A Low-Complexity Algorithm for Static Background Estimation from Cluttered Image Sequences in Surveillance Contexts

Vikas Reddy, Conrad Sanderson, Brian C. Lovell

NICTA, Australia
University of Queensland, Brisbane, Australia

***Abstract*—For the purposes of foreground estimation, the true background model is unavailable in many practical circumstances and needs to be estimated from cluttered image sequences. We propose a sequential technique for static background estimation in such conditions, with low computational and memory requirements. Image sequences are analysed on a block-by-block basis. For each block location a representative set is maintained which contains distinct blocks obtained along its temporal line. The background estimation is carried out in a Markov Random Field framework, where the optimal labelling solution is computed using iterated conditional modes. The clique potentials are computed based on the combined frequency response of the candidate block and its neighbourhood. It is assumed that the most appropriate block results in the smoothest response, indirectly enforcing the spatial continuity of structures within a scene. Experiments on real-life surveillance videos demonstrate that the proposed method obtains considerably better background estimates (both qualitatively and quantitatively) than median filtering and the recently proposed "intervals of stable intensity" method. Further experiments on the Wallflower dataset suggest that the combination of the proposed method with a foreground segmentation algorithm results in improved foreground segmentation.**

I. Introduction

Intelligent surveillance systems can be used effectively for monitoring critical infrastructure such as banks, airports and railway stations [1]. Some of the key tasks of these systems are real-time segmentation, tracking and analysis of foreground objects of interest [2], [3]. Many approaches for detecting and tracking objects are based on background subtraction techniques, where each frame is compared against a background model for foreground object detection.

The majority of background subtraction methods adaptively model and update the background for every new input frame. Surveys on this class of algorithms are found in [4], [5]. However, most methods presume the training image sequence used to model the background is free from foreground objects [6], [7], [8]. This assumption is often not true in the case of uncontrolled environments such as train stations and airports, where directly obtaining a clear background is almost impossible. Furthermore, in certain situations a strong illumination change can render the existing background model ineffective, thereby forcing us to compute a new background model. In such circumstances, it becomes inevitable to estimate the background using cluttered sequences (i.e. where parts of the background are occluded). A good background estimate will complement the succeeding background subtraction process, which can result in improved detection of foreground objects.

(i) **(ii)**

Fig. 1. Typical example of estimating the background from an cluttered image sequence: **(i)** input frames cluttered with foreground objects, where only parts of the background are visible; **(ii)** estimated background.

The problem can be paraphrased as follows: given a short image sequence captured from a stationary camera in which the background is occluded by foreground objects in every frame of the sequence for most of the time, the aim is to estimate its background, as illustrated in Figure 1. This problem is also known in the literature as background initialisation or bootstrapping [9]. Background estimation is related to, but distinct from, background modelling. Owing to the complex nature of the problem, we confine our estimation strategy to static backgrounds (e.g. no waving trees), which is quite common in urban surveillance environments such as banks, shopping malls, airports and train stations.

Existing background estimation techniques, such as simple median filtering, typically require the storage of all the input frames in memory before estimating the background. This increases memory requirements immensely. In this paper we propose a robust background estimation algorithm in a Markov Random Field (MRF) framework. It operates on the input frames sequentially, avoiding the need to store all the frames. It is also computationally less intensive, enabling the system to achieve real-time performance — this aspect is critical in video surveillance applications. This paper is a thoroughly revised and extended version of our previous work [10].

We continue as follows. Section II gives an overview of existing methods for background estimation. Section III describes the proposed algorithm in detail. Results from experiments on real-life surveillance videos are given in Section IV, followed by the main findings in Section V.

Published in: EURASIP Journal on Image and Video Processing, 2011.
http://dx.doi.org/10.1155/2011/164956
C++ source code available at **http://arma.sourceforge.net/background_est/**

II. Previous Work

Existing methods to address the cluttered background estimation problem can be broadly classified into three categories: **(i)** pixel-level processing, **(ii)** region-level processing, **(iii)** a hybrid of the first two. It must be noted that all methods assume the background to be static. The three categories are overviewed in the sections below.

A. Pixel-level Processing

In the first category the simplest techniques are based on applying a median filter on pixels at each location across all the frames. Lo and Velastin [11] apply this method to obtain reference background for detecting congestion on underground train platforms. However, its limitation is that the background is estimated correctly only if it is exposed for more than 50% of the time. Long and Yang [12] propose an algorithm that finds pixel intervals of stable intensity in the image sequence, then heuristically chooses the value of the longest stable interval to most likely represent the background. Bevilacqua [13] applies Bayes' theorem in his proposed approach. For every pixel it estimates the intensity value to which that pixel has the maximum posterior probability.

Wang and Suter [14] employ a two-staged approach. The first stage is similar to that of [12], followed by choosing background pixel values whose interval maximises an objective function. It is defined as N_{l_k}/S_{l_k} where N_{l_k} and S_{l_k} are the length and standard variance of the k-th interval of pixel sequence l. The method proposed by Kim et al. [15] quantises the temporal values of each pixel into distinct bins called codewords. For each codeword, it keeps a record of the maximum time interval during which it has not recurred. If this time period is greater than $N/2$, where N is the total number of frames in the sequence, the corresponding codeword is discarded as foreground pixel. The system recently proposed by Chiu et al. [16] estimates the background and utilises it for object segmentation. Pixels obtained from each location along its time axis are clustered based on a threshold. The pixel corresponding to the cluster having the maximum probability and greater than a time-varying threshold is extracted as background pixel.

All these pixel based techniques can perform well when the foreground objects are moving, but are likely to fail when the time interval of exposure of the background is less than that of the foreground.

