-	L	L
AB_H (AVSS2007) - 2	400	Indoor	H	M	L
AB_M (AVSS2007) - 3	400	Indoor	-	M	M
PV_E (AVSS2007) - 4	400	Outdoor	-	L	L
BSM_1 (LIMU) - 5	400	Outdoor	H	L	L
BSM_2 (LIMU) - 6	350	Outdoor	L	L	L
Inter (LIMU) - 7	400	Indoor	H	L	L
Park_a (PETS2009) - 8	339	Outdoor	-	M	H
Park_b (PETS2009) - 9	240	Outdoor	M	L	L
snellen (PBI) - 10	334	Indoor	-	H	M
LGW (TRECVID) - 11	400	Indoor	-	M	M
board (PBI) - 12	228	Indoor	H	M	M
bootstrap (Wallflower) - 13	294	Indoor	L	L	M
ca_vignal (PBI) - 14	258	Outdoor	M	L	L
cam4 (TRECVID) - 15	300	Indoor	M	L	L
foliage (PBI) - 16	258	Outdoor	-	H	L
granguardia (PBI) - 17	400	Outdoor	H	M	L
vid16 (LIRIS2012) - 18	380	Indoor	H	L	L
vid22 (LIRIS2012) - 19	345	Indoor	M	M	L
vid36 (LIRIS2012) - 20	128	Indoor	M	M	L
vid44 (LIRIS2012) - 21	254	Indoor	-	M	L
vid62 (LIRIS2012) - 22	208	Indoor	-	M	L
vid8 (LIRIS2012) - 23	315	Indoor	-	H	L
vid80 (LIRIS2012) - 24	400	Indoor	-	M	L
video11 (PBI) - 25	349	Outdoor	L	H	L
hall_a (HALL) - 26	400	Indoor	L	M	H
hall_b (HALL) - 27	220	Indoor	-	M	L
hall_c (HALL) - 28	400	Indoor	H	M	M
hall_d (HALL) - 29	399	Indoor	H	H	H
hall_e (HALL) - 30	400	Indoor	H	M	H
hall_f (HALL) - 31	400	Indoor	H	L	L

Table 4.1: Dataset properties. The properties displayed are: number of frames —Frames—, open air scenario or not —Scene—, the complexity of stationary regions —Stationarity—, the revelation of the True Background in matter of complexity —Visibility— and the level of shadows and illumination changes in the scene —Shadows—.

- Shadows —complexity in matter of amount of shadows and illumination changes—:
  - Low: Few shadows or illumination changes. No local or global changes in the illumination.
  - Medium: Several shadows and no illumination changes. No global changes in the illumination.
  - High: Several shadows or illumination changes affecting the scene.

The number of the video sequence displayed in Table 4.1 denotes the order in which the sequences are displayed in the results presented in section 4.2.

4.2 Experimental results

This section exposes several experiments performed with the proposed approach. First, the clustering technique proposed is compared with the clustering used in [3] (subsection 4.2.1). Secondly, the quality of the seed selection method in comparison with the traditional one [3] is computed (subsection 4.2.2). Finally, several comparisons between the proposed approach and state-of-the-art approaches are shown (subsection 4.2.3).

It is important to highlight that Average of Error pixels (AE) is the measure used for the experimental results and not the Number of Clustered error pixels (NC). This fact is due to NC measure penalizes region-wise approaches with respect to pixel-wise as an erroneous region in a region-wise approach is more likely to count a higher amount of NC errors —a pixel and its 4 neighbours fail— than an erroneous pixel in a pixel-wise approach.

The parameters included in the algorithm —k, i.e the $k-$separation introduced in the motion estimation of the temporal analysis (section 3.2) and ρ, i.e. the threshold to consider that two candidates are similar in the enhancements of the multipath scheme in the spatial analysis (section 3.3)— are heuristically set to $k = 3$ and $\rho = 5$.

4.2.1 Clustering method evaluation

The proposed approach, unlike [3], is able to perform the clustering task without parameters while keeping the required speed of the process thanks to the Dimensionality reduction stage (section 3.2.1) and the Motion based filter stage (section 3.2.2). The clustering in [3] is done in an online fashion by including in each new frame new members in the existent clusters when a distance (correlation) measure is lower (higher) than a fixed threshold or creating new clusters if not. In Figure 4.2 the performance of the proposed clustering and the one performed in [3] is measured. To this end, first the best TB is reconstructed by selecting in each spatial location

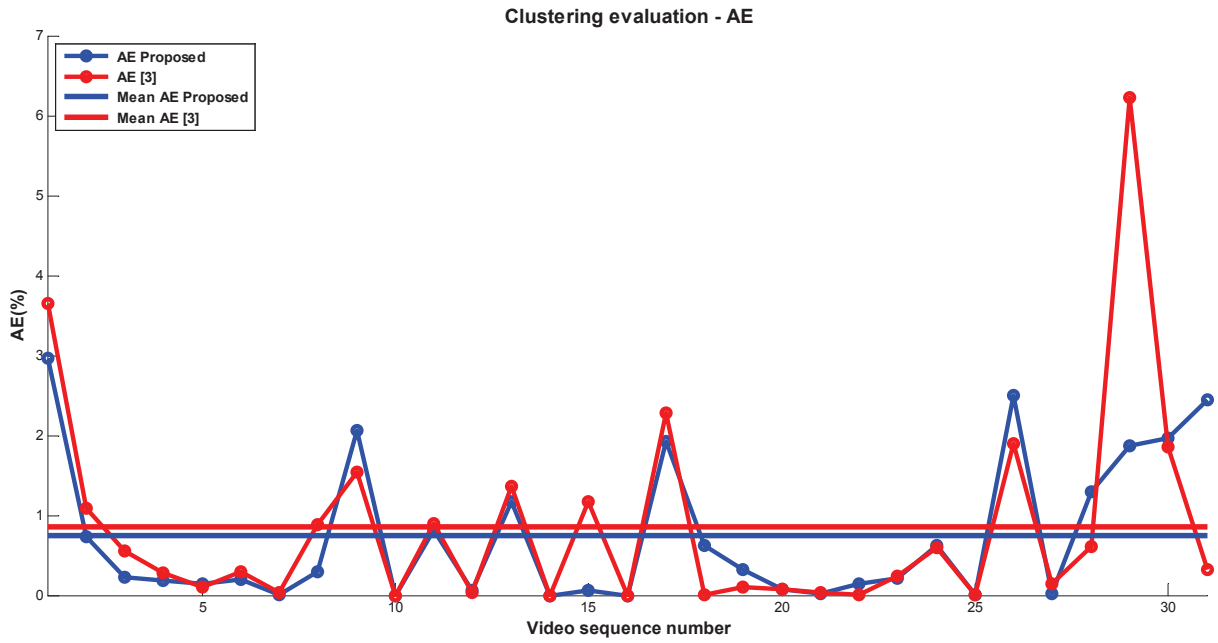

Figure 4.2: Clustering evaluation. The figure shows AE averaging each colour channel. The x-axis is the number of the video sequence referenced in Table 4.1. As can be appreciated, both clustering approaches show very close performance with the advantage of the proposed one of avoiding the use of thresholds in the clustering task. The average AE for [3] is 0.85% and for the proposed method 0.74%.

the target candidate U_*^s, as the U_p^s with lower distance with the GT at each s location —GT^s— :

$$dist^s = max_{(x,y)}\{|U_p^s - GT^s|\}, \tag{4.1}$$

$$U_*^s = min_{U_p^s}\{dist^s(U_p^s, GT^s)\}, \tag{4.2}$$

where the distance at block level is defined as the maximum difference at pixel level —pixel locations $(x, y) \in s$—. Secondly, AE measure with threshold error of 20 —explained in section 2.4— between the best TB built and the GT is estimated. The method [3] is chosen for comparison in the clustering task as it is the best state-of-the-art approach —as will be shown in the following sections—.

