Rejection based Multipath Reconstruction for Background estimation in Video Sequences with Stationary Objects

Diego Ortego[a,*], Juan C. SanMiguel[a], José M. Martínez[a]

[a] *Video Processing and Understanding Lab, Universidad Autónoma de Madrid, Spain*

Abstract

Background estimation in video consists in extracting a foreground-free image from a set of training frames. Moving and stationary objects may affect the background visibility, thus invalidating the assumption of many related literature where background is the temporal dominant data. In this paper, we present a temporal-spatial block-level approach for background estimation in video to cope with moving and stationary objects. First, a *Temporal Analysis* module obtains a compact representation of the training data by motion filtering and dimensionality reduction. Then, a threshold-free hierarchical clustering determines a set of candidates to represent the background for each spatial location (block). Second, a *Spatial Analysis* module iteratively reconstructs the background using these candidates. For each spatial location, multiple reconstruction hypotheses (paths) are explored to obtain its neighboring locations by enforcing inter-block similarities and intra-block homogeneity constraints in terms of color discontinuity, color dissimilarity and variability. The experimental results show that the proposed approach outperforms the related state-of-the-art over challenging video sequences in presence of moving and stationary objects.

Keywords: Background estimation, Stationary foreground, Background visibility, Clustering, Smoothness, Multipath

*Corresponding author

Email addresses: diego.ortego@uam.es (Diego Ortego), juancarlos.sanmiguel@uam.es (Juan C. SanMiguel), josem.martinez@uam.es (José M. Martínez)

1. Introduction

Segregating relevant moving objects is widely used in several applications of image processing and computer vision. This task often requires to estimate a foreground-free image (or background) under several visual challenges such as in Background Subtraction algorithms [1][2]. Background estimation (BE) finds applications not only in moving object segregation from video sequences [3] but also to represent redundancy in video compression [4], to repair deteriorated images for inpainting [5], to implement video-based privacy protection [6] and to obtain object-free images for computational photography [7].

Several state-of-the-art BE approaches easily capture the background by assuming the availability of a set of frames without foreground objects (*training frames*) [1]. This assumption may not be correct in many video-surveillance scenarios (e.g. shopping malls, airports or train stations) where many foreground objects may exist due to crowds and stationary objects, making very challenging the capture of the background. In general, BE faces two problems related with spatio-temporal scene variations: Background visibility and photometric factors. The former occurs when pixels or regions of the background are seen for short periods of time in the training frames (e.g. due to stationary objects or to high-density of moving foreground), thus the predominant temporal data is not the background. The latter affects BE performance by modifying the background (illumination changes) or by affecting to the employed features (shadows and camouflages). The presence of stationary objects is a major limitation in current approaches as background visibility is highly decreased in the training frames.

To overcome the above-mentioned limitations, we propose a block-level BE approach based on a temporal-spatial strategy that reconstructs an object-free background in presence of moving and stationary objects. For each spatial location, a temporal analysis module obtains a number of background candidates (blocks) via motion filtering, dimensionality reduction and threshold-free hierarchical clustering. Then, the spatial analysis module selects the most suitable candidate for each spatial location according to available candidates in neighboring locations. First, the spatial strategy partially approximates the background by setting a number of initial locations (seeds) based on the motion activity along the training frames. Second, an iterative process estimates the remaining background based on inter-block and intra-block smoothness constraints. The experimental work validates the utility of

the proposed approach, outperforming selected approaches in various datasets especially when dealing with stationary objects.

The contribution of the proposed approach is fourfold. First, we propose a threshold-free clustering technique to determine background candidates without requiring parameter tuning to achieve optimal performance [8][9]. Second, we obtain an initial background estimation (seeds selection) containing more data than state-of-the-art approaches [8][10][11] without introducing additional errors. Thus, fewer spatial locations need to be reconstructed, making the proposed approach less prone to estimation errors as compared to related approaches. Third, the iterative reconstruction estimates different hypotheses of the neighboring background at each location and selects one of them, unlike approaches based on single-hypothesis estimations which may have low-accuracy [8][10][11][12]. Fourth, a new performance measure is proposed to avoid the use of a unique threshold [8][10][11].

The paper is organized as follows: Section 2 discusses the related work and Section 3 overviews the proposed approach. Sections 4 and 5 describe the temporal and spatial analysis, respectively. Section 6 shows the experimental work. Finally, Section 7 presents some conclusions.

2. Related work

Different terms are used for BE [8][13]: Bootstrapping [9][14], Background estimation [3][12], Background generation [15][16] or Background reconstruction [17]. Moreover, BE literature can be categorized as [18]: Temporal Statistics, Sub-intervals of Stable Intensity, Iterative Model Completion and Optimal Labeling. In this section, we instead review related approaches focusing on the applied strategy: temporal and spatial. These strategies may use data in a batch or an online fashion, operating at pixel or region (block) level.

Approaches using *temporal* strategies are common in Background Subtraction [18], where the first frame is taken as the background image, which is updated by the successive frames [14][19][20]. Beyond these techniques, Robust Principal Component Analysis (RPCA) [21] models the background image of a video sequence by low-rank subspace analysis while the foreground is represented by the correlated sparse outliers. However, RPCA methods lose the temporal and spatial structure when representing each frame as a column vector, thus limiting the initialization capabilities. EigenBackground (EB)

methods compute a basis of eigenvectors from the training frames to model the background at image [22] or block [23] level. EB methods require a temporal consistency of the background for successful performance where short-term background occlusions are assumed [24]. RPCA and EB methods do not consider multiple basis to account for the range of appearances exhibited in the training frames and the relations between the basis of adjacent spatial locations, thus decreasing their performance in presence of slow-motion or stationary foreground. The temporal median at pixel level is widely used [25][26], but stationary objects for more than 50% of the training frames are included in the background. Motion information can be used to remove foreground objects from the background model such as optical flow [27][28][29] or inter-frame differences [9][15][29]. Temporal continuous stability of pixel intensity is also employed to obtain hypotheses for the background model in each spatial location [27][28][30] where non-continuous intervals are wrongly assumed as different background representations. Therefore, clustering of non-continuous intervals is preferred to address such assumption [8][10][11][12][31]. Furthermore, temporal variability of pixel values is used to keep occluded background values and to avoid wrong model updates with foreground data [3].

Although some approaches only use *temporal* analysis [26][30], a *spatial* analysis is needed in presence of moving and stationary objects since background may no longer be the dominant temporal information in the training frames. Smoothness constraints may be imposed in the background to decide whether new pixels or blocks belong to the background employing features such as color [15]. In [8] and [10], the Discrete Cosine Transform (DCT) is embedded in a Markov Random Field (MRF) framework to introduce smoothness in neighbors while iterative background estimations correct possible errors [8]. Alternatively, DCT can be replaced by the Hadamard transform to decrease computational complexity, which is combined with iterative corrections based on gradient features between candidates and their neighbors [11]. Smoothness can also be cast as finding the best partially-overlapping block between candidates and the already set background locations [12]. Moreover, block-level color and gradient constraints with the neighborhood can be applied to estimate the background [32]. Furthermore, other approaches encode spatial smoothness and temporal information in energy minimization frameworks such as Loopy Belief Propagation [33][34], Graph Cuts [5], Conditional Mixed-State MRFs [17] or dynamic MRFs [3]. Recently, [35] introduces spatial constraints through image segmentation.

Additionally, spatial information also considers optical flow in the neighborhood [28], correcting its density by handling objects moving at different depths [27].

In summary, several BE strategies have been proposed where recent approaches use temporal information and apply smoothness constraints over the estimated background. The main limitation of current approaches involves situations of low background visibility where existing smoothness schemes do not successfully deal with stationary objects.

3. Proposed approach: Overview

The proposed approach performs a temporal-spatial analysis at block level (see Figure 1) over a set of T training frames I_t, $\mathbb{F} = \{I_1 ... I_T\}$, to extract the reconstructed background image B free of moving and stationary objects. First, the *Splitting* module divides each I_t into non-overlapping blocks $R_t^{\mathbf{s}}$ of size $W \times W$, where $\mathbf{s}$ is the bi-dimensional index for the spatial location of each block. Second, the *Temporal Analysis* module creates a number of background candidates $C_l^{\mathbf{s}}$ for each spatial location $\mathbf{s}$, where $l \in \{1 \dots N^{\mathbf{s}}\}$ and $N^{\mathbf{s}} \leq T$ is the number of candidates. The *Temporal Analysis* consists of the *Motion Filtering* stage to discard $R_t^{\mathbf{s}}$ blocks where moving objects exist and the *Dimensionality Reduction* stage to decrease the amount of data analyzed by the *Clustering* stage which obtains a set of background candidates. Finally, the *Spatial Analysis* module reconstructs the background of each spatial location $\mathbf{s}$, partially estimated in the *Seed Selection* stage, by the *Multipath Reconstruction* stage to iteratively fill each spatial location $\mathbf{s}$ with the optimal candidate $C_*^{\mathbf{s}}$ using inter-block and intra-block smoothness constraints. The temporal and spatial analysis modules are described in Section 4 and Section 5, respectively. The key symbols we use in this paper are given in Table 1.

4. Temporal Analysis

The *Temporal Analysis* module generates the background candidates of each spatial location $\mathbf{s}$. It contains three stages (Figure 1): *Motion Filtering*, *Dimensionality Reduction* and *Clustering*.

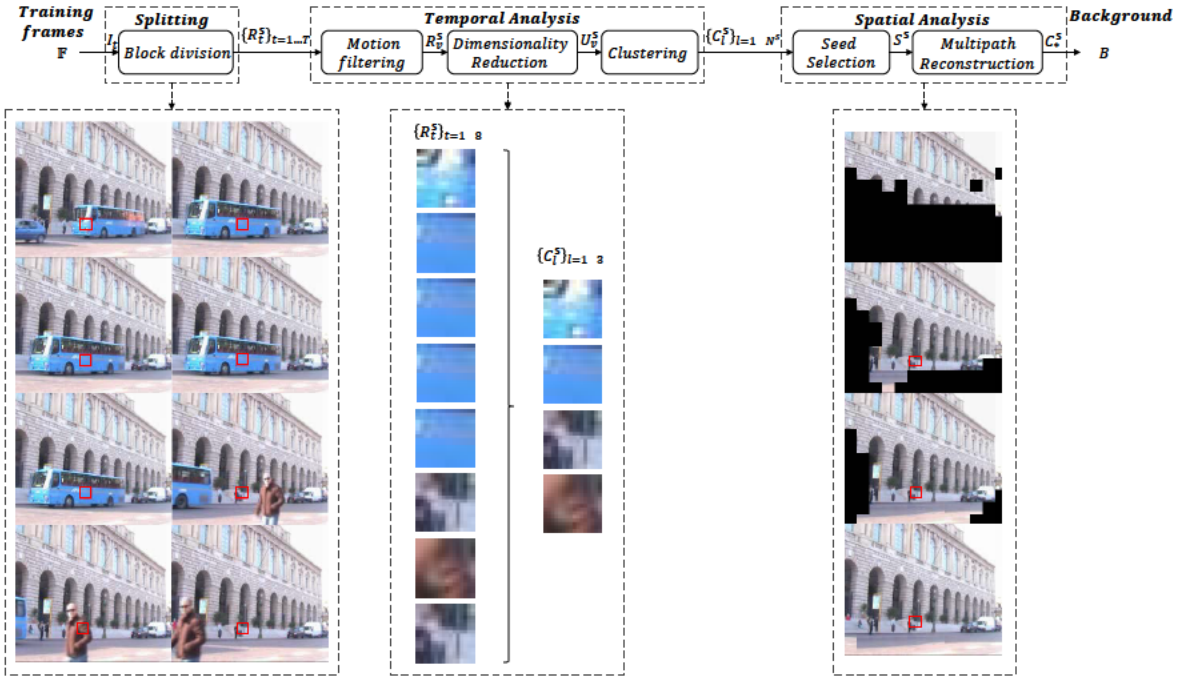


Figure 1: Overview of the proposed multipath approach for temporal-spatial block-level background initialization. Below each module, visual examples are provided for a selected spatial location s (marked in red). First, the *Splitting* module divides into blocks the training frames $\mathbb{F}$ (the selected block is show for the training frames: 28, 109, 190, 191, 192, 354, 371 and 386 of the sequence *guardia*). Second, the *Temporal Analysis* groups all blocks $R_t^{\mathbf{s}}$ extracted from $\mathbb{F}$, thus obtaining background candidates $C_l^{\mathbf{s}}$ via clustering as seen in the visual example (left: $R_t^{\mathbf{s}}$ blocks from previous example, right: $C_l^{\mathbf{s}}$ clusters computed at s). Finally, the *Spatial Analysis* reconstructs the background, starting from some selected seeds $S^{\mathbf{s}}$ and iteratively filling all spatial location s until the whole background is obtained as illustrated in the visual example (From top to bottom: initial selected seeds, two iterations of the multipath reconstruction and the final reconstructed background, where the red rectangle corresponds to the selected candidate $C_*^{\mathbf{s}}$).