B. Region-level Processing

In the second category, the method proposed by Farin et al. [17] performs a rough segmentation of input frames into foreground and background regions. To achieve this, each frame is divided into blocks, the temporal sum of absolute differences (SAD) of the co-located blocks is calculated, and a block similarity matrix is formed. The matrix elements that correspond to small SAD values are considered as stationary elements and high SAD values correspond to non-stationary elements. A median filter is applied only on the blocks classified as background. The algorithm works well in most scenarios, however, the spatial correlation of a given block with its neighbouring blocks already filled by background is not exploited, which can result in estimation errors if the objects are quasi-stationary for extended periods.

In the method proposed by Colombari et al. [18], each frame is divided into blocks of size $N \times N$ overlapping by 50% in both dimensions. These blocks are clustered using single linkage agglomerative clustering along their time-line. In the following step the background is built iteratively by selecting the best continuation block for the current background using the principles of visual grouping. The spatial correlations that naturally exist within small regions of the background image are considered during the estimation process. The algorithm can have problems with blending of the foreground and background due to slow moving or quasi-stationary objects. Furthermore, the algorithm is unlikely to achieve real-time performance due to its complexity.

C. Hybrid Approaches

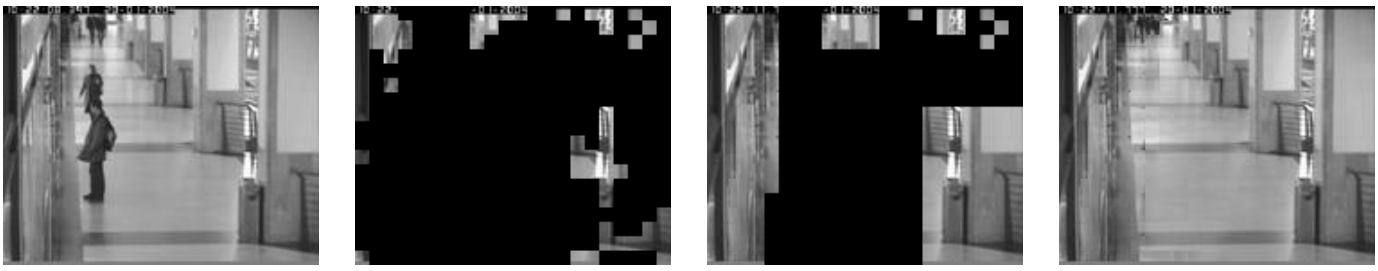
In the third category, the algorithm presented by Gutchess et al. [19] has two stages. The first stage is similar to that of [12], with the second stage estimating the likelihood of background visibility by computing the optical flow of blocks between successive frames. The motion information helps classify an intensity transition as background to foreground or vice versa. The results are typically good, but the usage of optical flow for each pixel makes it computationally intensive.

In [20], Cohen views the problem of estimating the background as an optimal labelling problem. The method defines an energy function which is minimised to achieve an optimal solution at each pixel location. It consists of *data* and *smoothness* terms. The data term accounts for pixel stationarity and motion boundary consistency while the smoothness term looks for spatial consistency in the neighbourhood. The function is minimised using the α–*expansion* algorithm [21] with suitable modifications. A similar approach with a different energy function is proposed by Xu and Huang [22]. The function is minimised using loopy belief propagation algorithm. Both solutions provide robust estimates, however, their main drawback is large computational complexity to process a small number of input frames. For instance, in [22] the authors report a prototype of the algorithm on Matlab takes about 2.5 minutes to estimate the background from a set of only 10 images of QVGA resolution (320×240).

III. Proposed Algorithm

We propose a computationally efficient, region-level algorithm that aims to address the problems described in the previous section. It has several additional advantages as well as novelties, including:

- The background estimation problem is recast into an MRF scheme, providing a theoretical framework.
- Unlike the techniques mentioned in Section II, it does not expect all frames of the sequence to be stored in memory simultaneously — instead, it processes frames sequentially, which results in a low memory footprint.



(i) **(ii)** **(iii)** **(iv)**

Fig. 2. **(i)** Example frame from an image sequence, **(ii)** partial background initialisation (after *Stage 2*), **(iii)** remaining background estimation in progress (*Stage 3*), **(iv)** estimated background.

- The formulation of the clique potential in the MRF scheme is based on the combined frequency response of the candidate block and its neighbourhood. It is assumed that the most appropriate configuration results in the smoothest response (minimum energy), indirectly exploiting the spatial correlations within small regions of a scene.
- Robustness against high frequency image noise. In the calculation of the energy potential we compute 2D Discrete Cosine Transform (DCT) of the clique. The high frequency DCT coefficients are ignored in the analysis as they typically represent image noise.

A. Overview of the Algorithm

In the text below we first provide an overview of the proposed algorithm, followed by a detailed description of its components (Sections III-B to III-E). It is assumed that at each block location: **(i)** the background is static and is revealed at some point in the training sequence for a short interval, and **(ii)** the camera is stationary. The background is estimated by recasting it as a labelling problem in an MRF framework. The algorithm has three stages.

Let the resolution of the greyscale image sequence I be $\mathcal{W} \times \mathcal{H}$. In the first stage, the frames are viewed as instances of an undirected graph, where the nodes of the graph are blocks of size $N \times N$ pixels[1]. We denote the nodes of the graph by $\mathcal{N}(i,j)$ for $i = 0, 1, 2, \cdots, (\mathcal{W}/N) - 1$, $j = 0, 1, 2, \cdots, (\mathcal{H}/N) - 1$. Let I_f be the f-th frame of the training image sequence and let its corresponding node labels be denoted by $\mathcal{L}_f(i,j)$, and $f = 1, 2, \cdots, F$, where F is the total number of frames. For convenience, each node label $\mathcal{L}_f(i,j)$ is vectorised into an N^2 dimensional vector $\mathbf{l}_f(i,j)$.