4.2.2 Seeds selection method evaluation

In this subsection the quality of the proposed seed selection technique (subsection 3.3.1) in comparison with the one proposed in [3] is presented. In [3], the seeds location(s) are selected where the highest amount of blocks are grouped in a cluster, and then that location(s) is(are) reconstructed with that cluster. As shown in Figure 4.3, the proposed method is able to initialize

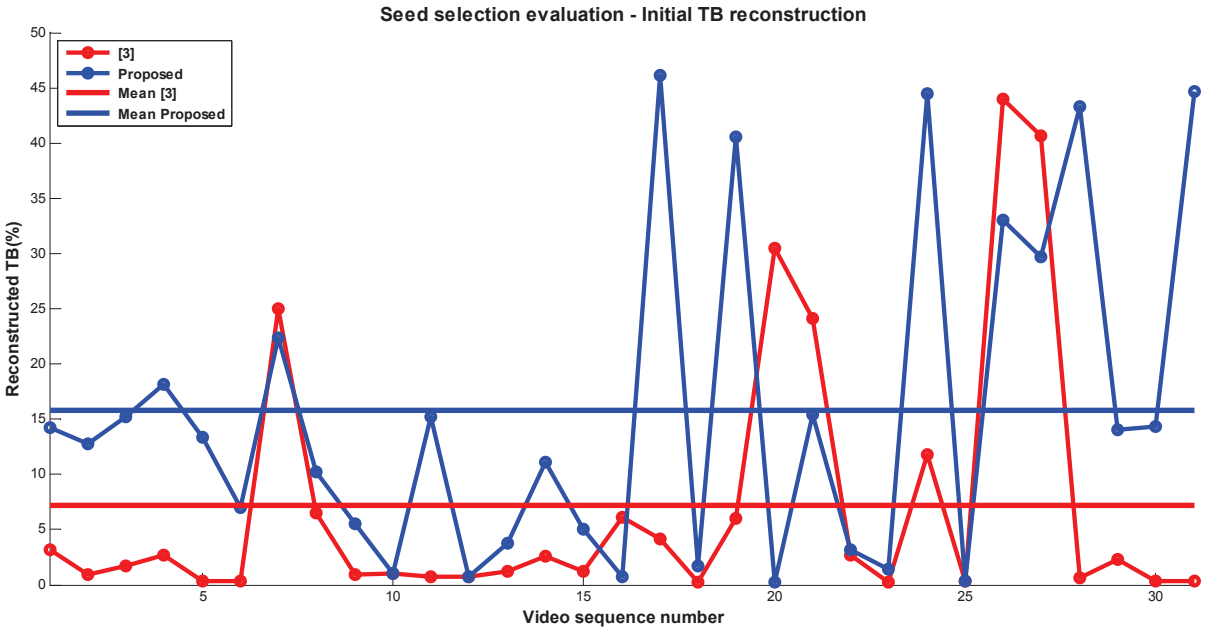

Figure 4.3: Seed selection method evaluation: Percentage of reconstructed True Background. The x-axis denotes the number of the video sequence referenced in Table 4.1. The average percentage of initialized TB with the seed selection method is of 7.21% for [3] and 15.77% for the proposed method.

a higher percentage of the TB —15.77%— than [3] —7.21%— while keeping the correct selection of initial TB blocks (see Figure 4.4, where AE is measured with an error threshold of 20). The fact of starting with a higher amount of initialized TB yields to more information for the iterative reconstruction and therefore less background blocks have to be initialized. As shown in Figure 4.4, high errors appears in sequences 28, 29 and 30 due to differences in the illumination between selected blocks an the GT. Therefore, a different thresholding in the computation of AE could reduce them.

4.2.3 Performance evaluation against State-of-the-art approaches

To evaluate the performance of the proposed approach, several approaches from the literature have been selected according to their code availability and performance results. The selected approaches can be classified into two categories:

- Background initialization approaches: DCT-1 [17] (own implementation), DCT-2 [3] (authors implementation), RDT [9](own implementation), PBI [2] (own implementation),

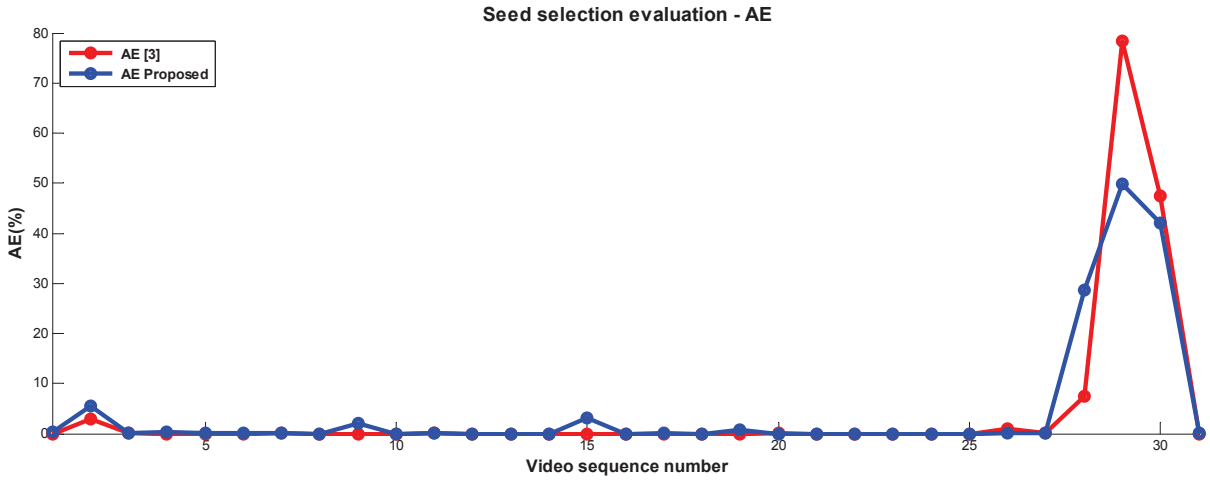

Figure 4.4: Seed selection method evaluation: AE. The x-axis denotes the number of the video sequence referenced in Table 4.1. The figure shows for every sequence the AE measure.

RSM [16] (own implementation) and the Median (MED) initialization used for example in [19]. These approaches are specifically designed for the BGI task.

- Background subtraction approaches: SC-SOBS [11] (authors implementation), SGMM-SOD [25](authors implementation) and FPCP [26](authors implementation). These are background subtraction approaches which provide an online update of the background for each frame. Therefore, the background image at the end of the training sequence has been selected as the result of the algorithms.

It is also important to mention the names to designate the developed proposals in this subsection, P1 and P2. The approach including the initial multipath reconstruction (subsection 3.3.2) is P1 and the proposed approach including the enhancements (subsection 3.3.3) is P2.

The measure selected for evaluation is AE. Nevertheless, instead of selecting a unique threshold as in the evaluations found in the literature, a sweep of the threshold in the interval $[15, 30]$ has been done. The purpose is to show the performance without dependence of a fixed threshold. The mentioned evaluation is shown in Figure 4.5, where P1 and P2 are evaluated against the aforementioned available approaches.

As shown in Figure 4.5, the proposed algorithm —P2— is the best one in the performance evaluation, very close to the algorithm proposed in [3] —DCT-2— . To appreciate better the differences between approaches, the area under the curve (AUC) of the average of the AE for each sequence is shown in Table 4.2. It is important to highlight that although P1 is the third approach in the results ranking, it introduces much more subjective errors —artefacts— than P2 as it can be shown in the results included in Appendix B.