4.1. Motion filtering

The *Motion filtering* stage discards $R_t^{\mathbf{s}}$ blocks corresponding to moving objects that cannot be candidates for the reconstructed background B. For all training frames, we compute the motion activity at block level $\lambda_t^{\mathbf{s}}$:

$$\lambda_t^{\mathbf{s}} = \begin{cases} 1 & if \quad \exists \mathbf{p} \in \mathbf{s} : \left| I_t^{\mathbf{p}} - I_{t-k}^{\mathbf{p}} \right| > \eta \\ 0 & otherwise \end{cases}, \tag{1}$$

where $\mathbf{p}$ is the bi-dimensional index for pixel locations in $\mathbf{s}$ and the threshold η is computed automatically [36] to detect intensity changes between $k-$separated frame differences due to moving objects

Table 1: Key symbols and notations

Symbol	Notation
t	Temporal index.
$\mathbf{p}$	Bi-dimensional index for pixel locations.
$\mathbf{s}$	Bi-dimensional index for block locations.
$\mathbb{F}$	Set of T training frames to reconstruct the background image.
I_t	Training frame at time t.
B	Reconstructed background image using $\mathbb{F}$.
$R_t^{\mathbf{s}}$	$W \times W$ block of I_t at time t and location $\mathbf{s}$.
$\lambda_t^{\mathbf{s}}$	Score for block-level activity at location $\mathbf{s}$.
$\mathbb{Y}^{\mathbf{s}}$	Set containing $M^{\mathbf{s}}$ motion-filtered blocks $R_t^{\mathbf{s}}$.
$U_v^{\mathbf{s}}$	PCA-reduced block v at location $\mathbf{s}$, where $v \in [1, M^{\mathbf{s}}]$.
$\mathbb{Z}^{\mathbf{s}}$	Set containing $M^{\mathbf{s}}$ PCA-reduced blocks $U_v^{\mathbf{s}}$.
$N^{\mathbf{s}}$	Number of clusters at location $\mathbf{s}$.
l	Index to denote a cluster at location $\mathbf{s}$, where $l \in [1, N^{\mathbf{s}}]$.
$K_l^{\mathbf{s}}$	Cluster l at location $\mathbf{s}$ that groups $U_t^{\mathbf{s}}$ (i.e. $R_t^{\mathbf{s}}$).
$\mathbb{P}_b^{\mathbf{s}}$	Cluster partition at location $\mathbf{s}$ with b clusters.
$\theta_{SI}(\mathbb{P}_b^{\mathbf{s}})$	Score for cluster partition $\mathbb{P}_b^{\mathbf{s}}$ (Silhouette).
$\theta_{DB}(\mathbb{P}_b^{\mathbf{s}})$	Score for cluster partition $\mathbb{P}_b^{\mathbf{s}}$ (Davies-Bouldin).
$\mathbb{P}_*^{\mathbf{s}}$	Optimal partition at location $\mathbf{s}$. It contains $N^{\mathbf{s}}$ clusters.
$C_l^{\mathbf{s}}$	Candidate to be background (i.e. represents the cluster K_l^s).
$S^{\mathbf{s}}$	Seed block at location $\mathbf{s}$.
$\xi^{\mathbf{s}}$	Activity score to compute seeds at location $\mathbf{s}$.
$\tilde{B}$	Iteratively reconstructed background image. $\tilde{B}$ is initialized with S and contains blocks $\tilde{B}^{\mathbf{s}}$.
$\mathbb{V}_8^{\mathbf{s}}$	8-connected block neighborhood at location $\mathbf{s}$.
$\mathbb{V}_4^{\mathbf{s}}$	4-connected block neighborhood at location $\mathbf{s}$.
$\Phi\left(C_l^{\mathbf{s}'}\right)$	Inter-block color discontinuity for candidate $C_l^{\mathbf{s}'}$.
$\Psi\left(C_l^{\mathbf{s}'}\right)$	Intra-block heterogeneity for candidate $C_l^{\mathbf{s}'}$.
$\Omega\left(C_l^{\mathbf{s}'}\right)$	Inter-block color dissimilarity for candidate $C_l^{\mathbf{s}'}$.
$\tilde{C}_{\Phi}^{\mathbf{s}',m}$	Temporary candidate selected using Φ, at location $\mathbf{s}'$ for path m.
$\tilde{C}_{\Psi}^{\mathbf{s}',m}$	Temporary candidate selected using Ψ, at location $\mathbf{s}'$ for path m.
$\tilde{C}_{\Omega}^{\mathbf{s}',m}$	Temporary candidate selected using Ω, at location $\mathbf{s}'$ for path m
$\tilde{C}^{\mathbf{s}',m}$	Temporary candidate selected at location $\mathbf{s}'$ for path m. Is selected among $\tilde{C}_{\Phi}^{\mathbf{s}',m}$, $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$.
$C_*^{\mathbf{s}'}$	Selected candidate at location $\mathbf{s}'$.
GT	Ground-truth background image that contains blocks $GT^{\mathbf{s}}$.
$B_{best}^{\mathbf{s}}$	Best background selecting at each location $\mathbf{s}$ the blocks $\tilde{B}^{\mathbf{s}}$ with lowest distance to $GT^{\mathbf{s}}$.

(k should be small). $\lambda_t^{\mathbf{s}}$ takes the value 1(0) when motion (no motion) is detected, thus rejecting (keeping) the associated block R_t^s. Note that Eq. 1 implies the visualization of the background for k consecutive frames, as often assumed in existing literature [8][27]. Finally, the selected data to compose the background at each location $\mathbf{s}$ is represented by $\mathbb{Y}^{\mathbf{s}} = \{R_v^{\mathbf{s}}\}_{v=1\dots M^{\mathbf{s}}}$, where $M^{\mathbf{s}}$ is the number of blocks without motion and $M^{\mathbf{s}} \leq T$, $\forall \mathbf{s}$.

4.2. Dimensionality Reduction

To further reduce the data to process, we apply Principal Component Analysis (PCA) [37] to $\mathbb{Y}^{\mathbf{s}}$ as the useful data to generate background candidates is driven by the block variance. Pixel locations with variations over time are relevant to group blocks whereas pixel locations without variability are redundant. PCA determines a transformation basis to project data where pixels with low variance over time are removed. PCA is applied to all blocks in $\mathbb{Y}^{\mathbf{s}}$, where each block is previously rasterized into a column vector of size $3W^2$ by concatenating its RGB channels. Finally, we obtain a matrix $\mathbb{Z}^{\mathbf{s}} = \{U_v^{\mathbf{s}}\}_{v=1\dots M^{\mathbf{s}}}$, where $|U_v^{\mathbf{s}}| \leq |R_v^{\mathbf{s}}|$ and $|\cdot|$ denotes the cardinality, i.e. the number of elements, representing the data in the PCA domain which is used exclusively for the clustering stage (Subsection 4.3). Note that the *Spatial Analysis* module (Section 5) uses the $W \times W$ blocks R_t^s to estimate the background image B instead of the PCA-reduced data $U_v^{\mathbf{s}}$.

4.3. Clustering

This stage generates a number of candidates $C_l^{\mathbf{s}}$ to be the background $B^{\mathbf{s}}$ for each location $\mathbf{s}$. Instead of using the raw data, we group the PCA-reduced data $\mathbb{Z}^{\mathbf{s}}$ into clusters $K_l^{\mathbf{s}}$ which are structured as partitions $\mathbb{P}_{N^{\mathbf{s}}}^{\mathbf{s}} = \{K_1^{\mathbf{s}} \dots K_{N^{\mathbf{s}}}^{\mathbf{s}}\}$ where $N^{\mathbf{s}}$ is the total number of clusters. As the optimum $N^{\mathbf{s}}$ is not known for each $\mathbf{s}$, hypotheses for the partitions are created for different values of $N^{\mathbf{s}}$. The optimal partition is found by validation indexes that maximize inter-cluster differences and intra-cluster similarities. The proposed approach provides a threshold-free clustering that leads to sub-optimal solutions containing the desired candidates. The candidates $C_l^{\mathbf{s}}$ represent each cluster $K_l^{\mathbf{s}}$ where the best candidate $C_*^{\mathbf{s}}$ is selected in the *Spatial Analysis* module (Section 5).

For generating the clusters, we employ agglomerative hierarchical clustering (AHC) [38] over matrices $\mathbb{Z}^{\mathbf{s}}$ where the distance between two clusters is defined as the highest Euclidean distance among

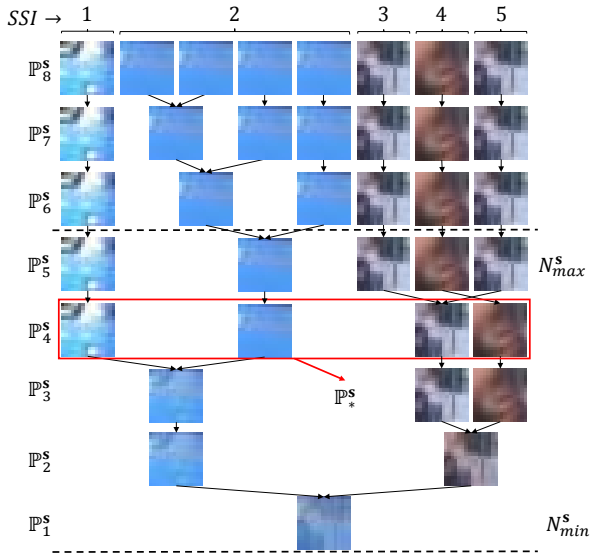


Figure 2: Example of a dendrogram to detect the optimal clustering partition $\mathbb{P}_*^{\mathbf{s}}$ for a 8-block set. Only partitions between $N_{min}^{\mathbf{s}}$ and $N_{max}^{\mathbf{s}}$ are considered (dashed lines). $\mathbb{P}_4^{\mathbf{s}}$ is selected as optimal partition as it has the highest $\theta_{SI}(\mathbb{P}_b^{\mathbf{s}}) + \theta_{DB}(\mathbb{P}_b^{\mathbf{s}})$, thus $N^{\mathbf{s}} = 4$. Albeit clustering uses PCA-reduced blocks $U_v^{\mathbf{s}}$, we show the associated blocks $R_t^{\mathbf{s}}$ for visualization purposes.

members $U_v^{\mathbf{s}}$ of both clusters. The AHC cluster structure can be represented as dendrograms, i.e. tree-like diagrams depicting partition hypotheses at different cluster distances. Thus, we limit the number of clustering hypotheses between a minimum and maximum value ($N_{min}^{\mathbf{s}}$ and $N_{max}^{\mathbf{s}}$, respectively). $N_{min}^{\mathbf{s}}$ is set to 1 (i.e. one cluster) which corresponds to an always-visible background. For each location $\mathbf{s}$, $N_{max}^{\mathbf{s}}$ is set to the number of identified Sub-intervals of Stable Intensity (SSI) [27][28], as SSIs may be caused by objects or background. SSIs are continuous temporal intervals without intensity variations, computed at block level using motion information from Eq. 1. Finally, partition hypotheses $\{\mathbb{P}_b^{\mathbf{s}}\}_{b=N_{min}^{\mathbf{s}},\ldots,N_{max}^{\mathbf{s}}}$ are generated where b is the number of clusters in the partition. Figure 2 shows a dendrogram for clustering eight blocks and an example of SSIs on top of Figure 2, where $N_{max}^{\mathbf{s}} = 5$.

Subsequently, clustering validation determines the best partition $\mathbb{P}_*^{\mathbf{s}}$ containing the optimal number of clusters $N^{\mathbf{s}}$. This validation employs the Silhouette θ_{SI} and Davies-Bouldin θ_{DB} indexes [39]. θ_{SI} measures the compactness and separation among clusters; a higher average value of this measure implies a better quality of the cluster. θ_{DB} measures the similarity between each cluster and its

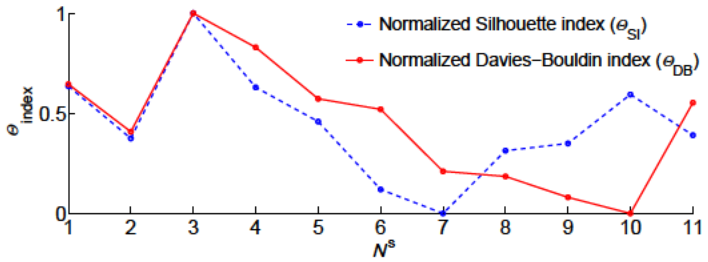


Figure 3: Example of normalized scores for clustering validation. Scores $\theta_{SI}(\mathbb{P}^s_b)$ and $\theta_{DB}(\mathbb{P}^s_b)$ are shown for each partition ranging from $N^s_{min} = 1$ to $N^s_{max} = 11$ clusters. The optimum is $N^s = 3$ as partition $\mathbb{P}^s_3$ obtains the highest $\theta_{SI}(\mathbb{P}^s_b) + \theta_{DB}(\mathbb{P}^s_b)$ score.

highest similar one; small values in this index correspond to compact clusters whose centroid is far from the others. After computing both indexes for each hypothesized partition $\mathbb{P}^s_b$, we normalize them by considering that maximum θ_{SI} and minimum θ_{DB} are preferred:

$$\theta_{SI}(\mathbb{P}^s_b) = \frac{SI(\mathbb{P}^s_b) - min(\mathbb{L})}{max(\mathbb{L}) - min(\mathbb{L})}, \tag{2}$$

$$\theta_{DB}(\mathbb{P}^s_b) = \frac{DB(\mathbb{P}^s_b) - max(\mathbb{M})}{max(\mathbb{M}) - min(\mathbb{M})}, \tag{3}$$

where the sets $\mathbb{L} = \{\theta_{SI}(\mathbb{P}^s_b)\}_{b=N^s_{min},\dots,N^s_{max}}$ and $\mathbb{M} = \{\theta_{DB}(\mathbb{P}^s_b)\}_{b=N^s_{min},\dots,N^s_{max}}$ are all θ_{SI} and θ_{DB} scores, respectively. Then, both scores are combined for each partition $\mathbb{P}^s_b$ to determine the optimal $\mathbb{P}^s_*$:

$$\mathbb{P}^s_* = \underset{b=N^s_{min},\dots,N^s_{max}}{argmax} \left(\theta_{SI}(\mathbb{P}^s_b) + \theta_{DB}(\mathbb{P}^s_b)\right). \tag{4}$$

Figure 3 presents an example of clustering validation with 11 partitions, where the optimal one contains 3 clusters with the highest $\theta_{SI}(\mathbb{P}^s_b) + \theta_{DB}(\mathbb{P}^s_b)$ value.