At each node location (i,j), a representative set $\mathcal{R}(i,j)$ is maintained. It contains distinct labels that were obtained along its temporal line. Two labels are considered as distinct (visually different), if they fail to adhere to one of the constraints described in Section III-B. Let these unique representative labels be denoted by $\mathbf{r}_k(i,j)$ for $k = 1, 2, \cdots, S$ (with $S \leq F$), where $\mathbf{r}_k$ denotes the mean of all the labels which were considered as similar to each other (mean of the cluster). Each label $\mathbf{r}_k$ has an associated weight W_k which denotes its number of occurrences in the sequence, i.e., the number of labels at location (i,j) which are deemed to be the same as $\mathbf{r}_k(i,j)$. For every such match, the corresponding $\mathbf{r}_k(i,j)$ and its associated variance, $\mathbf{\Sigma}_k(i,j)$ are updated recursively as given below:

$$\mathbf{r}_k^{new} = \mathbf{r}_k^{old} + \frac{1}{W_k + 1}\left(\mathbf{l}_f - \mathbf{r}_k^{old}\right) \tag{1}$$

$$\mathbf{\Sigma}_k^{new} = \frac{W_k - 1}{W_k}\mathbf{\Sigma}_k^{old} + \frac{1}{W_k + 1}(\mathbf{l}_f - \mathbf{r}_k^{old})'(\mathbf{l}_f - \mathbf{r}_k^{old}) \tag{2}$$

where $\mathbf{r}_k^{old}$, $\mathbf{\Sigma}_k^{old}$ and $\mathbf{r}_k^{new}$, $\mathbf{\Sigma}_k^{new}$ are the values of $\mathbf{r}_k$ and its associated variance before and after the update respectively, and $\mathbf{l}_f$ is the incoming label which matched $\mathbf{r}_k^{old}$. It is assumed that one element of $\mathcal{R}(i,j)$ corresponds to the background.

In the second stage, representative sets $\mathcal{R}(i,j)$ having just one label are used to initialise the corresponding node locations $\mathcal{B}(i,j)$ in the background $\mathcal{B}$.

In the third stage, the remainder of the background is estimated iteratively. An optimal labelling solution is calculated by considering the likelihood of each of its labels along with the *a priori* knowledge of the local spatial neighbourhood modelled as a MRF. Iterated conditional mode (ICM), a deterministic relaxation technique, performs the optimisation. The framework is described in detail in Section III-C. The strategy for selecting the location of an empty background node to initialise a label is described in Section III-D. The procedure for calculating the energy potentials, a prerequisite in determining the *a priori* probability, is described in Section III-E.

The overall pseudo-code of the algorithm is given in Algorithm 1 and an example of the algorithm in action is shown in Figure 2.

B. Similarity Criteria for Labels

We assert that two labels $\mathbf{l}_f(i,j)$ and $\mathbf{r}_k(i,j)$ are similar if the following two constraints are satisfied:

$$\frac{(\mathbf{r}_k(i,j) - \mu_{r_k}(i,j))'\left(\mathbf{l}_f(i,j) - \mu_{l_f}(i,j)\right)}{\sigma_{r_k}\sigma_{l_f}} > \mathcal{T}_1 \tag{3}$$

and

$$\frac{1}{N^2}\sum\nolimits_{n=0}^{N^2-1} |d_{k_n}(i,j)| < \mathcal{T}_2 \tag{4}$$

[1] For implementation purposes, each block location and its instances at every frame are treated as a node and its labels, respectively.

Stage 1: Collection of Label Representatives

1) $\mathcal{R} \leftarrow \emptyset$ (null set)
2) **for** $f = 1$ to F **do**
   a) Split input frame I_f into node labels, each with a size of $N \times N$.
   b) **for each** node label $\mathcal{L}_f(i,j)$ **do**
      i) Vectorise node $\mathcal{L}_f(i,j)$ into $\mathbf{l}_f(i,j)$.
      ii) Find the representative label $\mathbf{r}_m(i,j)$ from the set $\mathcal{R}(i,j) = (\mathbf{r}_k(i,j) | 1 \leq k \leq S)$, matching to $\mathbf{l}_f(i,j)$ based on conditions in Eqns. (3) and (4).
         if ($\mathcal{R}(i,j)$ = $\{\emptyset\}$ or there is no match) **then**
            $k \leftarrow k + 1$.
            Add a new representative label $\mathbf{r}_k(i,j) \leftarrow \mathbf{l}_f(i,j)$ to set $\mathcal{R}(i,j)$ and initialise its weight, $W_k(i,j)$, to 1.
         else
            Recursively update the matched label $\mathbf{r}_m(i,j)$ and its variance given by Eqns. (1) and (2) respectively.
            $W_m(i,j) \leftarrow W_m(i,j) + 1$
         end if
      end for each
   end for

Stage 2: Partial Background Initialisation

1) $\mathcal{B} \leftarrow \emptyset$
2) **for each** set $\mathcal{R}(i,j)$ **do**
   if (size($\mathcal{R}(i,j)$) $= 1$) **then**
      $\mathcal{B}(i,j) \leftarrow \mathbf{r}_1(i,j)$.
   end if
   end for each

Stage 3: Estimation of the Remaining Background

1) *Full background initialisation*
   while ($\mathcal{B}$ not filled) **do**
      if $\mathcal{B}(i,j) = \emptyset$ and has neighbours as specified in Section III-D **then**
         $\mathcal{B}(i,j) \leftarrow \mathbf{r}_{max}(i,j)$, the label out of set $\mathcal{R}(i,j)$ which yields maximum value of the posterior probability described in Eqn. (12) (see Section III-C).
      end if
   end while
2) *Application of ICM*
   $iteration\_count \leftarrow 0$
   while ($iteration\_count < total\_iterations$) **do**
      for each set $\mathcal{R}(i,j)$ **do**
         if $P(\mathbf{r}_{new}(i,j)) > P(\mathbf{r}_{old}(i,j))$ **then**
            $\mathcal{B}(i,j) \leftarrow \mathbf{r}_{new}(i,j)$, where $P(\cdot)$ is the posterior probability defined by Eqn. (12).
         end if
      end for each
      $iteration\_count = iteration\_count + 1$
   end while