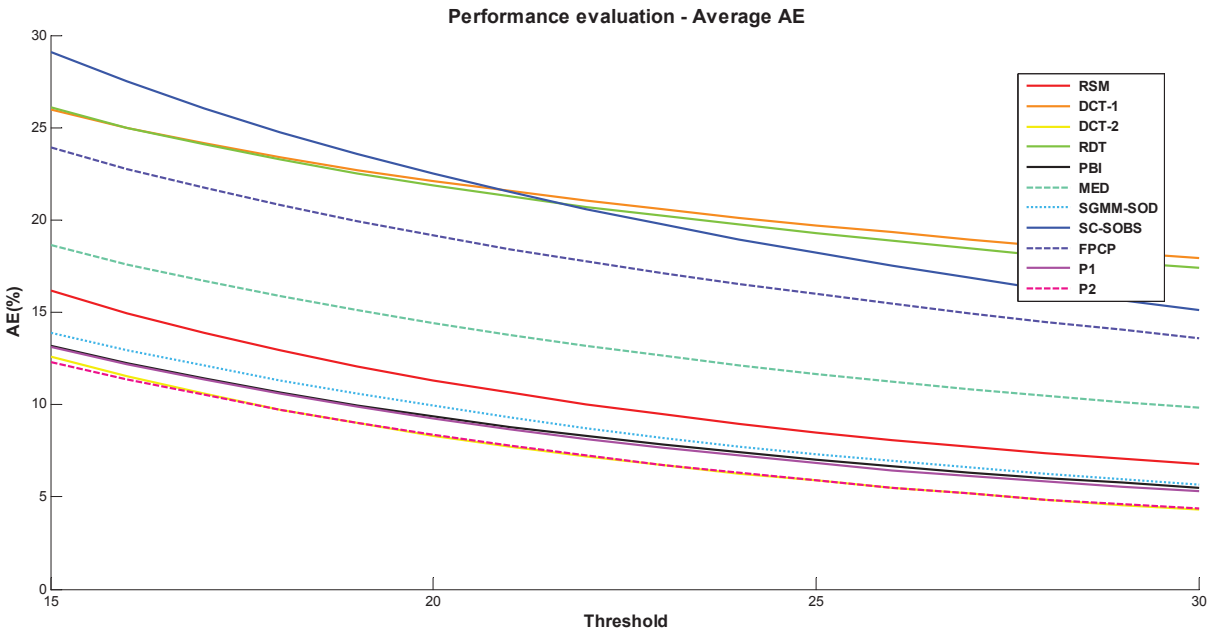

Figure 4.5: Evaluation of proposed approach —P2— against state-of-the-art methods. P1 is the initial proposal without enhancements from subsection 3.3.3. P2 achieves the best performance together with DCT-2 —both lines in the figure are overlapped—.

Approach	AUC
RSM	1.54
DCT-1	3.46
DCT-2	**1.13**
RDT	3.34
PBI	1.23
MED	2.34
SGMM-SOD	1.56
SC-SOBS	3.66
FPCP	2.74
P1	1.18
P2	**1.10**

Table 4.2: AUC of the Average *AE* measure. A lower measure means lower error and therefore a better performance. The table shows that the best approach is the proposed algorithm, very close to the approach DCT-2 from the state-of-the-art. Bold means best results.

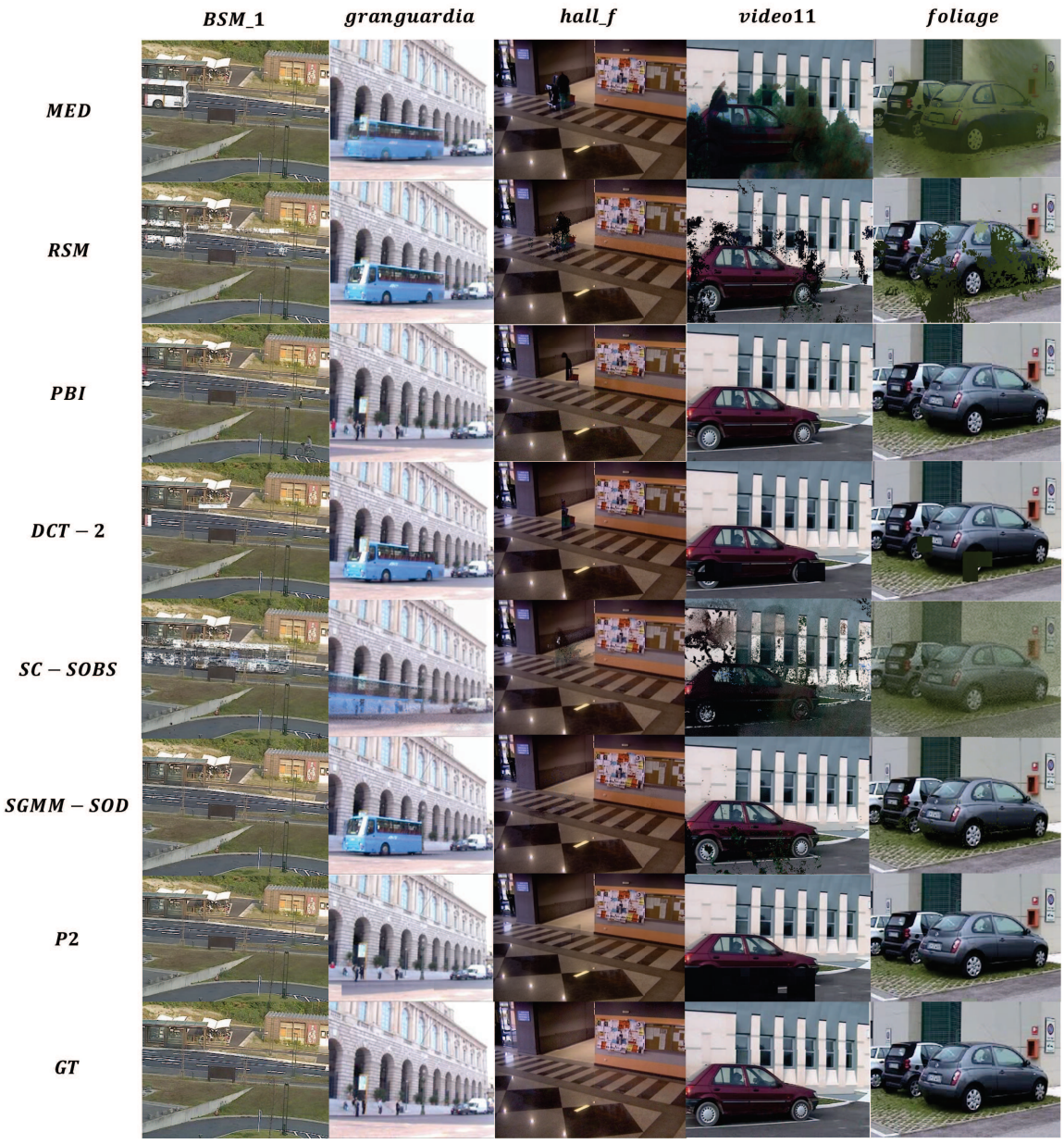

Figure 4.6: Generated background examples. The figure presents the results of 7 algorithms —approaches with lower performance than MED approach are not included— for 5 sequences and the ground-truth (GT) images. From left to right: BSM_1, granguardia and hall_f are examples with high complexity of stationarity solved successfully, while many approaches of the literature fail; video11 and foliage are two examples of high complexity of background visibility, the first one is solved successfully only for PBI and the latter is solved successfully by the proposed approach and several approaches.