Finally, we compute each background candidate C^s_l as the average of members in the cluster K^s_l, using the $W \times W$ blocks R^s_t instead of the PCA-reduced data U^s_v, similarly to the widely used K-means clustering [40], which also reduces noise in the final candidate.

5. Spatial Analysis

This module obtains each background block $B^{\mathbf{s}}$ by selecting the best candidate $C_*^{\mathbf{s}}$ among the set of background candidates $C_l^{\mathbf{s}}$. For each location, a multipath reconstruction of the background is proposed to enforce background smoothness among selected candidates in neighboring locations. The reconstruction process is divided in two stages (see Figure 1): *Seed Selection* and *Multipath Reconstruction*. For the latter, the explanation is divided into *Sequential Multipath Reconstruction* (Subsection 5.2) and *Rejection based Multipath Reconstruction* (Subsection 5.3) for readability.

5.1. Seed Selection

An initial partial background estimation is provided for selected locations by seed blocks $S^{\mathbf{s}}$ defined as highly-reliable background candidates. Existing approaches often establish this candidate-seed correspondence for the $\mathbf{s}$ locations with one cluster and, therefore, a unique candidate $C_l^{\mathbf{s}}$ for $B^{\mathbf{s}}$ to be selected [8][10][11]. When these single-candidate clusters do not exist, a *major cluster* $\hat{C}_{l'}^{\mathbf{s}}$ at each spatial location $\mathbf{s}$ can be identified as the cluster with maximum size:

$$\hat{C}_{l'}^{\mathbf{s}} = C_{l'}^{\mathbf{s}} : |K_{l'}^{\mathbf{s}}| > |K_l^{\mathbf{s}}|, \forall l = 1, \ldots, N^{\mathbf{s}}, \tag{5}$$

where major clusters are selected as seeds when their cardinality is equal to the maximum one for all locations $\max_{\mathbf{s}} \{|K_{l'}^{\mathbf{s}}|\}$. However, Eq. (5) initializes few blocks where stationary objects may be temporally dominant and be wrongly selected as seeds. Errors in this initial background estimation are critical since they are propagated in the subsequent stages.

We address such limitation by proposing a unified analysis of stationarity and motion activity along training frames. We detect locations $\mathbf{s}$ with low motion or without stationary objects over time as suitable locations to initialize with seeds. For such detection, we assume that stationary objects occluding the background in I_1 are not going to remain in the same location in I_T. This assumption is reasonable, as objects not moving for all training frames can be considered as background. Hence, an activity score at block level $\xi^{\mathbf{s}}$ is computed as:

$$\xi^{\mathbf{s}} = max\Big\{ f\left(I_1^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_1^{\mathbf{p}}\}\right) + f\left(I_T^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_T^{\mathbf{p}}\}\right) \Big\}_{\forall\, \mathbf{p} \in \mathbf{s}}, \tag{6}$$

where $\mathbf{p}$ is a pixel location; $\mathbb{F}^{\mathbf{p}}$, $I_1^{\mathbf{p}}$ and $I_T^{\mathbf{p}}$ are the gray-level pixel values at location $\mathbf{p}$ of the training sequence, initial frame and final frames, respectively; $\mathbb{F}^{\mathbf{p}} \setminus \{I_1^{\mathbf{p}}\}$ and $\mathbb{F}^{\mathbf{p}} \setminus \{I_T^{\mathbf{p}}\}$ are the set of training frames except the initial and final ones, respectively. The function $f(\cdot,\cdot)$ computes the average value for the absolute pixel-level difference:

$$f\left(I_t^{\mathbf{p}}, \mathbb{I}^{\mathbf{p}}\right) = \frac{1}{|\mathbb{I}^{\mathbf{p}}|} \sum_{q=1}^{|\mathbb{I}^{\mathbf{p}}|} \begin{cases} 1 \; if & \left|I_t^{\mathbf{p}} - I_q^{\mathbf{p}}\right| > \tau \\ 0 & otherwise \end{cases}, \tag{7}$$

where $\mathbb{I}^{\mathbf{p}} = \left\{I_q^{\mathbf{p}}\right\}$ is a generic set of pixels at location $\mathbf{p}$ and τ is a detection threshold computed automatically [36]. The forward activity score $f\left(I_1^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_1^{\mathbf{p}}\}\right)$ compares the pixels of the first frame against the other frames. Similarly, the backward activity score $f\left(I_T^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_T^{\mathbf{p}}\}\right)$ compares the pixels of the last frame against the other frames. Finally, the initial background estimation with seeds $S^{\mathbf{s}}$ is obtained only in locations with minimum $\xi^{\mathbf{s}}$:

$$S^{\mathbf{s}} = \begin{cases} \hat{C}_l^{\mathbf{s}} & if\ \xi^{\mathbf{s}} = min\{\xi^{\mathbf{s}'}\}_{\forall \mathbf{s}' \in I} \\ \emptyset & otherwise \end{cases}, \tag{8}$$

where $\hat{C}_l^{\mathbf{s}}$ is the major cluster and the empty locations $\mathbf{s}$ will be filled by the *Multipath Reconstruction*. Figure 4 presents an example of the activity scores where locations with minimum $\xi^{\mathbf{s}}$ conform the seeds $S^{\mathbf{s}}$. The initial partial background $\tilde{B}$ to be reconstructed is obtained using the seeds, i.e. $\tilde{B}^{\mathbf{s}} = S^{\mathbf{s}}$, $\forall \mathbf{s}$.

5.2. Sequential Multipath Reconstruction

This subsection describes the framework for Sequential Multipath Reconstruction (SMR) to iteratively reconstruct the background from the initial estimation (Eq. 8).

If we consider the location index $\mathbf{s}$ as a bi-dimensional vector (i.e. $B^{\mathbf{s}} \equiv B^{(i,j)}$), the 4-connected neighborhood $\mathbb{V}_4^{\mathbf{s}}$ is defined as:

$$\mathbb{V}_4^{\mathbf{s}} = \left\{B^{(i-1,j)}, B^{(i,j+1)}, B^{(i+1,j)}, B^{(i,j-1)}\right\}, \tag{9}$$

whereas the 8-connected neighborhood $\mathbb{V}_8^{\mathbf{s}}$ is defined as:

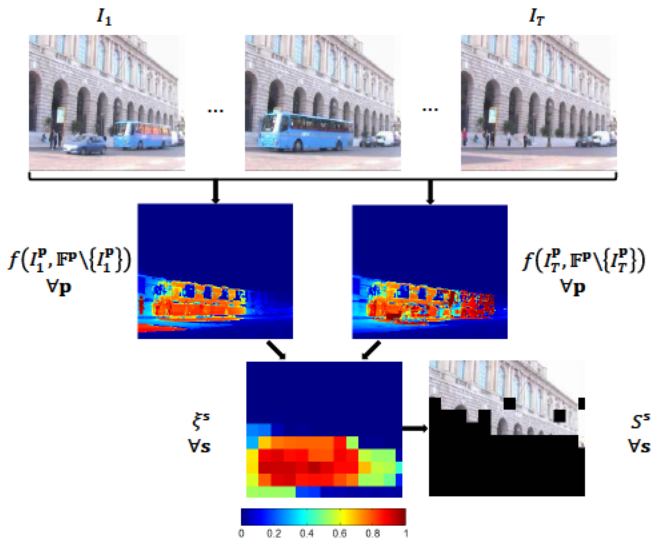


Figure 4: Seed Selection example. From top (set of frames) to bottom ($S^{\mathbf{s}}$) the *Seed Selection* process is presented. Key. $f\left(I_1^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_1^{\mathbf{p}}\}\right)$: forward activity score (pixel level). $f\left(I_T^{\mathbf{p}}, \mathbb{F}^{\mathbf{p}} \setminus \{I_T^{\mathbf{p}}\}\right)$: backward activity score (pixel level). $\xi^{\mathbf{s}}$: activity score (block level). $S^{\mathbf{s}}$: seeds image.

$$\mathbb{V}_8^{\mathbf{s}} = \{B^{(i-1,j)}, B^{(i-1,j+1)}, B^{(i,j+1)}, B^{(i+1,j+1)},$$
$$B^{(i+1,j)}, B^{(i+1,j-1)}, B^{(i,j-1)}, B^{(i-1,j-1)}\}. \quad (10)$$

SMR starts each iteration of the background reconstruction from a partial background $\tilde{B}$ with empty locations. Then, SMR chooses a background block $\tilde{B}^{\mathbf{s}}$ and maximum number of non-empty neighbors in $\mathbb{V}_8^{\mathbf{s}}$, where empty locations are reconstructed by m paths or hypotheses. Each path starts from one side of $\tilde{B}^{\mathbf{s}}$ (top, bottom, left or right), employs a direction (clockwise or counterclockwise) and sequentially fills all empty locations $\mathbf{s}' \in \mathbb{V}_8^{\mathbf{s}}$ with candidates $C_l^{\mathbf{s}'}$. Multi-path reconstruction improves robustness against wrong candidate selections due to objects or other artifacts. Figure 5(a) shows the selected block $\tilde{B}^{\mathbf{s}}$ whose neighborhood $\mathbb{V}_8^{\mathbf{s}}$ is explored using 8 paths traversed as presented in Figure 5(b). Some blocks already exist in each path and therefore, they are not reconstructed. Figure 5(c) presents an example where some $\tilde{B}^{\mathbf{s}}$ neighbors exist.

For each m-path, we select suitable candidates to fill empty locations by employing a fitness function

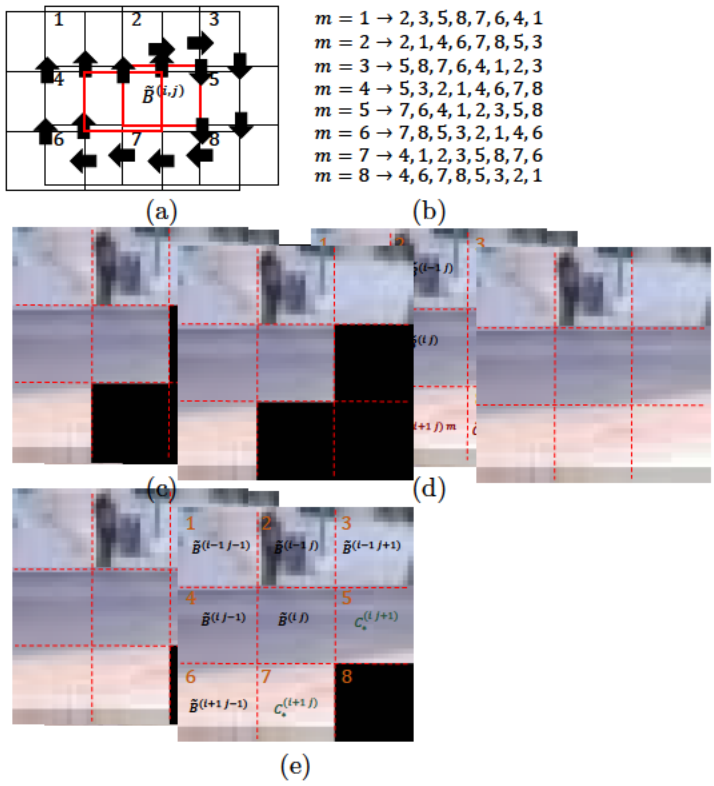
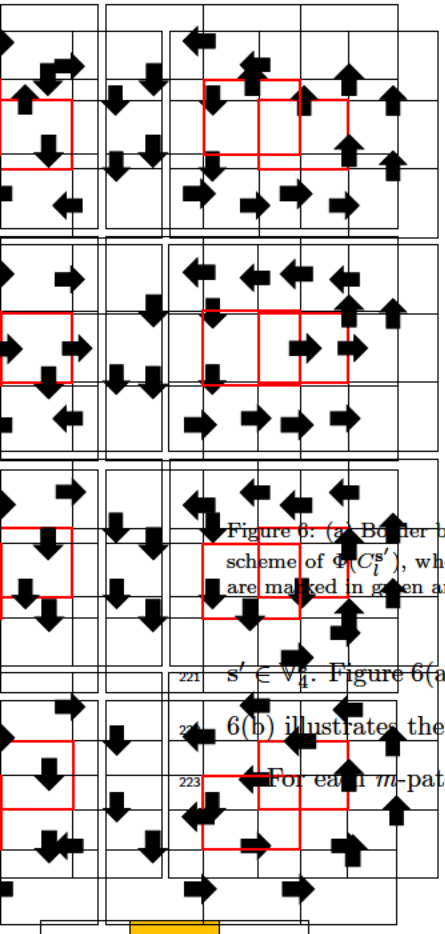


Figure 5: Multipath reconstruction scheme for each iteration of $\tilde{B}^{\mathbf{s}} \equiv \tilde{B}^{(i,j)}$. (a) First path ($m = 1$) to reconstruct $\mathbb{V}_8^{(i,j)}$. Black arrows describe the path direction. (b) Locations explored for the $m = 1 \ldots 8$ paths, which assign a temporary block $\tilde{C}^{\mathbf{s}',m}$ for each empty location $\mathbf{s}' \in \mathbb{V}_8^{(i,j)}$. (c) Example of a seed $\tilde{B}^{(i,j)}$ and its $\mathbb{V}_8^{(i,j)}$. (d) Result of the reconstruction of $\mathbb{V}_8^{(i,j)}$ in (c) using the $m = 1$ path, thus temporary blocks $\tilde{C}^{\mathbf{s}',m}$ are selected (dark red in locations 5, 7 and 8). (e) Final reconstruction of $\mathbb{V}_4^{(i,j)}$ for $\tilde{B}^{(i,j)}$ where the reconstructed blocks $C_*^{s'}$ (dark green in locations 5 and 7) are selected.