Algorithm 1. Pseudo-code for the proposed algorithm. A C++ implementation is available at **http://arma.sourceforge.net/background_est/**

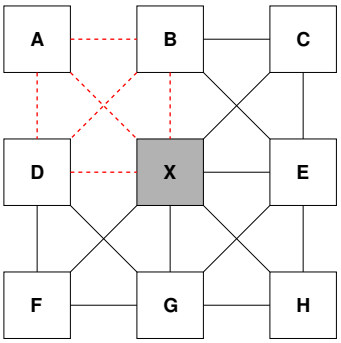


Fig. 3. The local neighbourhood system and its four cliques. Each clique is comprised of 4 nodes (blocks). To demonstrate one of the cliques, the the top-left clique has dashed links.

Equations (3) and (4), respectively, evaluate the correlation coefficient and the mean of absolute differences (MAD) between the two labels, with the latter constraint ensuring that the labels are close in N^2 dimensional space. μ_{r_k}, μ_{l_f} and $\sigma_{r_k}, \sigma_{l_f}$ are the mean and standard deviation of the elements of labels $\mathbf{r}_k$ and $\mathbf{l}_f$ respectively, while $\mathbf{d}_k(i,j) = \mathbf{l}_f(i,j) - \mathbf{r}_k(i,j)$.

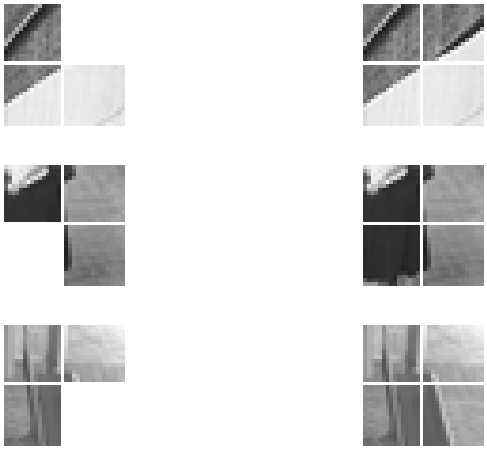
$\mathcal{T}_1$ is selected empirically (see Section IV), to ensure that two visually identical labels are not treated as being different due to image noise. $\mathcal{T}_2$ is proportional to image noise and is found automatically as follows. Using a short training video, the MAD between co-located labels of successive frames is calculated. Let the number of frames be L and N_b be the number of labels per frame. The total MAD points obtained will be $(L-1)N_b$. These points are sorted in ascending order and divided into quartiles. The points lying between quartiles Q_3 and Q_1 are considered. Their mean, $\mu_{\mathrm{Q}_{31}}$ and standard deviation, $\sigma_{\mathrm{Q}_{31}}$, are used to estimate $\mathcal{T}_2$ as $2\times(\mu_{\mathrm{Q}_{31}}+2\sigma_{\mathrm{Q}_{31}})$. This ensures that low MAD values (close or equal to zero) and high MAD values (arising due to movement of objects) are ignored (i.e. treated as outliers).

We note that both constraints (3) and (4) are necessary. As an example, two vectors $[1, 2, \cdots, 16]$ and $[101, 102, \cdots, 116]$ have a perfect correlation of 1 but their MAD will be higher than $\mathcal{T}_2$. On the other hand, if a thin edge of the foreground object is contained in one of the labels, their MAD may be well within $\mathcal{T}_2$. However, Eqn. (3) will be low enough to indicate the dissimilarity of the labels. In contrast, we note that in [18] the similarity criteria is just based on the sum of squared distances between the two blocks.

C. Markov Random Field (MRF) Framework

Markov random field/probabilistic undirected graphical model theory provides a coherent way of modelling context-dependent entities such as pixels or edges of an image. It has a set of nodes, each of which corresponds to a variable or a group of variables, and set of links each of which connects a pair of nodes. In the field of image processing it has been widely employed to address many problems that can be modelled as labelling problem with contextual information [23], [24].

Let $\mathbf{X}$ be a 2D random field, where each random variate $X_{(i,j)}$ $(\forall\, i,j)$ takes values in discrete *state space* Λ. Let $\omega \in$



(i) **(ii)**

Fig. 4. **(i)** Three cliques each of which has an empty node. The gaps between the blocks are for ease of interpretation only. **(ii)** Same cliques where the empty node has been labelled. The constraint of 3 neighbouring nodes to be available in 3 different directions as illustrated ensures that arbitrary edge continuities are taken into account while assigning the label at the empty node.

Ω be a *configuration* of the variates in $\mathbf{X}$, and let Ω be the set of all such configurations. The joint probability distribution of $\mathbf{X}$ is considered Markov if

$$p(\mathbf{X} = \omega) > 0,\ \forall\ \omega \in \Omega \tag{5}$$

and

$$p\left(X_{(i,j)} | X_{(p,q)}, (i,j) \neq (p,q)\right) = p\left(X_{(i,j)} | X_{N_{(i,j)}}\right) \tag{6}$$

where $X_{N_{(i,j)}}$ refers to the local *neighbourhood system* of $X_{(i,j)}$.