In Figure 4.6 five examples of generated TB are given in order to appreciate the behaviour against the stationarity, low visibility and camouflages issues —exposed in section 2.5—. In the

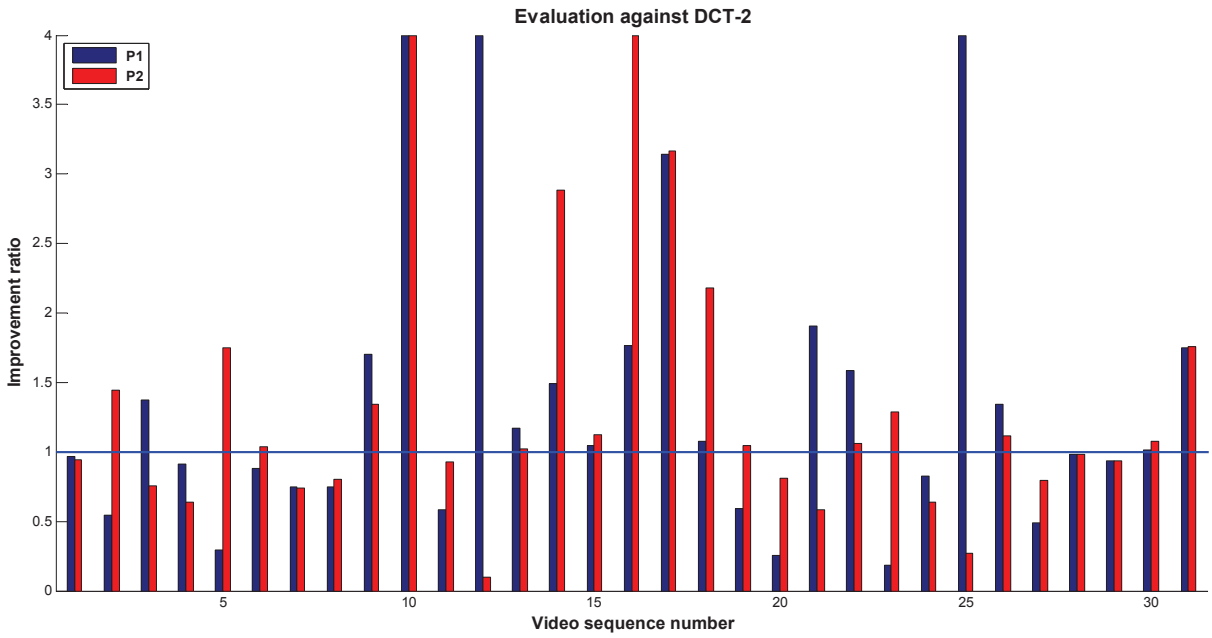

Figure 4.7: Sequence by sequence evaluation of the proposed approach against DCT-2. The x-axis is the number of the video sequence referenced in Table 4.1. If the bar is upper than 1, that means that P1/P2 approach has a better performance —AUC— than DCT-2.

Figure 4.6, a video sequence —video11— where P2 fails is shown. The reason of that error is the combination of several errors: colour continuity (CC) measure fails in one iteration of the reconstruction and subsequently the error is spread in the following ones; the critical situation in which colour similarity is checked —enhancements proposed in subsection 3.3.3— does not come into play due to it is not detected any similar candidate in matter of CC; and erroneous blocks that belong to motion foreground are not filtered in the Motion candidate filtering stage —subsection 3.2.2— due to the camouflage issue between dark foreground and dark background.

Furthermore, a sequence by sequence evaluation is shown in Figure 4.7, where the improvement ratio

$$Improvement\,ratio = \frac{AUC\{DCT-2\}}{AUC\{P2\}}, \tag{4.3}$$

against DCT-2 is shown.

As shown in Figure 4.7, the proposed approach is better than DCT-2 in 17 sequences and worse in 14. In sequence 12 —board— a low performance is obtained due to differences of illumination in the selected candidates in comparison with the GT. In sequence 25 —video11— there is a low performance due to the propagation of and error for several blocks. Furthermore, all the images generated by the algorithms of the Figure 4.7 are included in Appendix B, where

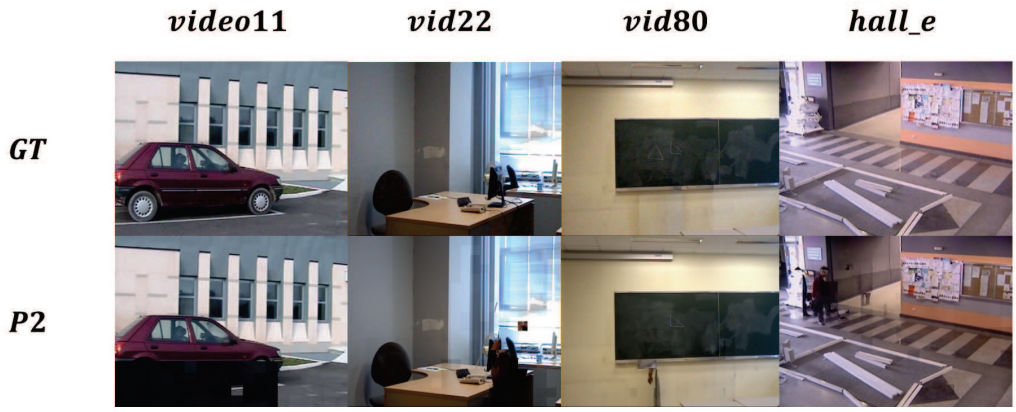

Figure 4.8: Examples of errors of the proposed approach. From left to right: video11, vid22 and vid80 show errors due to the non-compliance of the CC measure and hall_e shows errors due to the non visualization of the foreground in several regions at the selected block size.

it can be shown the good visual initialization achieved for most of the sequences. Additionally, the error measured with AE with four different thresholds is also presented in Appendix B in order to provide percentages of the error in each sequence instead of an improvement ratio.

Finally, it is interesting to describe the current errors of the algorithm. The source of errors is twofold: errors due to failures of CC measure and errors due to the non visualization of the background with the block size selected of $16x16$. Related to the first issue, an erroneous block could be selected if the seed re-selection or colour similarity enhancements —introduced in subsection 3.3.3— do not come into play. Related to the second issue, the non visualization of the background due to the selected block size could introduce errors in crowded environments where the TB is hardly disclosed at pixel level. Figure 4.8 present some examples of the commented issues.

Chapter 5

Conclusions and future work

5.1 Conclusions

In this Master Thesis, we have developed a Background Initialization (BGI) algorithm able to reconstruct the True Background (TB) of a video sequence and providing a notable robustness against the stationary objects and low background visibility issues via the spatial analysis performed.

First, a study of the previous approaches developed for BGI was done. From this study, the main approaches, the strategies followed and the limitations of the BGI task were extracted. Furthermore, at this point we compiled several sequences to build a dataset covering different complexities in order to evaluate the available state-of-the-art algorithms. From this initial evaluation, we realized that the Iterative Model Completion (IMC) alternative relying on spatio-temporal continuities was a suitable alternative for the BGI task. This initial evaluation was done due to the lack of comparative evaluations among the state-of-the-art approaches.

Once the literature of BGI was studied, we started building a new algorithm conformed by two stages: temporal analysis and spatial analysis. The temporal stage built is strongly based on previous work [21] of the VPU Lab and our contribution here is to use motion information to discard blocks aiming to reduce candidates in the agglomerative hierarchical clustering and therefore possible errors in the spatial stage.

On the other hand, the spatial stage is entirely developed in this Master Thesis. First, a seed selection scheme or initial reconstruction of the TB was proposed to increment the percentage of correctly initialized TB with respect to the literature methods in order to maximize the information available in the following iterative spatial reconstruction. Secondly, an iterative multipath spatial reconstruction based on the background smoothness idea was proposed. The motivation of a multipath scheme is to overcome errors of the continuity measure used, through arriving at each spatial location by different paths. Finally, the issues found in the multipath scheme —the external edges of a local neighbourhood are not analysed— were palliated via

seed re-selection, intra-block homogeneity and inter-block colour similarity in certain situations. Therefore, the proposed approach was built enforcing the background smoothness.

To perform the experimental results, the compiled dataset was used. Three evaluations were made: the clustering method —to check that no errors were introduced—, the seed selection method —to check the increment of correct initial TB blocks— and a comparative evaluation of the proposed approach against the state-of-the-art —to proof the quality of our approach—. For the latter evaluation, experiments based on a sweep of threshold values for the AE measure was proposed —different from the state-of-the-art approaches— in order to palliate the effect of a threshold value. Nevertheless, for some sequences the illumination changes due to shadows in the scenes induce errors due to differences between the ground-truth and the estimated TB.

As general conclusions from the algorithm, it is important to mention some drawbacks or limitations of the proposed approach. Taking more advantage of the available information to begin the spatial reconstruction, we could improve the difference with the approaches from the literature. It is also important to comment that although the multipath scheme works fast, the proposed approach is slow due to the seed re-selection enhancement. Moreover, there still being some unresolved issues of the continuity measure, due to the non compliance of the idea of correct TB candidates being the ones with best colour continuity in the edges. This latter problem can easily lead to the propagation of errors, thus decreasing the performance. Also some errors arise due to the non visualization of the background with the block size selected of $16x16$.