Φ based on the inter-block color discontinuity in the neighborhood $\mathbb{V}_4^{\mathbf{s}'}$ of the location to be filled:

$$\Phi\left(C_l^{\mathbf{s}'}\right) = \frac{1}{\left|\mathbb{V}_4^{\mathbf{s}'}\right|} \sum_{\mathbf{s}'' \in \mathbb{V}_4^{\mathbf{s}'}} \left(\frac{1}{W} \sum_{p,p' \in \mathbb{E}} \left| C_l^{\mathbf{s}'}(\mathbf{p}) - \tilde{C}^{\mathbf{s}''}(\mathbf{p}') \right| \right), \tag{11}$$

where the $\tilde{C}^{\mathbf{s}''}$ are the already set neighbors in $\mathbf{s}'' \in \mathbb{V}_4^{\mathbf{s}'}$ (temporary blocks selected during the path reconstruction or previously estimated $\tilde{B}^{\mathbf{s}'}$). $\mathbb{E}$ denotes the set of pixel locations pairs $\mathbf{p}$ and $\mathbf{p}'$ in the border between blocks $C_l^{\mathbf{s}'}$ and $\tilde{C}^{\mathbf{s}''}$, respectively. Therefore, Φ employs 1 to 4 borders depending on the non-empty locations in $\mathbb{V}_4^{\mathbf{s}'}$. Figure 6 shows the $\mathbb{V}_4^{\mathbf{s}'}$ reconstruction scheme using Φ for the location



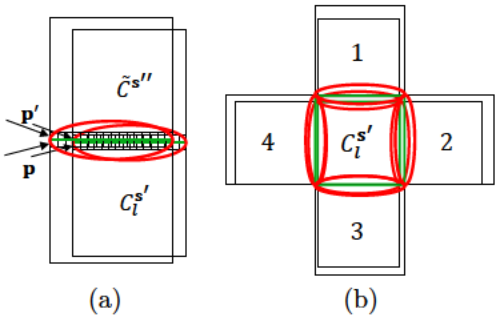


Figure 6: (a) Border between $C_l^{s'}$ and one neighboring block where discontinuities are analyzed. (b) Color discontinuity scheme of $\Phi(C_l^{s'})$, where $\mathbb{V}_4^{s'}$, i.e. 1, 2, 3 and 4, are used to analyze discontinuities with $C_l^{s'}$. Borders between blocks are marked in green and adjacent pixels of the border are circled in red.

$s' \in \mathbb{V}_4^s$. Figure 6(a) presents the pixel locations considered to compare two adjacent blocks and Figure 6(b) illustrates the $\mathbb{V}_4^{s'}$ neighborhood employed.

For each m-path, the candidate $\tilde{C}^{s',m}$ is selected by minimizing Φ:

$$\tilde{C}^{s',m} = \underset{\forall l \in \{1,..,N^{s'}\}}{argmin} \; \Phi(C_l^{s'}), \tag{12}$$

where $m \in \{1 \dots 8\}$ and $C_l^{s'}\ \forall l$ are the available candidates. Figure 5(e) shows the reconstruction of $\mathbb{V}_4^s$ starting from the initial estimation in Figure 5(c) where Figure 5(d) presents a temporary $\mathbb{V}_8^s$ single-path reconstruction.

Finally, we obtain the best estimation for the $\mathbb{V}_4^s$ neighborhood using the $m = 8$ paths. We select the best candidate $C_*^{s'}$ among the temporary blocks $\tilde{C}^{s',m}$:

$$C_*^{s'} = \underset{\forall m \in \{1,..,8\}}{argmin} \; \Phi(\tilde{C}^{s',m}), \tag{13}$$

where $\Phi(\tilde{C}^{s',m})$ is the Φ value obtained by the candidate during the m-path reconstruction (Eq. 11). As temporary blocks in $\mathbb{V}_4^s$ (top-center, bottom-center, middle-left and middle-right locations) employ three borders and temporary blocks in $\mathbb{V}_8^s$ (top-left, top-right, bottom-left and bottom-right locations) employ only two borders, we only select $C_*^{s'}$ for $\mathbb{V}_4^s$ due to its higher reliability. Then, the process of selecting $\tilde{B}^s$ and reconstructing its $\mathbb{V}_4^s$ is repeated until the complete background is generated. A

Algorithm 1 Sequential Multipath Reconstruction (SMR)

Input: $S^{\mathbf{s}}$ seeds and $C_l^{\mathbf{s}}$ candidates
Output: $B : B^{\mathbf{s}} \neq \emptyset, \forall \mathbf{s}$.

while $(\exists\, \tilde{B}^{\mathbf{s}} = \emptyset)$
  Selection of $\mathbf{s} : \tilde{B}^{\mathbf{s}} \neq \emptyset$
  for $m = 1$ to 8 **do**
    for $\mathbf{s}' \in \mathbb{V}_8^{\mathbf{s}}$
      if $\tilde{B}^{\mathbf{s}'} = \emptyset$ **then**
        Select $\tilde{C}^{\mathbf{s}',m}$ with Eq. 12
      end
    end
  end
  for $\mathbf{s}' \in \mathbb{V}_4^{\mathbf{s}}$
    Select $C_*^{\mathbf{s}'}$ with Eq. 13
    $\tilde{B}^{\mathbf{s}'} = C_*^{\mathbf{s}'}$
  end
end
$B = \tilde{B}$

summary of SMR is given in Algorithm 1.

5.3. Rejection based Multipath Reconstruction

SMR focuses on smoothness between adjacent blocks (external continuity, Φ similarity in Eq. 11) and, therefore, objects far from block boundaries may be unnoticed (e.g. stationary objects). These objects may have the minimum Φ value and be wrongly selected as the best candidate (Eq. 13). Moreover, another source of error exists as all external borders are not analyzed in $\mathbb{V}_8^{\mathbf{s}}$.

Extending SMR, we propose a *Rejection based Multipath Reconstruction* (RMR) scheme to overcome these limitations by rejecting reconstructions with high uncertainty, i.e. where some candidates $C_l^{\mathbf{s}'}$ have similar Φ value to the selected $C_*^{\mathbf{s}'}$ in Eq. 13. We disambiguate such selection by analyzing internal variations via intra-block heterogeneity Ψ and similarities to adjacent neighbors via inter-block color dissimilarity Ω. Figure 7 presents the diagram of operations performed by RMR.

RMR starts from an initial background estimation $\tilde{B}$ containing seeds $S^{\mathbf{s}}$ and empty locations (*Estimate initial background* stage in Figure 7). Then, RMR iteratively chooses a location $\mathbf{s}$ to reconstruct its empty neighbors via multiple paths $m \in \{1 \dots 8\}$ similarly to SMR (*Find location* $\mathbf{s}$ stage in Figure 7).

For each m-path, we then obtain the best candidate $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ using Φ as in Eq. 12. To infer high

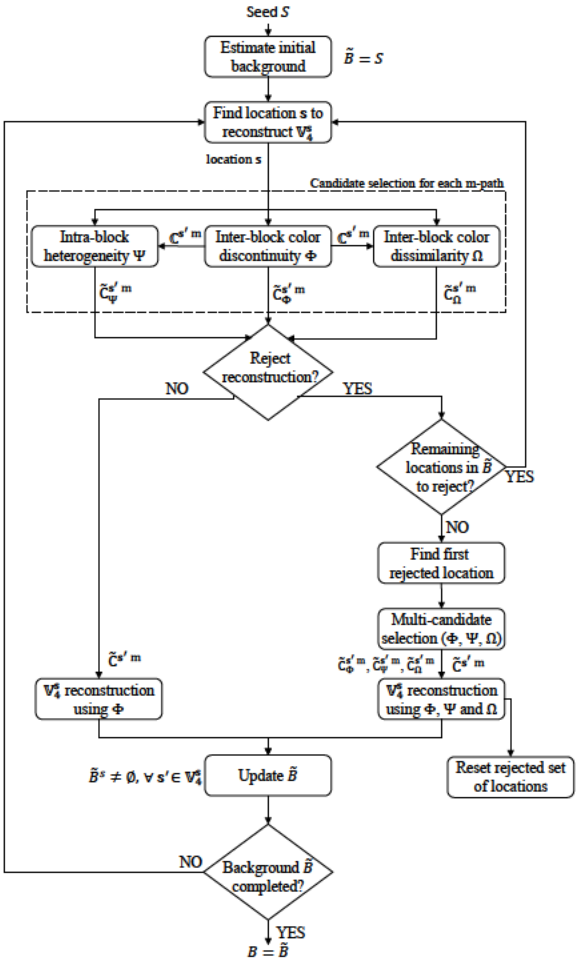


Figure 7: RMR diagram of operations. The diagram starts in the the top and ends in the bottom.

uncertain selections in the location s', a subset of candidates is obtained from the available ones $\left\{C_l^{\mathbf{s}'}\right\}_{l=1\ldots N^{\mathbf{s}'}}$:

$$\mathbb{C}^{\mathbf{s}',m} = \left\{C_l^{\mathbf{s}'}\ \forall l\ :\ \left|\Phi(C_l^{\mathbf{s}'}) - \Phi(\tilde{C}_\Phi^{\mathbf{s}',m})\right| < \rho\right\}, \tag{14}$$

where ρ is a similarity threshold with a small value to obtain highly similar candidates to the best selection $\tilde{C}_\Phi^{\mathbf{s}',m}$ that satisfy the smoothness constraints of the neighborhood.

To resolve such uncertainty in the selection using Φ, we employ intra-block heterogeneity Ψ and

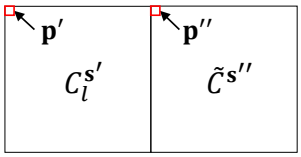


Figure 8: Scheme used to compute the inter-block color dissimilarity measure Ω. Pixel distances between $\mathbf{p}'$ and $\mathbf{p}''$ from blocks $C_l^{\mathbf{s}'}$ and $\tilde{C}^{\mathbf{s}''}$ are computed.

inter-block color dissimilarity Ω to the subset of candidates $C_l^{\mathbf{s}'} \in \mathbb{C}_l^{\mathbf{s}',m}$:

$$\Psi\left(C_l^{\mathbf{s}'}\right) = \sum_{q=1}^{64} \left| A_q(C_l^{\mathbf{s}'}) \right|^2, \tag{15}$$

$$\Omega\left(C_l^{\mathbf{s}'}\right) = \frac{1}{\left|\mathbb{V}_4^{\mathbf{s}'}\right|} \sum_{\mathbf{s}'' \in \mathbb{V}_4^{\mathbf{s}'}} \sum_{\substack{\mathbf{p}' \in \mathbf{s}' \\ \mathbf{p}'' \in \mathbf{s}''}} 1 - g\left(C_l^{\mathbf{s}'}\left(\mathbf{p}'\right), \tilde{C}^{\mathbf{s}''}\left(\mathbf{p}''\right)\right), \tag{16}$$

where A_q are the coefficients of the Discrete Cosine Transform (A_1 is set to 0 to remove zero-frequency data) [41] and $g(\cdot,\cdot)$ is the cosine similarity [42] between two pixels $\mathbf{p}'$ and $\mathbf{p}''$ from blocks $C_l^{\mathbf{s}'}$ and $\tilde{C}^{\mathbf{s}''}$. Figure 8 illustrates the scheme to compute Ω between blocks $C_l^{\mathbf{s}'}$ and $\tilde{C}^{\mathbf{s}''}$. $\Psi(C_l^{\mathbf{s}'})$ measures the variability of RGB values for the block considered whereas $\Omega(C_l^{\mathbf{s}'})$ measures the average pixel-level difference between RGB values of pixels in $C_l^{\mathbf{s}'}$ and $\tilde{C}^{\mathbf{s}''}$. Figure 9 presents a comparative example of the $\mathbb{V}_4^{\mathbf{s}}$ reconstruction. SMR selects a wrong candidate when an artifact appears in Figure 9(a) (e.g. block $C_*^{s'}$ with part of a blue bus occluding the background). As the measures Ψ and Ω have high values for this artifact, RMR correctly reconstructs the background as depicted in Figure 9(b). Note that the use of inter-block measures (Φ and Ω) minimizes discontinuities between blocks, thus reducing the block effect.