Unfortunately, the theoretical factorisation of the joint probability distribution of the MRF turns out to be intractable. To simplify and provide computationally efficient factorisation, Hammersley-Clifford theorem [25] states that an MRF can equivalently be characterised by a Gibbs distribution. Thus

$$p(\mathbf{X} = \omega) = \frac{e^{-U(\omega)/T}}{Z} \tag{7}$$

where

$$Z = \sum\nolimits_{\omega} e^{-U(\omega)/T} \tag{8}$$

is a normalisation constant known as the *partition function*, T is a constant used to moderate the peaks of the distribution and $U(\omega)$ is an *energy function* which is the sum of *clique/energy potentials* V_c over all possible cliques C:

$$U(\omega) = \sum\nolimits_{c \in \boldsymbol{C}} V_c(\omega) \tag{9}$$

The value of $V_c(\omega)$ depends on the local configuration of clique c.

In our framework, information from two disparate sources is combined using Bayes' rule. The local visual observations at each node to be labelled yield label likelihoods. The resulting label likelihoods are combined with *a priori* spatial knowledge of the neighbourhood represented as an MRF.

Let each input image I_f be treated as a realisation of the random field $\mathcal{B}$. For each node $\mathcal{B}(i,j)$, the representative set $\mathcal{R}(i,j)$ (see Section III-A) containing unique labels is treated as its *state space* with each $\mathbf{r}_k(i,j)$ as its plausible label[2].

Using Bayes' rule, the posterior probability for every label at each node is derived from the *a priori* probabilities and the observation-dependent likelihoods given by

$$P(\mathbf{r}_k) = l(\mathbf{r}_k)p(\mathbf{r}_k) \tag{10}$$

The product is comprised of likelihood $l(\mathbf{r}_k)$ of each label $\mathbf{r}_k$ of set $\mathcal{R}$ and its *a priori* probability density $p(\mathbf{r}_k)$, conditioned on its local neighbourhood. In the derivation of likelihood function it is assumed that at each node the observation components $\mathbf{r}_k$ are conditionally independent and have the same known conditional density function dependent only on that node.

At a given node, the label that yields maximum *a posteriori* (MAP) probability is chosen as the best continuation of the background at that node.

To optimise the MRF-based function defined in Eqn. (10), ICM is used since it is computationally efficient and avoids large scale effects[3] [24]. ICM maximises local conditional probabilities iteratively until convergence is achieved.

Typically, in ICM an initial estimate of the labels is obtained by maximising the likelihood function. However, in our framework an initial estimate consists of partial reconstruction of the background at nodes having just one label which is assumed to be the background. Using the available background information, the remaining unknown background is estimated progressively (see Section III-D).

At every node, the likelihood of each of its labels $\mathbf{r}_k$ $(k = 1, 2, \cdots, S)$ is calculated using corresponding weights W_k (see Section III-A). The higher the occurrences of a label, the more is its likelihood to be part of the background. Empirically, the likelihood function is modelled by a simple weighted function given by:

$$l(\mathbf{r}_k) = \frac{W_{c_k}}{\sum_{k=1}^{S} W_{c_k}} \tag{11}$$

where $W_{c_k} = \min(W_{max}, W_k)$ and $W_{max} = 5 \times$ frame rate of the captured sequence[4].

As evident, the weight W of a label greater than W_{max} will be capped to W_{max}. Setting a maximum threshold value is necessary in circumstances where the image sequence has a stationary foreground object visible for an exceedingly long period when compared to the background occluded by it. For example, in a 1000-frame sequence, a car might be parked for the first 950 frames and in the last 50 frames it drives away. In this scenario, without the cap the likelihood of the car being part of the background will be too high compared to the true background and this will bias the overall estimation process causing errors in the estimated background.

[2]To simplify the notations, index term (i,j) has been henceforth omitted.

[3]An undesired characteristic where a single label wrongly gets assigned to most of the nodes of the random field.

[4] It is assumed that the likelihood of a label exposed for a duration of 5 seconds is good enough to be regarded as a potential candidate for the background.

Relying on this likelihood function alone is insufficient since it may still introduce estimation errors even when the foreground object is exposed for just slightly longer duration compared to the background.

Hence, to overcome this limitation, the spatial neighbourhood modelled as Gibbs distribution (given by Eqn. (7)) is encoded into an *a priori* probability density. The formulation of the clique potential $V_c(\omega)$ referred in Eqn. (9) is described in the Section III-E. Using Eqns. (7), (8) and (9) the calculated clique potentials $V_c(\omega)$ are transformed into *a priori* probabilities. For a given label, the smaller the value of energy function, the greater is its probability in being the best match with respect to its neighbours.

In our evaluation of the posterior probability given by Eqn. (10), the local spatial context term is assigned more weight than the likelihood function which is just based on temporal statistics. Thus, taking log of Eqn. (10) and assigning a weight to the prior, we get:

$$\log\left(P(\mathbf{r}_k)\right) = \log\left(l(\mathbf{r}_k)\right) + \eta \log\left(p(\mathbf{r}_k)\right) \tag{12}$$

where η has been empirically set to number of neighbouring nodes used in clique potential calculation (typically η = 3).

The weight is required in order to address the scenario where the true background label is visible for a short interval of time when compared to labels containing the foreground. For example, in Figure 2, a sequence consisting of 450 frames was used to estimate its background. The person was standing as shown in Figure 2(i) for the first 350 frames and eventually walked off during the last 100 frames. The algorithm was able to estimate the background occluded by the standing person. It must be noted that pixel-level processing techniques are likely to fail in this case.

D. Node Initialisation

Nodes containing a single label in their representative set are directly initialised with that label in the background (see Figure 2(ii)). However, in some rare situations there is a possibility that all the sets may contain more than one label. In such a case, the algorithm heuristically picks the label having the largest weight W from the representative sets of the four corner nodes as an initial seed to initialise the background. It is assumed atleast one of the corner regions in the video frames corresponds to a static region.