5.2 Future work

The BGI task is a labour with many room for improvement as no algorithm is able to behave well in every situation. In the case of IMC approaches —that is the area where we have deepened due to a better performance— this occurs due to two reasons: the reconstruction scheme proposed and the continuity measures used. The following ideas for future work are mostly related with this thought.

Relative to our approach, the issue that must be tackled is the fixed division into blocks. Having a fixed division could lead to the situation of adjacent block locations where the best edge colour continuity criteria to select among the candidates is not correct. Therefore, a multi-resolution scheme where different reconstructions are performed and combined could handle this problem. Furthermore, a proper multi-resolution scheme could lead to avoid the problem of non visualization of the background due to a large block size.

On the other hand, the initial multipath scheme could be redefined to tackle the external edge issue via some kind of local iterative reconstruction —as in DCT-2 [3]—, in order to check possible errors in the reconstructed paths and therefore reducing errors before the combination

of the 8 reconstructions.

Attending to the above mentioned lines of enhancements, simple measures as colour continuity could be used if the reconstruction scheme is enough complex to cover all limitations. Therefore, complex measures like DCT of macro-blocks used in [3], [9] and [17] could be avoided —which is very likely to avoid as smooth response of the DCT in a macro-block does no necessarily means best continuity—.

Additionally to the above mentioned lines of investigation, the proposed initial evaluation framework could be used as a starting point to define an evaluation framework which is a lack in the literature. Furthermore, it could be interesting for future approaches to include in the evaluation full-reference —a ground-truth image is necessary for comparison— image quality assessment measures non dependent of the illumination in order to avoid the threshold issue of the measures used.

Bibliography

[1] C. Chia-Chih and J.K. Aggarwal. An adaptive background model initialization algorithm with objects moving at different depths. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2664–2667, Oct 2008. i, 5, 6, 9, 10, 13, 23

[2] A. Colombari and A. Fusiello. Patch-based background initialization in heavily cluttered video. *Image Processing, IEEE Transactions on*, 19(4):926–933, April 2010. i, 5, 7, 9, 11, 27, 35, 40

[3] V. Reddy, C. Sanderson, and B.C. Lovell. A low-complexity algorithm for static background estimation from cluttered image sequences in surveillance contexts. *J. Image Video Process.*, 2011:1–14, January 2011. i, 5, 8, 9, 12, 15, 20, 23, 27, 38, 39, 40, 41, 48, 49

[4] X. Xun and T.S. Huang. A loopy belief propagation approach for robust background estimation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7, June 2008. i, 5, 8, 9, 13, 15

[5] D. Park and H. Byun. A unified approach to background adaptation and initialization in public scenes. *Pattern Recognition*, 46(7):1985 – 1997, 2013. i, 5, 9, 14, 15

[6] T. Bouwmans. Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review*, 11 - 12(0):31 – 66, 2014. 1, 2, 5

[7] R. Zhang, W. Gong, A. Yaworski, and M. Greenspan. Nonparametric on-line background generation for surveillance video. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1177–1180, Nov 2012. 2, 5, 6, 9

[8] L. Maddalena and A. Petrosino. chapter Background Model Initialization for Static Cameras. 2014. 5, 6

[9] D. Baltieri, R. Vezzani, and R. Cucchiara. Fast background initialization with recursive hadamard transform. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 165–171, Aug 2010. 5, 7, 9, 15, 27, 40, 49

[10] H. Hsiao and J. Leou. Background initialization and foreground segmentation for bootstrapping video sequences. *EURASIP J. Image and Video Processing*, 12, 2013. 5, 8, 9, 15

[11] L. Maddalena and A Petrosino. The sobs algorithm: What are the limits? In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 21–26, June 2012. 5, 6, 9, 41

[12] X. Chen, Y. Shen, and Y.H. Yang. Background estimation using graph cuts and inpainting. In *Proceedings of Graphics Interface 2010*, GI '10, pages 97–103, 2010. 5, 8, 9, 15

[13] R.V.H.M. Colque and G. Camara-Chavez. Progressive background image generation of surveillance traffic videos based on a temporal histogram ruled by a reward/penalty function. In *Graphics, Patterns and Images (Sibgrapi), 2011 24th SIBGRAPI Conference on*, pages 297–304, Aug 2011. 5, 15

[14] T. Crivelli, P. Bouthemy, B. Cernuschi-Frías, and Jian-feng Yao. Simultaneous motion detection and background reconstruction with a conditional mixed-state markov random field. *International Journal of Computer Vision*, 94(3):295–316, 2011. 5, 8, 9, 15

[15] D. Gutchess, M. Trajkovics, E. Cohen-Solal, D. Lyons, and A K. Jain. A background model initialization algorithm for video surveillance. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 733–740, 2001. 5, 7, 9, 11, 13, 15

[16] H. Wang and D. Suter. A novel robust statistical method for background initialization and visual surveillance. In *Computer Vision - ACCV 2006*, volume 3851, pages 328–337. 2006. 5, 6, 9, 13, 15, 41

[17] V. Reddy, C. Sanderson, and B.C. Lovell. An efficient and robust sequential algorithm for background estimation in video surveillance. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 1109–1112, Nov 2009. 5, 7, 8, 9, 12, 13, 15, 27, 40, 49

[18] C. Guo, S. Gao, and D. Zhang. Belief propagation algorithm for background estimation based on local maximum weight matching. In *Image and Signal Processing (CISP), 2012 5th International Congress on*, pages 82–85, Oct 2012. 5, 9, 15

[19] L. Maddalena and A. Petrosino. The 3dsobs+ algorithm for moving object detection. *Computer Vision and Image Understanding*, 122:65 – 73, 2014. 6, 9, 41

[20] W. Long and Y. Yang. Stationary background generation: An alternative to the difference of two images. *Pattern Recognition*, 23(12):1351 – 1359, 1990. 7

[21] J.C. SanMiguel A. Muñoz. Generación de fondo de escena en secuencias de vídeo-seguridad. Master's thesis, 2012. 20, 21, 22, 23, 25, 47

[22] J. Kapur, P. Sahoo, and A. Wong. A new method for graylevel picture thresholding using the entropy of the histogram. *Computer Graph and Image Process*, 29(3):273–285, 1985. 22, 27

[23] G.e Karypis, E. Han, and V. Kumar. Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8):68–75, 1999. 23

[24] K. Wang, B. Wang, and L. Peng. Cvap: Validation for cluster analyses. *Data Science Journal*, 8:88–93, 2009. 25

[25] R.H. Evangelio, M. Patzold, I Keller, and T. Sikora. Adaptively splitted gmm with feedback improvement for the task of background subtraction. *Information Forensics and Security, IEEE Transactions on*, 9(5):863–874, May 2014. 41

[26] P. Rodriguez and B. Wohlberg. Fast principal component pursuit via alternating minimization. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 69–73, Sept 2013. 41

Appendix A

Details of the source of the dataset sequences

As mentioned in section 4.1, the sequences used in the dataset are built using parts of the original ones. Details are given below:

- AB_E: AVSS_AB_Easy sequence from AVSS2007 dataset bounded from frames 251 to 651.
- AB_H: AVSS_AB_Hard sequence from AVSS2007 dataset bounded from frames 250 to 650.
- AB_M: AVSS_AB_Medium sequence from AVSS2007 dataset bounded from frames 2400 to 2800.
- PV_E: AVSS_AB_Medium sequence from AVSS2007 dataset bounded from frames 251 to 651.
- BSM_1: Bus Stop in the morning sequence from LIMU dataset bounded from frames 1870 to 2270.
- BSM_2: Bus Stop in the morning sequence from LIMU dataset bounded from frames 4580 to 4630.
- Inter: Intersection sequence from LIMU dataset bounded from 1275 a 1675.
- Park_a: PETS2009_S2_L2_view0001 sequence from PETS2009 dataset bounded from 1 to 339.
- Park_b: PETS2009_S2_L3_view0001 sequence from PETS2009 dataset.