For each m-path, we apply $\Psi(C_l^{\mathbf{s}'})$ and $\Omega(C_l^{\mathbf{s}'})$ to the subset of candidates $C_l^{\mathbf{s}'} \in \mathbb{C}_l^{\mathbf{s}',m}$ in order to obtain two additional best candidates $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ as:

$$\tilde{C}_{\Psi}^{\mathbf{s}',m} = \underset{\forall C_l^{\mathbf{s}'} \in \mathbb{C}_l^{\mathbf{s}'},\, l \in \{1 \ldots N^{\mathbf{s}'}\}}{argmin} \Psi(C_l^{\mathbf{s}'}), \tag{17}$$

$$\tilde{C}_{\Omega}^{\mathbf{s}',m} = \underset{\forall C_l^{\mathbf{s}'} \in \mathbb{C}_l^{\mathbf{s}'},\, l \in \{1 \ldots N^{\mathbf{s}'}\}}{argmin} \Omega(C_l^{\mathbf{s}'}). \tag{18}$$

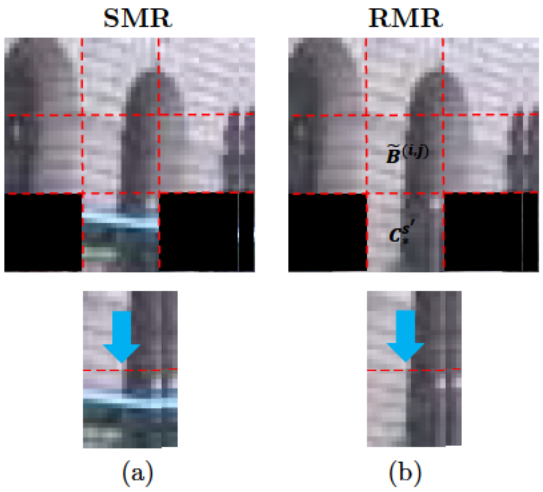


Figure 9: Example of the benefits that introduces the use of intra-block heterogeneity (Ψ) and inter-block dissimilarity (Ω) in RMR. (a) and (b) show the reconstruction of $\mathbb{V}_4^{(i,j)}$ of $\tilde{B}^{(i,j)}$ using SMR and RMR respectively, where the reconstructed block $C_*^{s'}$ was the only one unset from $\mathbb{V}_4^{(i,j)}$. Note that RMR is able to select the correct background through Ψ and Ω as they enforce the background smoothness, thus preventing the selection of artifacts done by SMR.

Thus, we infer highly-uncertain candidates when the three best selections $\tilde{C}_{\Phi}^{\mathbf{s}',m}$, $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ disagree (*Reject reconstruction?* stage in Figure 7). Therefore, we reject the assignment of a candidate to the background when:

$$Rejection = \begin{cases} 1 \; if & \neg(\tilde{C}_{\Phi}^{\mathbf{s}',m} = \tilde{C}_{\Psi}^{\mathbf{s}',m} = \tilde{C}_{\Omega}^{\mathbf{s}',m}) \\ 0 & otherwise \end{cases} \tag{19}$$

This rejection identifies when the candidate $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ is more homogeneous (low Ψ) or the candidate $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ is more similar to its neighborhood (low Ω) as compared to $\tilde{C}_{\Phi}^{\mathbf{s}',m}$. Hence, no assignment is done since a more suitable candidate may be employed ($\tilde{C}_{\Psi}^{\mathbf{s}',m}$ or $\tilde{C}_{\Omega}^{\mathbf{s}',m}$). The $\mathbb{V}_4^{\mathbf{s}}$ reconstruction of $\tilde{B}^{\mathbf{s}}$ is not performed when any m-path is rejected. Conversely, the $\mathbb{V}_4^{\mathbf{s}}$ reconstruction is performed as for SMR when none of the m-paths is rejected.

After rejecting all remaining locations (*Remaining locations in $\tilde{B}$ to reject*? stage in Figure 7), we analyze these rejected locations to complete the background reconstruction. Another iterative process begins to determine the next location $\tilde{B}^{\mathbf{s}}$ (*Find first rejected location* stage in Figure 7) and to select

the candidate $\tilde{C}^{\mathbf{s}',m}$ for each m-path (*Multi-candidate selection* stage in Figure 7) using a set of rules:

$$\tilde{C}^{\mathbf{s}',m} = \begin{cases} \tilde{C}_{\Phi}^{\mathbf{s}',m} \;\; if \;\; \tilde{C}_{\Phi}^{\mathbf{s}',m} = \tilde{C}_{\Psi}^{\mathbf{s}',m} = \tilde{C}_{\Omega}^{\mathbf{s}',m} \\ \tilde{C}_{\Psi}^{\mathbf{s}',m} \;\; if \;\; \tilde{C}_{\Phi}^{\mathbf{s}',m} \neq \tilde{C}_{\Psi}^{\mathbf{s}',m} \\ \tilde{C}_{\Omega}^{\mathbf{s}',m} \;\; if \;\; \tilde{C}_{\Phi}^{\mathbf{s}',m} = \tilde{C}_{\Psi}^{\mathbf{s}',m} \wedge \tilde{C}_{\Omega}^{\mathbf{s}',m} \neq \tilde{C}_{\Phi}^{\mathbf{s}',m} \end{cases}, \tag{20}$$

where $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ is selected when all blocks are the same, $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ is selected when it has better homogeneity than $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ as this may denote the presence of an artifact and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ is selected when the second condition does not occur and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ has better color similarity than $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ with its neighbors, i.e. there is a block with better Ω denoting that $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ may contain an artifact.

After selecting the m-candidates $\tilde{C}_{\Phi}^{\mathbf{s}',m}$, $\tilde{C}_{\Psi}^{\mathbf{s}',m}$ and $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ for all m-paths in Eq. (20), we combine them to obtain the best candidate $C_{*}^{\mathbf{s}'}$ for the location $\mathbf{s}'$:

$$C_{*}^{\mathbf{s}'} = \underset{m\in\{1,\ldots,8\}}{argmin} \; \Gamma\left(\tilde{C}_{\Phi}^{\mathbf{s}',m}, \tilde{C}_{\Psi}^{\mathbf{s}',m}, \tilde{C}_{\Omega}^{\mathbf{s}',m}\right), \tag{21}$$

where Γ combines the Φ, Ψ and Ω measures for the candidates for each m-path as:

$$\Gamma = \begin{cases} \overline{\Phi}(\tilde{C}^{\mathbf{s}',m}) & if \;\; \tilde{C}^{\mathbf{s}',m} = \tilde{C}_{\Phi}^{\mathbf{s}',m} \\ \overline{\Phi}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) + \overline{\Psi}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) + \overline{\Omega}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) & if \;\; (\tilde{C}^{\mathbf{s}',m} = \tilde{C}_{\Psi}^{\mathbf{s}',m}) \wedge \\ & \quad (\overline{\Omega}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) \leq \overline{\Omega}(\tilde{C}_{\Phi}^{\mathbf{s}',m})) \\ \overline{\Phi}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) + \overline{\Psi}(\tilde{C}_{\Psi}^{\mathbf{s}',m}) & if \;\; \tilde{C}^{\mathbf{s}',m} = \tilde{C}_{\Psi}^{\mathbf{s}',m} \\ \overline{\Phi}(\tilde{C}_{\Omega}^{\mathbf{s}',m}) + \overline{\Omega}(\tilde{C}_{\Omega}^{\mathbf{s}',m}) & if \;\; \tilde{C}^{\mathbf{s}',m} = \tilde{C}_{\Omega}^{\mathbf{s}',m} \end{cases}, \tag{22}$$

where the location $\mathbf{s}' \in \mathbb{V}_4^{\mathbf{s}}$; $\overline{\Phi}$, $\overline{\Psi}$ and $\overline{\Omega}$ are the normalized measures to the range [0,1] by their maximum value for all m-paths. Each case represents a different rejection, where the first one is applied when no rejection is detected in $\mathbf{s}'$, while the second, third and fourth cases apply to rejections due to Ψ and Ω, only Ψ and only Ω, respectively. This reconstruction of $\mathbb{V}_4^{\mathbf{s}}$ updates $\tilde{B}$ and it is iteratively performed until the entire background $\tilde{B}$ is reconstructed (*Background $\tilde{B}$ completed?* stage in Figure 7). The final estimated background B corresponds to the last iterative update of $\tilde{B}$. A summary of RMR is presented in algorithm 2.

Algorithm 2 Rejection based Multipath Reconstruction (RMR)

Input: $S^{\mathbf{s}}$ seeds and $C_l^{\mathbf{s}}$ candidates
Output: $B : B^{\mathbf{s}} \neq \emptyset, \forall \mathbf{s}$.

while $(\exists\, \tilde{B}^{\mathbf{s}} = \emptyset)$
  $\mathbb{K} = \emptyset$ (set of currently rejected locations)
  Selection of $\mathbf{s} : \tilde{B}^{\mathbf{s}} \neq \emptyset \wedge \mathbf{s} \notin \mathbb{K}$
  $Assigned = 0$
  $allR = 0$
  while $(Assigned = 0)$
    $Rejection = 0$
    for $m = 1$ to 8 **do**
      for $\mathbf{s}' \in \mathbb{V}_8^{\mathbf{s}}$
        if $\tilde{B}^{\mathbf{s}'} = \emptyset$ **then**
          Select $\tilde{C}_{\Phi}^{\mathbf{s}',m}$, $\tilde{C}_{\Psi}^{\mathbf{s}',m}$, $\tilde{C}_{\Omega}^{\mathbf{s}',m}$ with Eqs. 12, 17, 18
          if $\tilde{C}_{\Psi}^{\mathbf{s}',m} \neq \tilde{C}_{\Phi}^{\mathbf{s}',m} \vee \tilde{C}_{\Omega}^{\mathbf{s}',m} \neq \tilde{C}_{\Phi}^{\mathbf{s}',m} \wedge allR = 0$ **then**
            add $\mathbf{s}$ to $\mathbb{K}$
            $Rejection = 1$
            break
          else
            Select $\tilde{C}^{\mathbf{s}',m}$ with Eq. 20
          end
        end
      end
      if $Rejection = 1$ **then**
        break
      end
    end
    if $Rejection = 1$ **then**
      if all $\mathbf{s}$ are rejected **then**
        $\mathbb{K} = \emptyset$, $Rejection = 0$
        $allR = 1$
      else
        break
    else
      $Assigned = 1$
      for $\mathbf{s}' \in \mathbb{V}_4^{\mathbf{s}}$
        Select $C_*^{\mathbf{s}'}$ using Eq. 21
        $\tilde{B}^{\mathbf{s}'} = C_*^{\mathbf{s}'}$
      end
    end
  end
end
$B = \tilde{B}$

6. Experimental work

We evaluate the temporal and spatial analysis of the proposed approach, *Rejection based Multipath Reconstruction* (RMR), and provide comparisons against representative state-of-the-art approaches.

Figure 10: Visual examples of the selected sequences for evaluation. The IDs on the left correspond to the ones in Table 2.

6.1. Evaluation framework

6.1.1. Dataset

For evaluation we use 29 real sequences selected from public datasets (TRECVID[1], PBI[2] [12], AVSS 2007[3], LIRIS 2012[4] [43], CDNET[5] [44], Wallflower[6] [45], LIMU[7], CUHK[8] [46], COST211[9], IDIAP[10] [47], PETS 2009[11] [48], SAIVT-Campus [49]), covering different scenarios and complexities (see Figure 10), mainly stationary objects and crowds. Ground-truth data[12] has been manually composed from instants where the scene (or part of it) is foreground-free. Table 2 describes the properties of video sequences in terms of foreground *Stationarity*, according to size and duration; *Visibility*, according

[1] http://trecvid.nist.gov/trecvid.data.html
[2] http://www.diegm.uniud.it/fusiello/demo/bkg/
[3] http://www.eecs.qmul.ac.uk/~andrea/avss2007.html
[4] http://liris.cnrs.fr/voir/activities-dataset/videoframes.html
[5] http://changedetection.net/
[6] http://research.microsoft.com/en-us/um/people/jckrumm/WallFlower/TestImages.htm
[7] http://limu.ait.kyushu-u.ac.jp/dataset/en/
[8] http://www.ee.cuhk.edu.hk/~xgwang/CUHK_square.html
[9] http://www.csd.uoc.gr/~tziritas/cost.html
[10] http://www.idiap.ch/~odobez/RESSOURCES/DataRelease-TrafficJunction.php
[11] http://www.cvg.reading.ac.uk/PETS2009/
[12] Software and ground-truth data available at http://www-vpu.eps.uam.es/publications/BE_RMR

Table 2: Dataset description. Key. #f: Number of frames. T: Type. I: Indoor. O: Outdoor. S: Stationary region complexity. V: Visibility of empty scene complexity. SI: Shadows and Illumination changes complexity. L, M and H mean low, medium and high levels, respectively.

ID	Video	Dataset	#f	T	S	V	SI
1	*AB_H*	AVSS 2007	400	I	H	M	M
2	*PV_E*	AVSS 2007	500	I	H	L	M
3	*BSM*	LIMU	400	O	H	L	L
4	*SQ*	CUHK	500	O	H	L	L
5	*FGA*	Wallflower	400	I	H	L	L
6	*TREC1*	TRECVID	498	I	H	H	M
7	*TREC2*	TRECVID	699	I	L	H	M
8	MO	Wallflower	300	I	H	L	L
9	*PETS1*	PETS 2009	221	O	L	H	H
10	*PETS2*	PETS 2009	240	O	M	H	H
11	*PETS3*	PETS 2009	378	O	H	H	M
12	*Test*	SAIVT Campus	500	I	L	M	M
13	*Train*	SAIVT Campus	500	I	L	H	H
14	*TREC3*	TRECVID	400	I	M	M	M
15	*AB_Box*	CDNET	500	O	H	M	L
16	*bootstrap*	Wallflower	294	I	L	L	H
17	*ca_vignal*	PBI	258	O	M	L	L
18	*cam4*	TRECVID	300	I	M	L	L
19	*guardia*	PBI	400	O	H	M	L
20	*hall_m*	COST	300	I	M	M	L
21	*parking*	CDNET	400	O	H	L	L
22	*sofa*	CDNET	400	I	H	L	L
23	*st_light*	CDNET	400	O	H	H	L
24	*traffic*	IDIAP	500	O	H	L	L
25	*tramp*	CDNET	400	O	H	H	L
26	*vid16*	LIRIS 2012	380	I	H	L	L
27	*vid22*	LIRIS 2012	345	I	M	M	L
28	*vid36*	LIRIS 2012	128	I	M	M	L
29	*winter*	CDNET	500	O	H	L	M

to duration and size of the background visualized along time; *Shadows and Illumination changes*, according to the amount of these photometric factors. The ID of the video sequences displayed in Table 2 is used to report results. Additionally, comparisons are provided for the SBMI2015 dataset[13] [50] that contains 7 video sequences with their ground-truth images for the task of BE.