The rest of the nodes are initialised based on constraints as explained below. In our framework, the local *neighbourhood system* [23] of a node and the corresponding cliques are defined as shown in Figure 3. A clique is defined as a subset of the nodes in the neighbourhood system that are fully connected. The background at an empty node will be assigned only if at least 2 neighbouring nodes of its 4-connected neighbours adjacent to each other and the diagonal node located between them are already assigned with background labels. For instance, in Figure 3, we can assign a label to node X if at least nodes B, D (adjacent 4-connected neighbours) and A (diagonal node) have already been assigned with labels. In other words, label assignment at node X is *conditionally*

independent of all other nodes given these 3 neighbouring nodes.

Node X has nodes D, B, E and G as its 4-connected neighbours. Let us assume that all nodes except X are labelled. To label node X the procedure is as follows. In Figure 3, four cliques involving X exist. For each candidate label at node X, the energy potential for each of the four cliques is evaluated independently given by Eqn. (13) and summed together to obtain its energy value. The label that yields the least value is likely to be assigned as the background.

Mandating that the background should be available in at least 3 neighbouring nodes located in three different directions with respect to node X ensures that the best match is obtained after evaluating the continuity of the pixels in all possible orientations. For example, in Figure 4, this constraint ensures that the edge orientations are well taken into account in the estimation process. It is evident from examples in Figure 4 that using either horizontal or vertical neighbours alone can cause errors in background estimation (particularly at edges).

Sometimes not all the three neighbours are available. In such cases, to assign a label at node X we use one of its 4-connected neighbours whose node has already been assigned with a label. Under these contexts, the clique is defined as two adjacent nodes either in the horizontal or vertical direction.

Typically, after initialising all the empty nodes an accurate estimate of the background is obtained. Nonetheless, in certain circumstances an incorrect label assignment at a node may cause an error to occur and propagate to its neighbourhood. Our previous algorithm [10] is prone to this type of problem. However, in the current framework the problem is successfully redressed by the application of ICM. In subsequent iterations, in order to avoid redundant calculations, the label process is carried out only at nodes where a change in the label of one of their 8-connected neighbours occurred in the previous iteration.

E. Calculation of the Energy Potential

In Figure 3, it is assumed that all nodes except X are assigned with the background labels. The algorithm needs to assign an optimal label at node X. Let node X have S labels in its state space $\mathcal{R}$ for $k = 1, 2, \cdots, S$ where one of them represents the true background. Choosing the best label is accomplished by analysing the spectral response of every possible clique constituting the unknown node X. For the decomposition we chose the Discrete Cosine Transform (DCT) [26] due to its decorrelation properties as well as ease of implementation in hardware. The DCT coefficients were also utilised by Wang et al. [27] to segment moving objects from compressed videos.

We consider the top left clique consisting of nodes A, B, D and X. Nodes A, B and C are assigned with background labels. Node X is assigned with one of S candidate labels. We take the 2D DCT of the resulting clique. The transform coefficients are stored in matrix $\mathbf{C}_k$ of size $M \times M$ ($M = 2N$) with its elements referred to as $C_k(v, u)$. The term $C_k(0, 0)$ (reflecting the sum of pixels at each node) is forced to 0 since we are interested in analysing the spatial variations of pixel values.

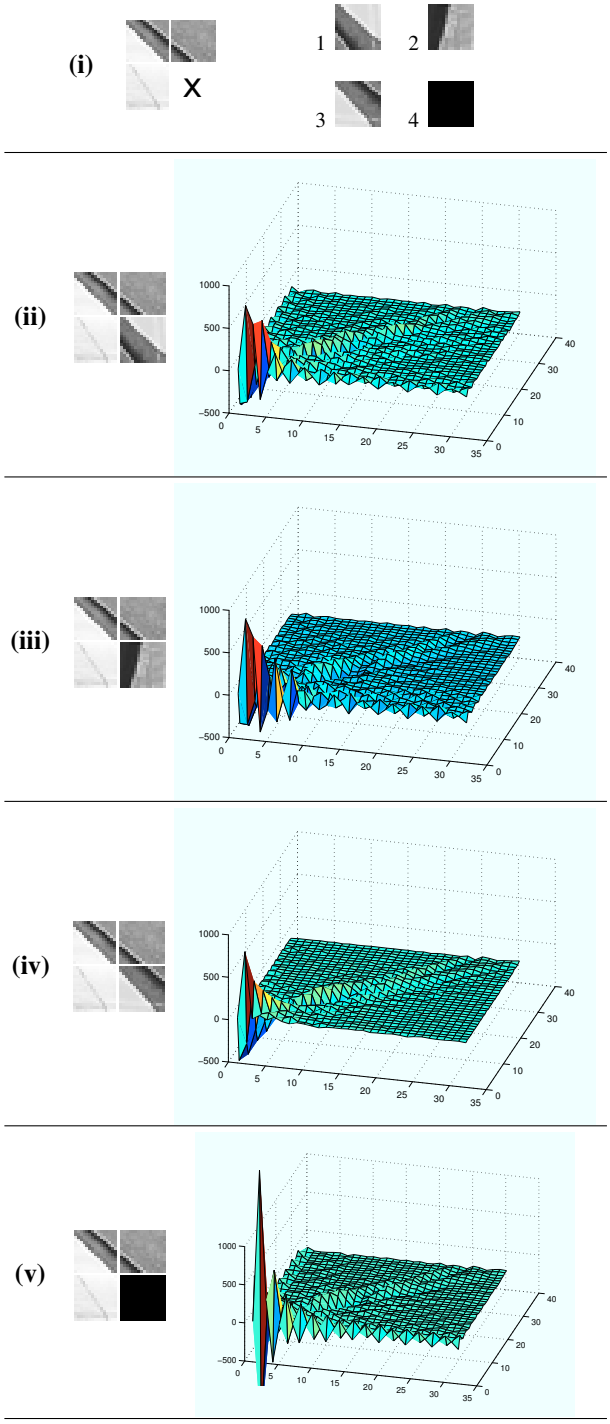


Fig. 5. An example of the processing done in Section III-E. **(i)** A clique involving empty node X with four candidate labels in its representative set. **(ii)** A clique and a graphical representation of its DCT coefficient matrix where node X is initialised with candidate label 1. The gaps between the blocks are for ease of interpretation only and are not present during DCT calculation. **(iii)** As per (ii), but using candidate label 2. **(iv)** As per (ii), but using candidate label 3. **(v)** As per (ii), but using candidate label 4. The smoother spectral distribution for candidate 3 suggests that it is a better fit than the other candidates.