- snellen: snellen sequence from PBI dataset.
- LGW: TRECVID_LGW_20071123_E1_CAM1 sequence from TRECVID bounded from frames 15400 to 15800.
- board: board sequence from PBI dataset.
- bootstrap: bootstrap sequence from Wallflower dataset.
- ca_vignal: ca_vignal sequence from PBI dataset.
- cam4: camara4 sequence from TRECVID dataset.
- foliage: foliage sequence from PBI dataset.
- granguardia: granguardia sequence from PBI dataset bounded from frames 1 to 400.
- vid16: vid16 sequence from LIRIS2012 dataset.
- vid22: vid22 sequence from LIRIS2012 dataset.
- vid36: vid36 sequence from LIRIS2012 dataset.
- vid44: vid44 sequence from LIRIS2012 dataset.
- vid62: vid62 sequence from LIRIS2012 dataset.
- vid8: vid8 sequence from LIRIS2012 dataset.
- vid80: vid80 sequence from LIRIS2012 dataset bounded from frames 1 to 400.
- video11: video11 sequence from PBI dataset.
- hall_a: sequence extracted from own video sequences not belonging to any dataset. Video_26-01-2013_12-13-08 frames 2200 to 2600.
- hall_b: sequence extracted from own video sequences not belonging to any dataset. Video_26-01-2013_12-13-08 frames 6640 to 6860.
- hall_c: sequence extracted from own video sequences not belonging to any dataset. Video_26-01-2013_12-18-08 frames 2660 to 3160.
- hall_d: sequence extracted from own video sequences not belonging to any dataset. Video_15-01-2013_12-54-54 frames 1 to 400.
- hall_e: sequence extracted from own video sequences not belonging to any dataset. Video_15-01-2013_12-54-54 frames 2440 to 2840.

- hall_f: sequence extracted from own video sequences not belonging to any dataset. Video_26-01-2013_13-58-09_frames 2800 to 3200.

Appendix B

Complete results of DCT-2 and proposals

In the Figures B.1, B.2, B.3 and B.4 GT and results of the TB generated by DCT-2 (subsection 2.3.3), P1 (approach from subsection 3.3.2) and the proposed approach P2 (approach from subsection 3.3.3) are shown. Furthermore, it is also interesting to appreciate the errors of each sequence not only in an improvement against the state-of-the art view or in a global *AE* view as done in chapter 4. To this end, we present in Table B.1 the performance of DCT-2 and the proposals —measured with *AE* for four different thresholds— for all the video sequences. In the mentioned table it is appreciated that the error in most of the sequences is very small, except in sequences hall_c, hall_d and hall_e where there is a different illumination between the GT and the results obtained due to the shadows that arise in the scenes. Having a small error does not necessary mean having a good performance, as usually the difficult areas of the background to initialize suppose a small percentage of the whole image. For example, sequences hall_f and granguardia have a low *AE* for the approach DCT-2 while the critical areas of the stationary objects are wrongly initialized (see Figures B.2 and B.4). Nevertheless, P2 and DCT-2 achieve very good results in the proposed dataset, as shown in subsection 4.2.3 and in this appendix.

A E												
Threshold	**15**			**20**			**25**			**30**		
Approach	**DCT-2**	**P1**	**P2**	**DCT-2**	**P1**	**P2**	**DCT-2**	**P1**	**P2**	**DCT-2**	**P1**	**P2**
Video												
AB_E	**6.71**	6.85	6.84	**5.14**	5.40	5.48	**4.61**	4.71	4.84	**4.29**	4.39	4.54
AB_H	8.61	13.16	**6.68**	5.84	10.43	**4.11**	4.41	8.92	**2.87**	3.54	7.93	**2.14**
AB_M	1.91	**1.40**	1.99	**1.04**	0.83	1.44	0.85	**0.58**	1.20	0.75	**0.42**	1.04
PV_E	**2.30**	2.52	3.32	**1.27**	1.39	1.98	**0.88**	0.96	1.44	**0.66**	0.73	1.12
BSM_1	2.92	7.28	**2.73**	1.58	5.35	**0.97**	1.17	4.57	**0.39**	1.00	4.15	**0.17**
BSM_2	3.67	3.83	**3.41**	1.24	1.46	**1.22**	**0.48**	0.58	**0.48**	0.20	0.26	**0.19**
Inter	**0.23**	0.26	0.26	**0.07**	0.10	0.10	0.02	**0.04**	**0.04**	**0.00**	0.01	0.02
Park_a	**4.3**	6.08	6.02	**2.63**	3.53	3.22	**1.76**	2.26	2.06	**1.30**	1.61	1.55
Park_b	24.83	**10.43**	14.22	10.09	**5.91**	7.59	5.06	**4.00**	4.62	**2.81**	2.83	3.54
snellen	2.63	**0.03**	0.94	1.49	**0.00**	0.15	1.07	**0.00**	0.03	0.71	**0.00**	**0.00**
LGW	**6.17**	9.98	8.32	**3.60**	5.94	3.83	2.69	4.89	**2.63**	2.27	4.36	**2.09**
board	1.52	**0.36**	11.15	0.53	**0.09**	5.36	0.08	**0.05**	1.92	0.04	**0.01**	0.63
bootstrap	4.64	**4.29**	4.85	2.87	**2.40**	2.83	1.88	**1.53**	1.73	1.22	**1.06**	1.10
ca_vignal	**0.03**	0.04	**0.03**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
cam4	15.83	15.20	**14.43**	10.87	10.27	**9.59**	8.13	7.76	**7.21**	6.37	6.16	**5.60**
foliage	2.69	1.68	**0.33**	2.32	1.31	**0.07**	2.06	1.14	**0.02**	1.83	1.00	**0.00**
granguardia	12.83	4.45	**4.33**	11.54	3.72	**3.69**	10.64	3.30	3.29	9.75	**2.95**	2.97
vid16	5.53	4.98	**3.50**	3.11	2.84	**1.51**	1.75	1.64	**0.49**	0.90	1.08	**0.11**
vid22	14.16	14.97	**10.03**	4.63	8.00	**4.46**	**2.50**	5.97	3.12	**1.81**	5.33	2.64
vid36	1.16	2.00	**0.78**	**0.53**	1.80	0.59	**0.25**	1.70	0.51	**0.14**	1.62	0.44
vid44	0.46	**0.44**	1.24	0.36	**0.19**	0.69	0.30	**0.11**	0.36	0.24	**0.06**	0.21
vid62	2.37	**0.97**	1.25	0.76	**0.52**	0.78	0.49	**0.37**	0.60	**0.30**	**0.30**	0.50
vid8	5.16	17.65	**4.61**	3.05	15.02	**2.30**	1.84	13.64	**1.37**	1.20	12.61	**0.85**
vid80	2.64	**2.13**	2.71	**1.15**	1.33	1.81	**0.63**	1.09	1.40	**0.50**	0.94	1.10
video11	3.72	**1.16**	13.50	3.46	**0.74**	12.58	3.24	**0.48**	11.87	3.03	**0.37**	11.34
hall_a	11.00	**8.56**	9.52	4.99	**3.63**	4.41	2.22	**1.66**	2.19	1.06	**0.83**	1.15
hall_b	**4.45**	7.00	5.75	**0.68**	1.50	0.80	**0.25**	0.79	0.30	**0.13**	0.57	0.18
hall_c	**69.23**	70.52	70.75	39.43	**42.13**	42.38	22.98	21.57	21.72	12.44	**11.33**	11.47
hall_d	**84.17**	87.75	87.66	**68.38**	73.50	73.58	**52.13**	55.77	55.97	**37.96**	40.70	40.38
hall_e	73.35	70.65	**69.11**	58.10	**56.26**	53.73	44.39	44.65	**41.49**	35.46	36.52	**33.03**
hall_f	15.56	10.23	**10.22**	10.04	5.52	**5.51**	6.03	3.182	**3.16**	3.44	1.90	**1.87**

Table B.1: *AE* measure for each video sequence using different thresholds. Bold denotes best results for each threshold.