[13] http://sbmi2015.na.icar.cnr.it/SBIdataset.html

6.1.2. Evaluation measures

We compute performance via six different error measures adopted from SBMI2015 [50]. Three SBMI2015 measures employ the absolute gray-level difference Δ, which is defined for each pixel as:

$$\Delta(\mathbf{p}) = |B(\mathbf{p}) - GT(\mathbf{p})|_Y \ , \tag{23}$$

where B and GT denote the estimated and the ground-truth backgrounds, respectively. $|\cdot|_Y$ is the pixel-level absolute difference using the luminance information Y. The first measure, Average Gray-level Error (AGE), is the mean Δ value over the image. The second measure, Average of Error pixels (AE), determines pixel errors by thresholding Δ with $\alpha = 20$ and computes the percentage of error pixels in the image. The third measure, Average of Clustered Error pixels (ACE), considers the average number of error pixels where their 4-connected neighbors are error pixels. The lower the value, the better performance for AGE, AE and ACE. The remaining three measures are Peak-Signal-to-Noise-Ratio (PSNR), Multi-Scale Structural Similarity index (MS-SSIM) and Color image Quality Measure (CQM). The higher the value, the better performance for these three measures.

Additionally, we propose a threshold-free error measure to avoid the threshold dependency exhibited by AE. A number of thresholds α_i are employed to generate a curve with the corresponding AE values where the Area Under the Curve (AUC) is reported for performance evaluation.

6.1.3. Parametrization

For the proposed approach, we use $W = 16$ as the block size similarly to [8][10][11]. We heuristically set $k = 3$ for inter-frame differences in Eq. 1 to increase the motion detected as compared to consecutive frame differences. Finally, $\rho = 5$ is heuristically set to select candidates with color discontinuity similar to the minimum value in Eq. 14, as they may be part of the background. Note that we use less heuristic parameters than related state-of-the-art approaches [8][9][10][11].

6.2. Temporal analysis evaluation

We compare the proposed clustering to generate background candidates (Subsection 4.3) against the sequential clustering of algorithm DCT [8], which is chosen as a top-ranked state-of-the-art result

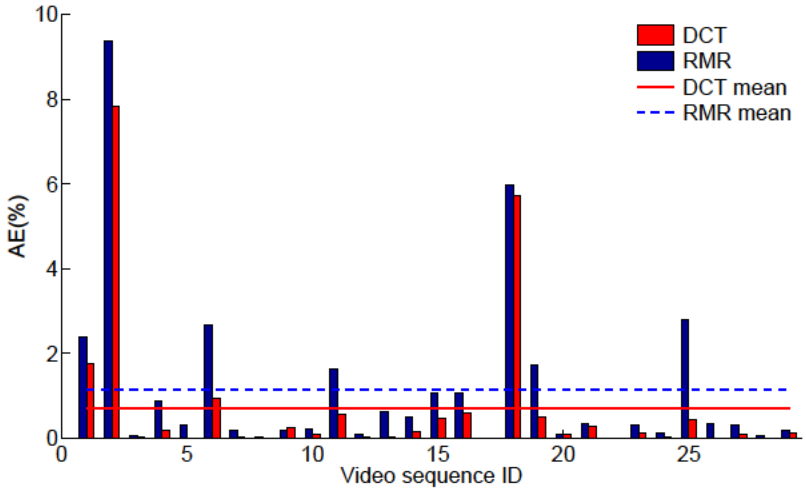


Figure 11: Clustering evaluation. The figure shows the AE error measure. The x-axis is the video sequence ID referenced in Table 2. The lower AE the better.

(as shown in Subsection 6.5). DCT clustering requires two thresholds to associate blocks into clusters; while the proposed clustering is automatic. We measure performance by inspecting whether any of the candidates $C_l^{\mathbf{s}}$ contains $GT^{\mathbf{s}}$ so the spatial analysis may be able to reconstruct the background. First, we determine the best matching between candidates and ground-truth $B_{best}^{\mathbf{s}}$ as follows:

$$B_{best}^{\mathbf{s}} = \underset{C_l^{\mathbf{s}}}{argmin} \left(\max_{\mathbf{p}} \left(\Delta \left(C_l^{\mathbf{s}}(\mathbf{p}), GT^{s}(\mathbf{p}) \right) \right) \right). \tag{24}$$

Second, we compute the AE measure ($\alpha = 20$) between $B_{best}^{\mathbf{s}}$ and the ground-truth data. Figure 11 compares mean AE performance for the proposed and selected approaches where both present similar scores, 1.157% for RMR and 0.699% for DCT. Attending to each sequence performance, both algorithms achieve low errors for all sequences except for 2 and 18, where some selected blocks of $B_{best}^{\mathbf{s}}$ differ from the ground-truth data due to variations in the illumination and reflections, respectively. Although RMR clustering slightly reduces performance compared to DCT [8], it has the advantage of being automatic (threshold-free), thus avoiding the adjustment needed in DCT clustering for different environments.

Table 3: Seed selection technique evaluation. Comparison between the selection described in DCT algorithm [8] and the proposed approach in RMR. As measures, we report the reconstruction percentage (RP) of the initial $\tilde{B}$ and AE. ID denotes the number of the video sequence referenced in Table 2. The higher RP the better. The lower AE the better. Green, black and red denotes better, equal and worse result than [8], respectively.

	RP		**AE**	
ID	**DCT [8]**	**RMR**	**DCT [8]**	**RMR**
1	4.11	12.80	0.14	1.92
2	3.62	4.11	3.70	3.40
3	0.33	13.33	0.00	0.00
4	0.48	34.30	0.00	0.2
5	1.25	1.25	0.39	0.017
6	5.31	24.88	0.00	0.00
7	0.72	2.17	0.13	0.00
8	13.75	1.25	0.00	0.00
9	14.12	56.71	0.00	0.00
10	0.23	26.62	0.00	0.16
11	3.70	18.29	0.00	0.00
12	13.89	18.18	0.05	0.00
13	1.52	10.10	0.00	0.00
14	6.28	15.22	0.00	0.89
15	0.41	12.35	0.00	0.00
16	1.25	3.75	0.00	0.00
17	22.22	11.11	0.00	3.13
18	51.25	5.00	5.68	0.00
19	39.16	46.15	0.00	0.00
20	2.120	31.82	0.00	0.10
21	62.67	59.00	0.25	0.00
22	15.33	45.00	0.00	0.00
23	12.33	14.00	0.00	0.00
24	0.97	17.87	0.00	0.00
25	0.21	0.21	0.00	0.00
26	0.48	1.69	0.00	0.00
27	0.24	40.58	0.00	0.00
28	0.24	0.24	0.00	0.00
29	12.00	23.67	0.00	0.00
Mean	10.01	19.02	0.004	0.004

6.3. Seed selection technique evaluation

We compare the performance of the RMR *Seed Selection* with the one proposed in DCT [8] where seed locations are selected when only a single candidate exists. As shown in Table 3, RMR initializes a higher percentage of the reconstructed background $\tilde{B}$ (19.02%) than DCT (10.01%), measured with

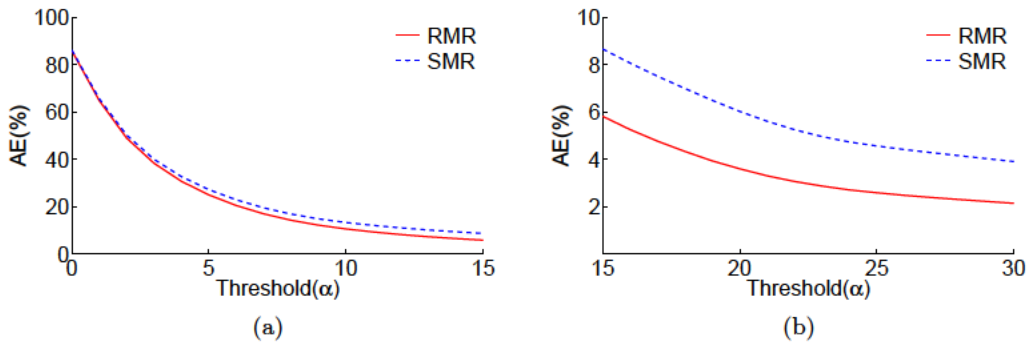


Figure 12: Comparison of SMR against RMR. Lower means better performance. (a) Comparison with $\alpha \in [0, 15]$. (b) Comparison with $\alpha \in [15, 30]$.

Reconstruction Percentage RP, i.e. amount of reconstructed blocks in the initialization, while keeping the correct selection of initial $\tilde{B}^{\mathbf{s}}$ blocks, i.e. $S^{\mathbf{s}}$, measured with AE ($\alpha = 20$). Low RP occurs when many block locations contain variations along the training frames, which is induced by low background visibility (7, 25, 28), background variations due to shadows (2, 16) or changing backgrounds (18) and large stationary objects (5, 8 and 26). Starting with a higher amount of initialized background blocks $\tilde{B}^{\mathbf{s}}$ provides more information for the iterative reconstructions which leads to improvements in the background estimation performance. Note that AE is computed over a partial reconstructed background whose average percentage RP is almost the double in RMR than in DCT, i.e. the initial estimation of the background contains more pixels and it may lead to more error pixels.

6.4. Spatial analysis evaluation

We compare RMR with SMR to show the benefits of iterative rejection. We avoid the threshold dependency of AE by computing multiple results using $\alpha \in [0, 30]$. The overall results for all sequences are shown in Figure 12 in terms of average AE. Figure 12(a) shows that SMR and RMR present similar results for low α values whereas Figure 12(b) indicates that RMR outperforms SMR due to its rejection capability. The Area Under the Curve (AUC) for SMR and RMR (lower area means better performance) is 392.86 and 359.13 for the evaluation interval $\alpha \in [0, 15]$ and 83.40 and 49.79 for $\alpha \in [15, 30]$.

The RMR improvement over SMR is illustrated by the examples in Figure 13, where reconstructions

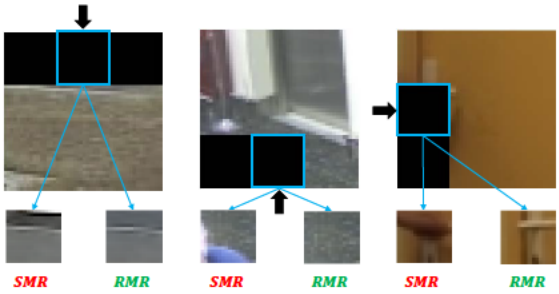


Figure 13: Examples of failures of SMR solved by RMR for the sequences *BSM* (left), *cam4* (middle) and *vid16* (right).

of $\mathbb{V}_4^{\mathbf{s}}$ for SMR and RMR are presented. For reconstructing the blue locations, SMR selects erroneous blocks, corresponding to artifacts (stationary objects), while RMR selects proper blocks. This occurs as SMR does not cope with the lack of not analyzing external edges of $\mathbb{V}_4^{\mathbf{s}}$ (black arrows), thus allowing discontinuities in that areas and due to failures of the fact that $\tilde{C}_{\Phi}^{\mathbf{s}',m}$ is the best candidate (as can be shown in the three examples of the figure). RMR solves these problems by analyzing Ψ and Ω of similar blocks belonging to $\mathbb{C}^{\mathbf{s}',m}$ (Eq. 14) and performing the rejection scheme.

6.5. Comparison against related approaches

We compare the proposed approach RMR against BE-specific approaches and top-ranked background subtraction algorithms. For BE, we select DCT [8], the Median (MED) [26], RSM [30] and IMBS-1 [51]. For Background Subtraction, we use SuBSENSE [20], 3dSOBS+ [26], Fuzzy [52], SC-SOBS [14], IMBS-2 [51], LOBSTER [53], SGMM-SOD [54] and two algorithms based on low-rank and sparse decomposition, LRGeomCG [55] and FPCP [56]. For these non-specific BE algorithms, we use the estimated background after processing all training frames. Note that their BE results may not reflect their performance for foreground detection. We use the BGSLibrary [2] (Fuzzy, LOBSTER and SuBSENSE) and the LRSLibrary [57] (LRGeomGC and FPCP). IMBS-1 uses IMBS initialization over the training frames, while IMBS-2 uses the default algorithm. We use default parameters for all approaches.

Figure 14 compares AE performance for the threshold $\alpha \in [0, 30]$ where results are split in two intervals to improve visibility. RMR has the best performance for both threshold intervals, followed by SGMM-SOD and DCT. The first sweep of the AE threshold ($\alpha \in [0, 15]$) presents high variation as low α values do not allow small variability with the ground-truth which should be handled as training

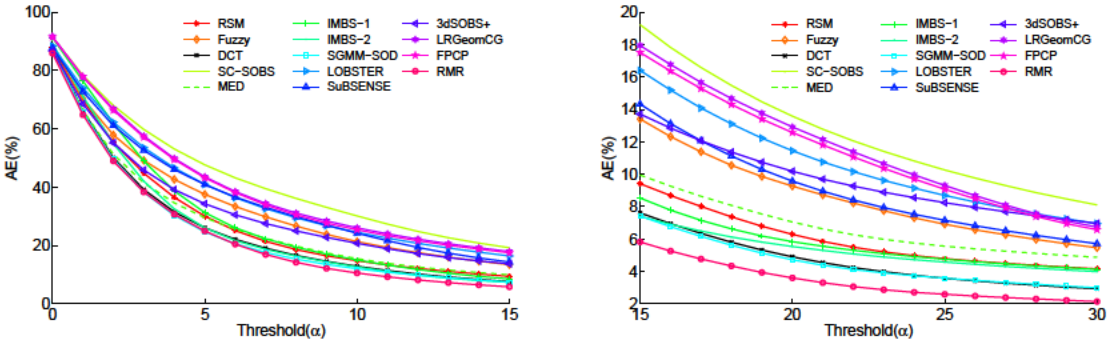


Figure 14: Evaluation of RMR against state-of-the-art methods for the task of BE and using the average of AE for all sequences. (a) Comparison with $\alpha \in [0, 15]$. (b) Comparison with $\alpha \in [15, 30]$.