Similarly, for other labels present in the state space of node X, we compute their corresponding 2D DCT as mentioned above. A graphical example of the procedure is shown in Figure 5.

Assuming that pixels close together have similar intensities, When the correct label is placed at node X, the resulting transformation has a smooth response (less high frequency components) when compared to other candidate labels.

The higher-order components typically correspond to high frequency image noise. Hence, in our energy potential calculation defined below we consider only the lower 75% of the frequency components after performing a zig-zag scan from the origin.

The energy potential for each label is calculated using:

$$V_c(\omega_k) = \left(\sum_{v=0}^{P-1} \sum_{u=0}^{P-1} |C_k(v,u)| \right) \qquad (13)$$

where $P = \text{ceil}\left(\sqrt{M^2 \times 0.75}\right)$ and ω_k is the local configuration involving label k. Similarly, the potentials over other three cliques in Figure 3 are calculated.

IV. Experiments

In our experiments the testing was limited to greyscale sequences. The size of each node was set to 16×16. The threshold $\mathcal{T}_1$ was empirically set to 0.8 based on preliminary experiments, discussed in subsection IV-A3. $\mathcal{T}_2$ (found automatically) was found to vary between 1 and 4 when tested on several image sequences ($\mathcal{T}_1$ and $\mathcal{T}_2$ are described in Section III-B).

A prototype of the algorithm using Matlab on a 1.6 GHz dual core processor yielded 17 fps. We expect that considerably higher performance can be attained by converting the implementation to C++, with the aid of libraries such as OpenCV [28] or Armadillo [29]. To emphasise the effectiveness of our approach, the estimated backgrounds were obtained by labelling all the nodes just once (no subsequent iterations were performed).

We conducted two separate set of experiments to verify the performance of the proposed method. In the first case, we measured the quality of the estimated backgrounds, while in the second case we evaluated the influence of the proposed method on a foreground segmentation algorithm. Details of both the experiments are described in Sections IV-A and IV-B, respectively.

A. Standalone Performance

We compared the proposed algorithm with a median filter based approach (i.e. applying filter on pixels at each location across all the frames) as well as finding intervals of stable intensity (ISI) method presented in [14]. We used a total of 20 surveillance videos: 7 obtained from CAVIAR dataset[5], 3 sequences from the abandoned object dataset used in the CANDELA project[6] and 10 unscripted sequences obtained from a railway station in Brisbane. The CAVIAR and and CANDELA sequences were chosen based on four criteria: **(i)** a minimum duration of 700 frames, **(ii)** containing significant background occlusions, **(iii)** the true background is available in at least one frame, and **(iv)** have largely static backgrounds. Having the true background allows for quantitative evaluation of the accuracy of background estimation. The sequences were resized to 320×240 pixels (QVGA resolution) in keeping with the resolution typically used in the literature.

The algorithms were subjected to both qualitative and quantitative evaluations. Subsections IV-A1 and IV-A2 respectively describe the experiments for both cases. Sensitivity of $\mathcal{T}_1$ is studied in subsection IV-A3.

1) Qualitative Evaluation: All 20 sequences were used for subjective evaluation of the quality of background estimation. Figure 6 shows example results on four sequences with differing complexities.

Going row by row, the first and second sequences are from a railway station in Brisbane, the third is from the CANDELA dataset and the last is from the CAVIAR dataset. In the first sequence, several commuters wait for a train, slowly moving around the platform. In the second sequence, two people (security guards) are standing on the platform for most of the time. In the third sequence, a person places a bag on the couch, abandons it and walks away. Later, the bag is picked up by another person. The bag is in the scene for about 80% of the time. In the last sequence two people converse for most of the time while others slowly walk along the corridor. All four sequences have foreground objects that are either dynamic or quasi-stationary for most of the time.

It can be observed that the estimated backgrounds obtained from median filtering (second column) and the ISI method (third column) have traces of foreground objects that were stationary for a relatively long time. The results of the proposed method appear in the fourth column and indicate visual improvements over the other two techniques. It must be noted that stationary objects can appear as background to the proposed algorithm, as indicated in the first row of the fourth column. Here a person is standing at the far end of the platform for the entire sequence.

2) Quantitative Evaluation: To objectively evaluate the quality of the estimated backgrounds we considered the test criteria described in [19], where the average grey-level error (AGE), total number of error pixels (EPs) and the number of "clustered" error pixels (CEPs) are used. AGE is the average of the difference between the true and estimated backgrounds. If the difference between estimated and true background pixel is greater than a threshold, then it is classified as an EP. We set the threshold to 20, to ensure good quality backgrounds. A CEP is defined as any error pixel whose 4-connected neighbours are also error pixels. As our method is based on region-level processing we calculated only the AGE and CEPs.