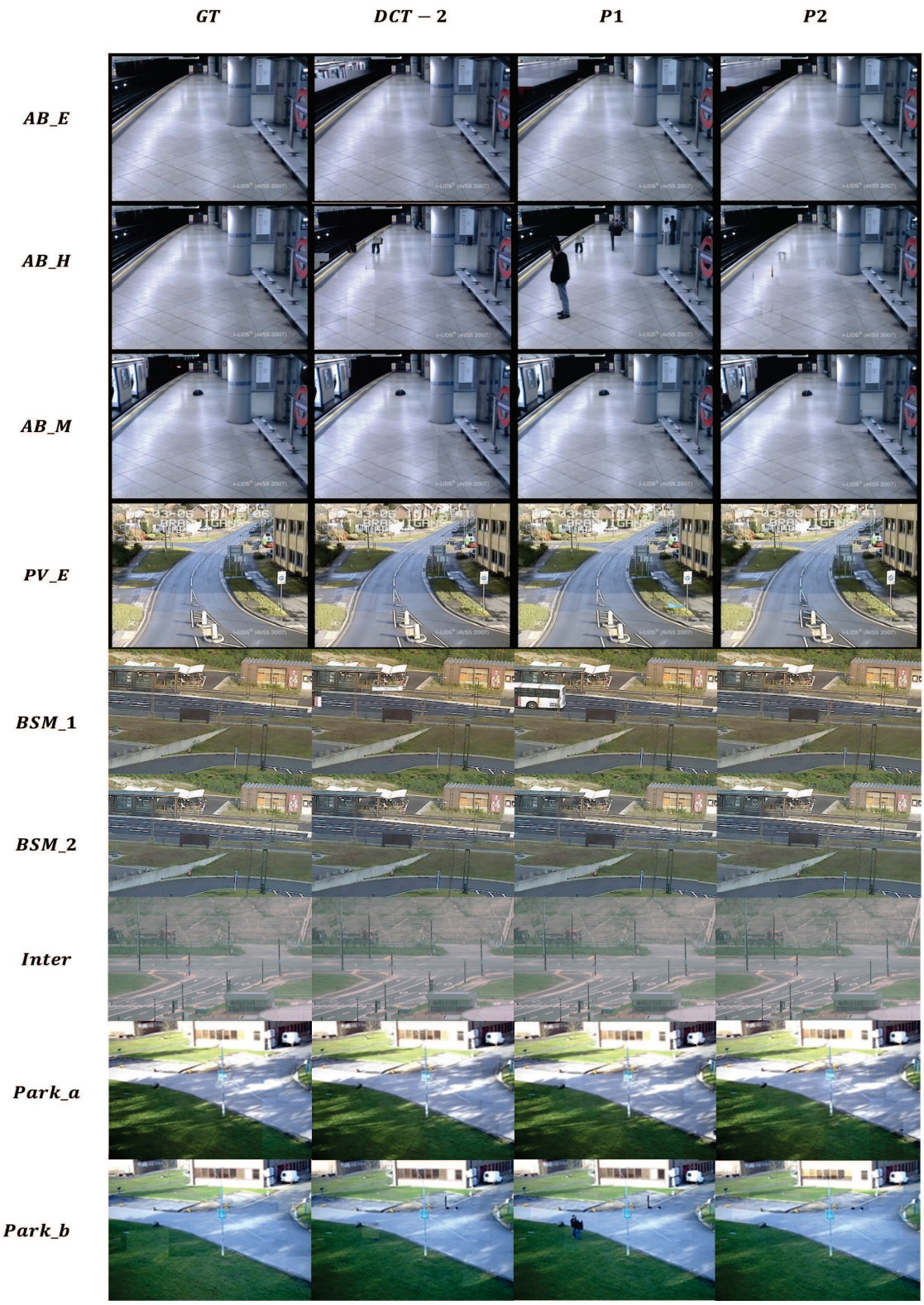

Figure B.1: Ground-truth and results from DCT-2, P1 and the proposed approach P2 (1). From left to right: Ground-truth (GT) image of the TB and generated TB from algorithms DCT-2, P1 and P2.

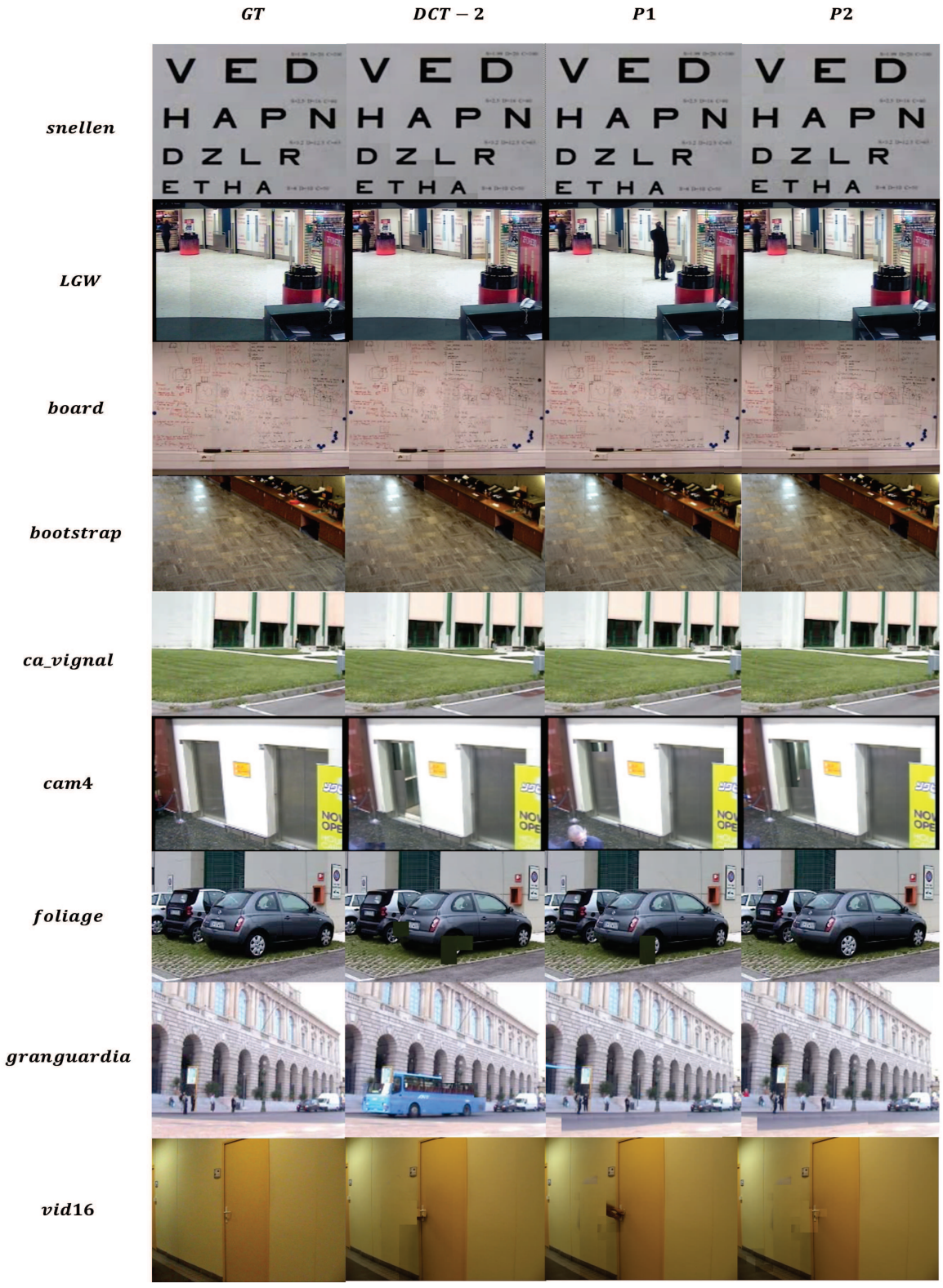

Figure B.2: Ground-truth and results from DCT-2, P1 and the proposed approach P2 (2). From left to right: Ground-truth (GT) image of the TB and generated TB from algorithms DCT-2, P1 and P2.