Table 4: Comparison in terms of AUC and SBMI2015 error measures for the proposed dataset of 29 video sequences. The lower AUC, AGE, AE and ACE the better performance, while the higher MS-SSIM, PSNR and CQM the better the performance. Methods are presented in descending ranking order according to AUC for $\alpha \in [15, 30]$. Note that CQM measure is not computed for FPCP and LRGemoCG as background is obtained in gray-scale. The percentage of improvement compared to best state-of-the-art approach is shown under RMR performance.

Approach	AUC		AGE	AE	ACE	MS-SSIM	PSNR	CQM
	$\alpha \in [0, 15]$	$\alpha \in [15, 30]$						
RMR	**359.13**	**49.79**	**5.37**	**3.60**	**1.67**	**0.955**	**30.17**	**40.77**
	+3.6%	+25.0%	+10.3%	+23.7%	+19.7%	+1.6%	+6.3%	+2.4%
SGMM-SOD	372.56	66.38	5.99	4.72	2.08	0.940	28.37	39.83
DCT	384.79	67.55	6.12	4.90	2.35	0.939	27.66	38.94
IMBS-2	396.47	78.14	7.08	5.54	2.60	0.908	25.23	37.07
IMBS-1	451.83	83.15	7.83	5.84	2.74	0.907	24.44	36.00
RSM	428.39	88.00	7.65	6.29	2.96	0.899	24.40	36.73
MED	420.94	98.88	7.86	7.05	4.35	0.900	24.52	37.85
Fuzzy	510.20	126.24	8.29	9.26	5.79	0.911	25.67	39.06
SuBSENSE	549.23	131.69	9.36	9.57	4.36	0.899	24.31	35.83
3dSOBS+	486.64	142.45	9.18	10.18	6.10	0.881	24.32	36.72
LOBSTER	559.79	157.10	10.07	11.45	4.75	0.875	23.94	34.95
FPCP	588.08	166.71	9.79	12.57	8.56	0.898	24.52	-
LRGeomCG	594.87	170.99	9.95	12.92	8.57	0.896	24.41	-
SC-SOBS	639.78	185.33	11.68	13.58	6.46	0.839	22.24	35.23

frames may contain additive noise. Therefore, the sweep $\alpha \in [15, 30]$ is preferable to compute the performance. Table 4 includes further details in terms of AUC and SBMI2015 error measures. For all measures RMR outperforms state-of-the-art results, coping with stationary objects much better. Due to the variability of AE for $\alpha \in [0, 15]$, AUC from $\alpha \in [15, 30]$ better reflects the performance,

Table 5: Comparison in terms of AUC and SBMI2015 error measures for the SBMI dataset. The lower AUC, AGE, AE and ACE the better performance, while the higher MS-SSIM, PSNR and CQM the better the performance. Methods are presented in descending ranking order according to AUC for $\alpha \in [15, 30]$. Note that CQM measure is not computed for FPCP and LRGemoCG as background is obtained in gray-scale.The percentage of improvement compared to best state-of-the-art approach is shown under RMR performance.

Approach	AUC		AGE	AE	ACE	MS-SSIM	PSNR	CQM
	$\alpha \in [0, 15]$	$\alpha \in [15, 30]$						
RMR	**692.06**	**79.49**	**9.75**	**5.21**	**3.61**	**0.964**	**28.52**	39.54
	+6.1%	+50.0%	+23.9%	+50.2%	+49.1%	+6.5%	+8.6%	-1.7%
DCT	743.88	158.97	12.81	10.47	7.09	0.905	26.25	37.50
SGMM-SOD	755.26	209.84	16.19	13.34	9.83	0.884	25.73	35.52
RSM	737.00	236.63	17.00	15.96	10.55	0.816	23.30	35.13
IMBS-1	852.01	247.03	19.40	16.57	8.85	0.831	22.78	33.67
IMBS-2	834.12	279.84	20.72	19.25	10.32	0.795	22.37	33.60
LOBSTER	800.89	347.98	19.06	24.52	14.86	0.812	20.99	31.66
3dSOBS+	794.30	381.02	22.17	25.95	20.78	0.772	21.92	35.94
MED	771.76	393.81	21.31	27.19	22.39	0.806	23.41	37.27
Fuzzy	809.71	449.53	18.87	32.28	26.44	0.882	24.46	**40.23**
SuBSENSE	819.26	453.56	20.89	31.79	23.46	0.845	22.63	37.09
SC-SOBS	912.81	497.13	22.91	35.26	24.91	0.810	21.00	36.77
FPCP	1003.50	646.32	22.53	46.34	40.84	0.891	21.59	-
LRGeomCG	1012.30	656.29	22.90	47.37	40.26	0.885	21.41	-

being the best state-of-the-art approaches SGMM-SOD and DCT as both use smoothness constraints. Improvements can be analyzed regarding two sets of measures; the first includes AUC (significant AUC interval $\alpha \in [15, 30]$), AGE, AE and ACE; and the second one includes MS-SSIM, PSNR and CQM. For the first set of measures, we reduce the error in a range of 10.3 % (AGE) to 25.0% (AUC) compared to SGMM-SOD. For the second set of measures, the improvement compared to SGMM-SOD ranges from 1.6% (MS-SSIM) to 6.3% (PSNR). Additionally, experiments in the SBMI2015 dataset have been carried out (see Table 5) where again the proposed approach RMR outperforms the related work and where best compared approaches are again SGMM-SOD and DCT.

In Figure 15, sequence results are shown in terms of AUC against the DCT and SGMM-SOD approach (best related works), for $\alpha \in [15, 30]$. As shown in Figure 15, the proposed approach is better than DCT in 23 sequences and worse in 6, while compared to SGMM-SOD the proposed approach is better in 19 and worse in 10. The reasons of performance decrease can be compiled into failure of background smoothness assumption (sequences 4, 20 and 23), block effect (sequences 13 and

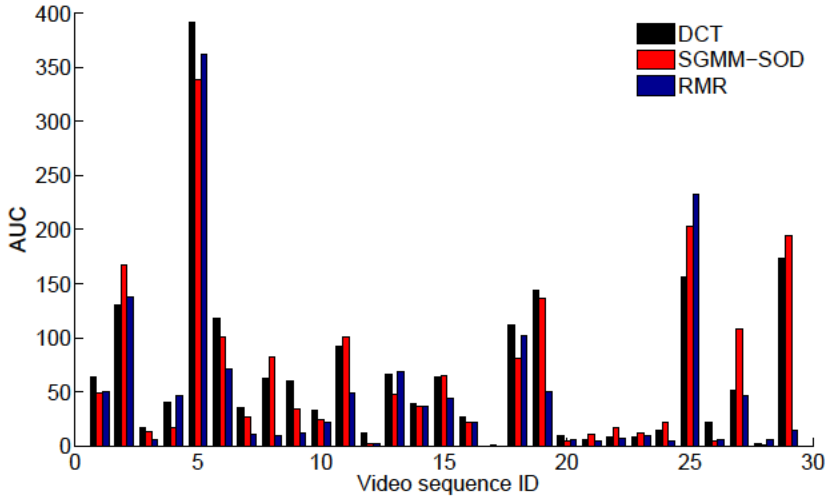


Figure 15: Sequence by sequence AUC ($\alpha \in [15, 30]$) of RMR (blue) against DCT (black) and SGMM-SOD (red) for the task of BE. The x-axis is the video sequence ID referenced in Table 2. The lower AUC the better performance.

406 26), differences between reconstructed background and ground-truth caused by illumination changes
407 or dynamic objects (sequences 2, 5, 12, 18, 28) and erroneous initialization in all algorithms where
408 high error propagation occurs (sequences 1 and 25). Therefore, regarding the stationarity challenge,
409 improvement is obtained in almost all sequences by RMR.

410 Figure 16 shows eight examples of the qualitative results in presence of stationarity, low visibility
411 and camouflages issues. In these examples, unlike most of the state-of-the-art approaches, the proposed
412 approach removes long-term stationary objects and crowds from the reconstructed background B (see
413 3, 4, 6, 13, 19, 24 and 29). However, video sequence 4 ($CUHK$) introduces erroneous white blocks due
414 to a higher continuity of a block $C_l^{\mathbf{s}'}$ with a white car that is later propagated. Also, video sequence
415 25 (*tramp*) induces errors (also in all the compared state-of-the-art) due to the combination of several
416 problems: inter-block color discontinuity measure Φ fails in one iteration, correct $C_l^{\mathbf{s}'}$ does not belong
417 to $\mathbb{C}^{\mathbf{s}',m}$ so the failure of Φ is not handled and the blocks with foreground motion are not correctly
418 removed due to moving regions bigger than the block size.

419 The comparative evaluation shows low performance of recent Background Subtraction algorithms
420 (IMBS-2, LOBSTER, SuBSENSE, SC-SOBS, 3dSOBS+ LRGeomCG and FPCP) when applied to
421 capture the background in situations with crowds or stationary objects. Some of these algorithms

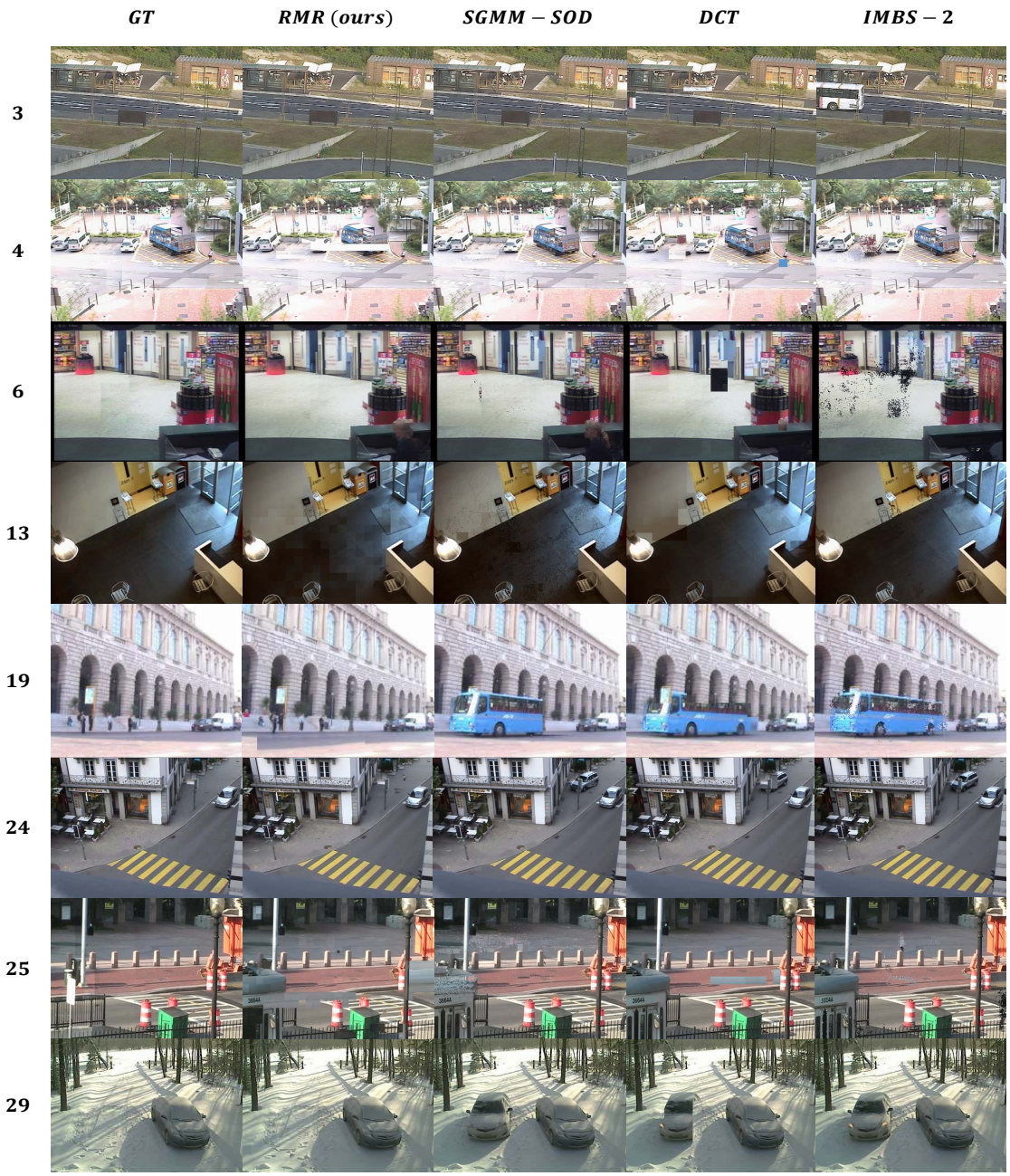


Figure 16: Qualitative results showing the estimated background B of top selected approaches for the BE task. From top to bottom rows: 3 (*BSM*), 4 (*CUHK*), 6 (*TREC1*), 19 (*guardia*), 24 (*traffic*) and 29 (*winter*) are examples with high complexity of stationarity solved successfully, while many approaches of the literature fail; 6 (*TREC1*) and 13 (*Train*) are examples where the background is successfully estimated under low visibility conditions; 25 (*tramp*) is an example of erroneous reconstruction due to non compliance of the rejection conditions. Each column corresponds to the results of a selected approach (first column is the manually extracted GT).