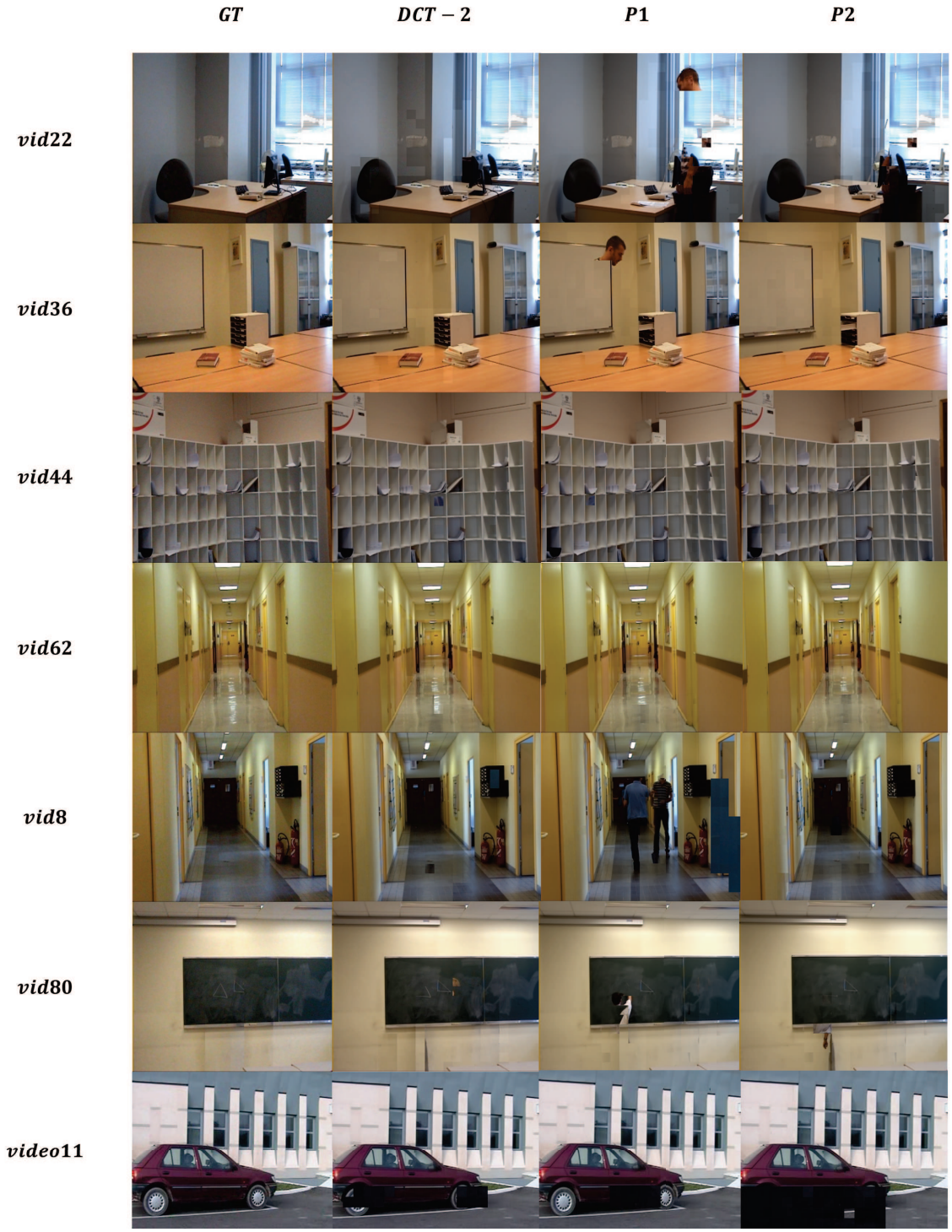

Figure B.3: Ground-truth and results from DCT-2, P1 and the proposed approach P2 (3). From left to right: Ground-truth (GT) image of the TB and generated TB from algorithms DCT-2, P1 and P2.

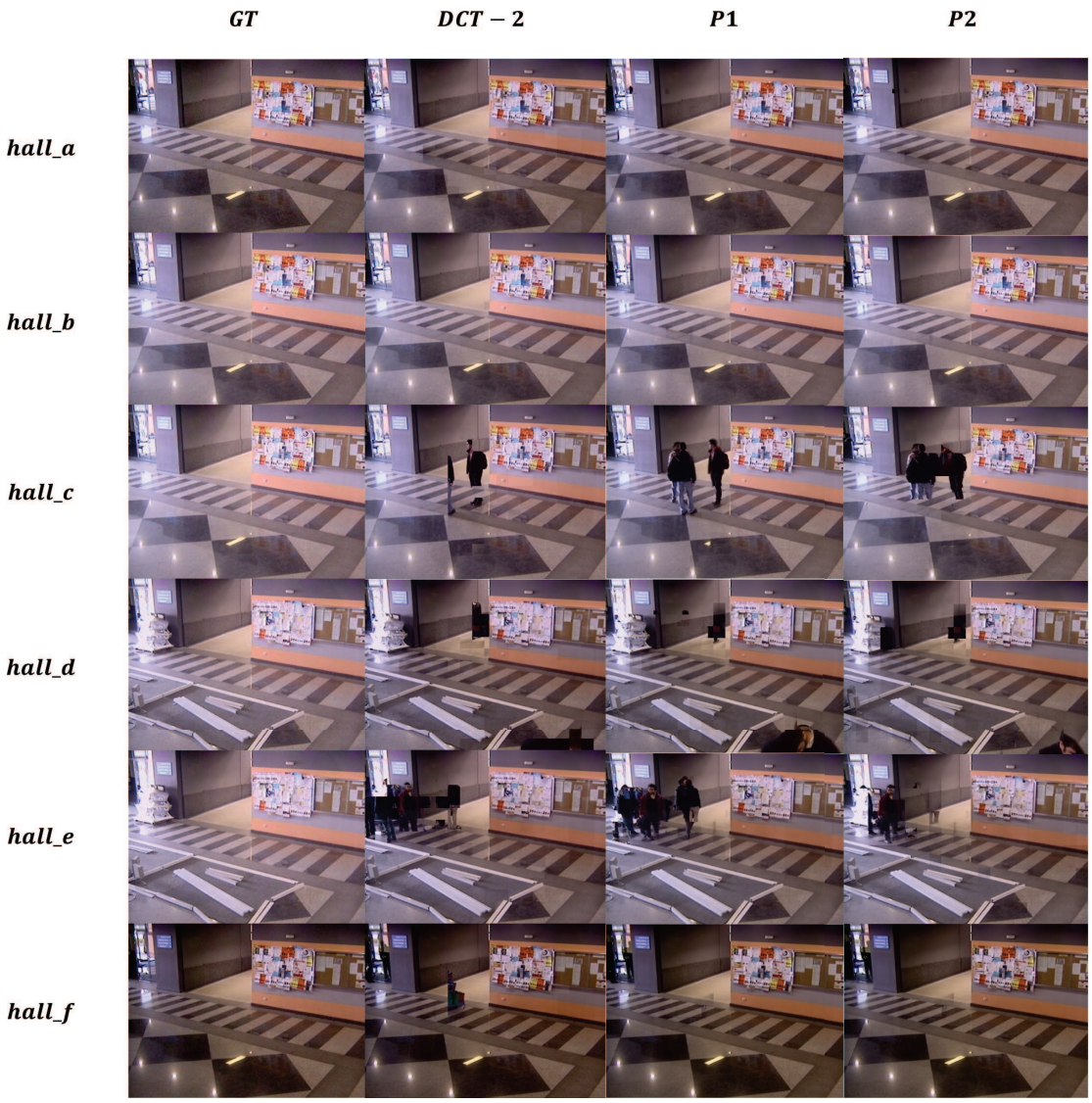

Figure B.4: Ground-truth and results from DCT-2, P1 and the proposed approach P2 (4). From left to right: Ground-truth (GT) image of the TB and generated TB from algorithms DCT-2, P1 and P2.