(IMBS-2, LOBSTER, SuBSENSE, 3dSOBS+ and SC-SOBS) are much faster than DCT and RMR at the cost of significant performance decreases because of the background assumptions, i.e. foreground is not representative in the training frames, which does not apply to stationary objects or crowds. Therefore, the spatial constrains introduced by RMR or DCT are needed to improve performance for background estimation in complex situations. One exception is SGMM-SOD that removes foreground ghosts based on spatial constraints, allowing a faster background update when stationary objects leave the scene. However, such update depends on the temporal duration of the stationary objects and training frames, obtaining errors when background has low visibility (see sequences 19, 24 and 29 in Figure 16) whereas RMR does not have such duration constraints.

The computational cost of the proposed approach is mainly due to the *Clustering* and *Multipath Reconstruction* stages, that consume approximately 28% and 70% of processing time. Our unoptimized MATLAB implementation of the proposed approach has an average running time of 5.3 μs/pixel (e.g. 200 color 350x240 frames with average resolution of 240x349 in around 4.5 minutes). Regarding the state-of-the-art, our approach performs similarly to other approaches. For example, RPCA methods use MATLAB implementations to run between 9.82 and 476 μs/pixel [21]. More complex background initialization approaches report a running time ranging from 65 to 312 μs/pixel [33][12], all using MATLAB. The current implementation of the proposed approach is currently restricted to offline operation, however significant speedups can be achieved by using other programming languages or by parallel processing.

7. Conclusions

We presented a block-wise BE approach to estimate the background of video sequences with moving and stationary objects. A clustering approach without the need of thresholds is performed over motion-filtered and dimension reduced data, which determines the candidates blocks to be background. Subsequently, a *Rejection based Multipath Reconstruction* based on background smoothness constraints selects the most suitable candidate. This multipath scheme includes a *Seed Selection* stage to initially estimate the background which is locally reconstructed using different paths (hypotheses), thus increasing the robustness against errors. An evaluation metric based on a sweep of threshold values is

proposed to avoid the threshold dependency of existing metric AE. The experiments validate the performance of the clustering analysis and the *Seed Selection* technique and provide comparisons against related work, demonstrating the advantages of the proposed approach. The results show that BE is highly complex since no algorithm is able to correctly perform in all situations.

As future work, we will explore the use of multi-resolution schemes, the improvement of background smoothness (e.g. by applying deblocking filters [58]) and the initialization-maintenance-detection interaction to improve Background Subtraction performance.

Acknowledgment

This work was partially supported by the Spanish Government (HAVideo, TEC2014-53176-R) and by the TEC department (Universidad Autónoma de Madrid).

References

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

[2] A. Sobral, A. Vacavant, A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos, Computer Vision and Image Understanding 122 (2014) 4–21.

[3] D. Park, H. Byun, A unified approach to background adaptation and initialization in public scenes, Pattern Recognition 46 (7) (2013) 1985–1997.

[4] M. Paul, Efficient video coding using optimal compression plane and background modelling, IET Image Processing 6 (9) (2012) 1311–1318.

[5] X. Chen, Y. Shen, Y. Yang, Background estimation using graph cuts and inpainting, in: Proceedings of Graphics Interface (GI), 2010, pp. 97–103.

[6] Y. Nakashima, N. Babaguchi, F. Jianping, Automatic generation of privacy-protected videos using background estimation, in: Proceedings of IEEE International Conference on Multimedia and Expo (ICME), 2011, pp. 1–6.

[7] M. Granados, H. Seidel, H. Lensch, Background estimation from non-time sequence images, in: Proceedings of Graphics Interface (GI), 2008, pp. 33–40.

[8] V. Reddy, C. Sanderson, B. Lovell, A low-complexity algorithm for static background estimation from cluttered image sequences in surveillance contexts, EURASIP Journal on Image and Video Processing (2011) 1–14.

[9] H. Hsiao, J. Leou, Background initialization and foreground segmentation for bootstrapping video sequences, EURASIP Journal on Image and Video Processing 12 (2013) 1–19.

[10] V. Reddy, C. Sanderson, B. Lovell, An efficient and robust sequential algorithm for background estimation in video surveillance, in: Proceedings of IEEE International Conference on Image Processing (ICIP), 2009, pp. 1109–1112.

[11] D. Baltieri, R. Vezzani, R. Cucchiara, Fast background initialization with recursive hadamard transform, in: Proceedings of IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2010, pp. 165–171.

[12] A. Colombari, A. Fusiello, Patch-based background initialization in heavily cluttered video, IEEE Transactions on Image Processing 19 (4) (2010) 926–933.

[13] M. Balcilar, A. Sonmez, Background estimation method with incremental iterative re-weighted least squares, Signal, Image and Video Processing (2015) 1–8.

[14] L. Maddalena, A. Petrosino, The SOBS algorithm: What are the limits?, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2012, pp. 21–26.

[15] R. Zhang, W. Gong, A. Yaworski, M. Greenspan, Nonparametric on-line background generation for surveillance video, in: Proceedings of International Conference on Pattern Recognition (ICPR), 2012, pp. 1177–1180.

[16] R. Colque, G. Camara-Chavez, Progressive background image generation of surveillance traffic videos based on a temporal histogram ruled by a reward/penalty function, in: Proceedings of Conference on Graphics, Patterns and Images (SIBGRAPI), 2011, pp. 297–304.

[17] T. Crivelli, P. Bouthemy, B. Cernuschi-Frías, J.-f. Yao, Simultaneous motion detection and background reconstruction with a conditional mixed-state markov random field, International Journal of Computer Vision 94 (3) (2011) 295–316.

[18] L. Maddalena, A. Petrosino, Background model initialization for static cameras, in: Background Modeling and Foreground Detection for Video Surveillance (Eds. T. Bouwmans, F. Porikli, B. Höferlin and A. Vacavant), Chapman and Hall/CRC 2014, 2014, Ch. 3, pp. 1–16.

[19] C. Zezhi, T. Ellis, A self-adaptive gaussian mixture model, Computer Vision and Image Understanding 122 (2014) 35–46.

[20] P. St-Charles, G. Bilodeau, R. Bergevin, Subsense: A universal change detection method with local adaptive sensitivity, IEEE Transactions on Image Processing 24 (1) (2015) 359–373.

[21] T. Bouwmans, E. H. Zahzah, Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance, Computer Vision and Image Understanding 122 (2014) 22–34.

[22] N. Oliver, B. Rosario, A. Pentland, A bayesian computer vision system for modeling human interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 831–843.

[23] Z. Hu, G. Ye, G. Jia, X. Chen, Q. Hu, K. Jiang, Y. Wang, L. Qing, Y. Tian, X. Wu, W. Gao, Pku@ trecvid2009: Single-actor and pair-activity event detection in surveillance video, in: Proceedings of TRECVID Workshop, 2009.

[24] Y. Tian, Y. Wang, Z. Hu, T. Huang, Selective eigenbackground for background modeling and subtraction in crowded scenes, IEEE Transactions on Circuits and Systems for Video Technology 23 (11) (2013) 1849–1864.

[25] H. Eng, K.-A. Toh, A. Kam, J. Wang, W.-Y. Yau, An automatic drowning detection surveillance system for challenging outdoor pool environments, in: Proceedings of IEEE International Conference on Computer Vision (ICCV), Vol. 1, 2003, pp. 532–539.

[26] L. Maddalena, A. Petrosino, The 3dSOBS+ algorithm for moving object detection, Computer Vision and Image Understanding 122 (2014) 65–73.

[27] C. Chia-Chih, J. Aggarwal, An adaptive background model initialization algorithm with objects moving at different depths, in: Proceedings of IEEE International Conference on Image Processing (ICIP), 2008, pp. 2664–2667.

[28] D. Gutchess, M. Trajkovics, E. Cohen-Solal, D. Lyons, A. K. Jain, A background model initialization algorithm for video surveillance, in: Proceedings of IEEE International Conference on Computer Vision (ICCV), Vol. 1, 2001, pp. 733–740.

[29] H.-H. Lin, T.-L. Liu, J.-H. Chuang, Learning a scene background model via classification, IEEE Transactions on Signal Processing 57 (5) (2009) 1641–1654.

[30] H. Wang, D. Suter, A novel robust statistical method for background initialization and visual surveillance, in: Proceedings of Asian Conference on Computer Vision (ACCV), Vol. 3851 of Lecture Notes in Computer Science, 2006, pp. 328–337.

[31] M. Benalia, S. Ait-Aoudia, An improved basic sequential clustering algorithm for background construction and motion detection, in: Image Analysis and Recognition, Vol. 7324 of Lecture Notes in Computer Science, 2012, pp. 216–223.

[32] A. Shrotre, L. Karam, Background recovery from multiple images, in: Proceedings of IEEE Digital Signal Processing and Signal Processing Education Meeting (DSP/SPE), 2013, pp. 135–140.

[33] X. Xun, T. Huang, A loopy belief propagation approach for robust background estimation, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–7.

[34] C. Guo, S. Gao, D. Zhang, Belief propagation algorithm for background estimation based on local maximum weight matching, in: Proceedings of International Congress on Image and Signal Processing (CISP), 2012, pp. 82–85.

[35] M. Chacon-Murguia, J. Ramirez-Quintana, D. Urias-Zavala, Segmentation of video background regions based on a dtcnn-clustering approach, Signal, Image and Video Processing (2014) 1–10.

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

[37] I. Jolliffe, Principal Component Analysis, John Wiley & Sons, Ltd, 2005.

[38] A. K. Jain, M. N. Murty, P. J. Flynn, Data clustering: A review, ACM Comput. Surv. 31 (3) (1999) 264–323.

[39] K. Wang, B. Wang, L. Peng, Cvap: Validation for cluster analyses, Data Science Journal 8 (2009) 88–93.

[40] J. Hartigan, Clustering Algorithms, John Wiley & Sons Inc., 1975.

[41] N. Ahmed, T. Natarajan, K. Rao, Discrete cosine transform, IEEE Transactions on Computers C-23 (1) (1974) 90–93.

[42] R. Dony, S. Wesolkowski, Edge detection on color images using rgb vector angles, in: Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering, Vol. 2, 1999, pp. 687–692.

[43] C. Wolf, E. Lombardi, J. Mille, O. Celiktutan, M. Jiu, E. Dogan, G. Eren, M. Baccouche, E. Dellandréa, C. Bichot, C. Garcia, B. Sankur, Evaluation of video activity localizations integrating quality and quantity measurements, Computer Vision and Image Understanding 127 (2014) 14–30.

[44] Y. Wang, P. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, P. Ishwar, Cdnet 2014: An expanded change detection benchmark dataset, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2014, pp. 393–400.

[45] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, Wallflower: principles and practice of background maintenance, in: In proceedings of IEEE International Conference on Computer Vision (ICCV), Vol. 1, 1999, pp. 255–261.

[46] M. Wang, W. Li, X. Wang, Transferring a generic pedestrian detector towards specific scenes, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 3274–3281.

[47] J. Varadarajan, J. Odobez, Topic models for scene analysis and abnormality detection, in: Proceedings of IEEE International Conference on Computer Vision Workshops (ICCVW), 2009, pp. 1338–1345.

[48] A. Ellis, A. Shahrokni, J. Ferryman, PETS2009 and winter-PETS 2009 results: A combined evaluation, in: IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), 2009, pp. 1–8.

[49] J. Xu, S. Denman, S. Sridharan, C. Fookes, Activity analysis in complicated scenes using dft coefficients of particle trajectories, in: IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), 2012, pp. 82–87.

[50] L. Maddalena, A. Petrosino, Towards benchmarking scene background initialization, in: New Trends in Image Analysis and Processing (ICIAP Workshops), Vol. 9281 of LNCS, 2015, pp. 469–476.

[51] D. Bloisi, A. Pennisi, L. Iocchi, Background modeling in the maritime domain, Machine Vision and Applications 25 (5) (2014) 1257–1269.

[52] F. El Baf, T. Bouwmans, B. Vachon, Fuzzy integral for moving object detection, in: IEEE International Conference on Fuzzy Systems (Fuzz-IEEE), 2008, pp. 1729–1736.

[53] P. St-Charles, G. Bilodeau, Improving background subtraction using local binary similarity patterns, in: IEEE Winter Conference on Applications of Computer Vision (WACV), 2014, pp. 509–515.

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

[55] B. Vandereycken, Low-rank matrix completion by Riemannian optimization, SIAM Journal on Optimization 23 (2) (2013) 1214–1236.

[56] P. Rodriguez, B. Wohlberg, Fast principal component pursuit via alternating minimization, in: Proceedings of IEEE International Conference on Image Processing (ICIP), 2013, pp. 69–73.

[57] A. Sobral, T. Bouwmans, E.-h. Zahzah, Lrslibrary: Low-rank and sparse tools for background modeling and subtraction in videos, in: Robust Low-Rank and Sparse Matrix Decomposition: Applications in Image and Video Processing, CRC Press, Taylor and Francis Group., 2015.

[58] P. List, A. Joch, J. Lainema, G. Bjontegaard, M. Karczewicz, Adaptive deblocking filter, IEEE Transactions on Circuits and Systems for Video Technology 13 (7) (2003) 614–619.