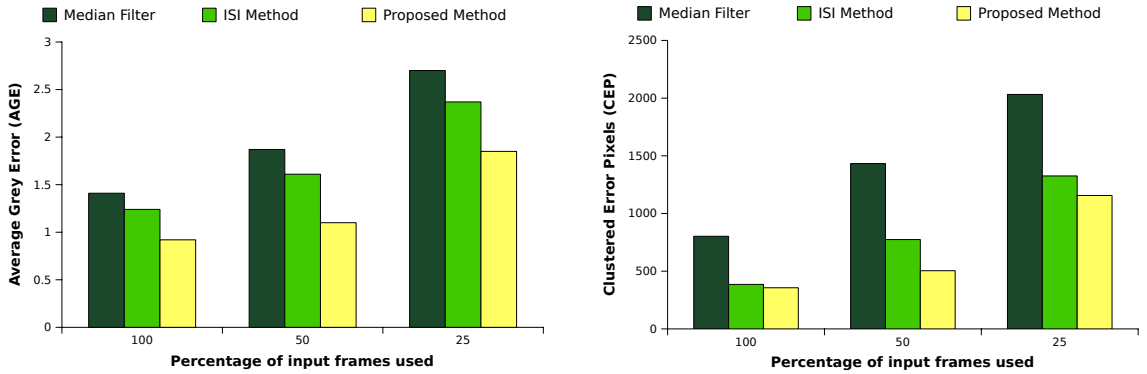
The Brisbane railway station sequences were not used as their true background was unavailable. The remaining 10 image sequences were used as listed in Table I. To maintain uniformity across sequences, the experiments were conducted using the first 700 frames from each sequence. The background was estimated in three cases. In the first case, all 700 frames (100%) were used to estimate the background. To

[5] http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/

[6] http://www.multitel.be/∼va/candela/

Fig. 6. **(i)** Example frames from four videos, and the reconstructed background using: **(ii)** median filter, **(iii)** ISI method [14], **(iv)** proposed method.

evaluate the quality when less frames are available (e.g. the background needs to be updated more often), in the second case the sequences were split into halves of 350 frames (50%) each. Each sub-sequence was used independently for background estimation and the obtained results were averaged. In the third case each sub-sequence was further split into halves (i.e., 25% of the total length). Further division of the input resulted in sub-sequences in which parts of the background were always occluded and hence were not utilised. The averaged AGE and CEP values in all three cases are graphically illustrated in Figure 7 and tabulated in Tables I and II. The visual results in Figure 6 confirm the objective results, with the proposed method producing better quality backgrounds than the median filter approach and the ISI method.



(i) **(ii)**

Fig. 7. Averaged values of AGE **(i)** and CEPs **(ii)** obtained by using 100%, 50% and 25% of the input sequences.

Sequence	case 1: 100%			case 2: 50%			case 3: 25%		
	Number of input frames = 700			Number of input frames = 350			Number of input frames = 175		
	median filter	ISI method	proposed method	median filter	ISI method	proposed method	median filter	ISI method	proposed method
m1.10_abandoned_object.avi	0.88	0.88	0.42	1.45	1.08	0.70	1.27	1.3	1.25
m1.16_abandoned_object.avi	2.02	1.69	1.93	2.06	2.03	2.25	2.38	2.36	2.65
m1.15_abandoned_object.avi	0.50	0.59	1.03	0.51	0.64	0.79	1.26	1.1	0.87
OneStopEnter1cor.mpg	0.99	0.98	0.85	0.50	0.39	0.59	0.65	0.63	0.73
OneStopEnter2cor.mpg	1.37	1.16	0.82	1.04	0.91	1.06	1.23	1.06	1.13
OneStopNoEnter1cor.mpg	0.90	0.96	0.21	0.56	0.92	0.42	1.65	1.44	0.49
OneStopNoEnter2cor.mpg	1.01	1.62	0.53	2.44	1.67	1.40	2.99	2.15	1.92
OneStopMoveEnter1cor.mpg	3.69	2.15	0.73	6.37	2.45	1.53	7.31	4.02	4.92
OneStopMoveNoEnter2cor.mpg	0.64	0.49	0.81	0.94	1.01	0.79	1.87	1.45	1.19
TwoEnterShop1cor.mpg	2.12	1.86	1.85	3.49	3.21	1.51	4.35	4.66	3.38
Average	**1.41**	**1.24**	**0.92**	**1.87**	**1.61**	**1.10**	**2.7**	**2.37**	**1.85**

TABLE I

AVERAGED GREY-LEVEL ERROR (AGE) RESULTS FROM EXPERIMENTS ON 10 IMAGE SEQUENCES. THE RESULTS UNDER CASE 2 AND CASE 3 (USING 50% AND 25% OF THE INPUT SEQUENCE, RESPECTIVELY) WERE OBTAINED BY AVERAGING OVER THE TWO AND FOUR SUB-SEQUENCES RESPECTIVELY.

Sequence	case 1: 100%			case 2: 50%			case 3: 25%		
	Number of input frames = 700			Number of input frames = 350			Number of input frames = 175		
	median filter	ISI method	proposed method	median filter	ISI method	proposed method	median filter	ISI method	proposed method
m1.10_abandoned_object.avi	258.00	208.00	0.00	976.50	423.50	133.50	664.75	673.25	660.75
m1.16_abandoned_object.avi	455.00	320.00	322.00	463.00	333.50	467.00	358.25	378	528.75
m1.15_abandoned_object.avi	0.00	95.00	86.00	0.00	92.00	38.00	773	521.75	135.25
OneStopEnter1cor.mpg	37.00	7.00	348.00	184.50	13.00	177.00	374.5	172.5	380.50
OneStopEnter2cor.mpg	358.00	85.00	29.00	482.00	230.50	266.00	640	351.25	374.50
OneStopNoEnter1cor.mpg	141.00	104.00	67.00	437.50	466.50	252.50	1224	819	286.25
OneStopNoEnter2cor.mpg	103.00	406.00	35.00	1919.50	854.00	678.00	2282.5	1224.25	1244.00
OneStopMoveEnter1cor.mpg	3931.00	1196.00	714.00	5756.00	2503.00	1289.50	8365.25	4622.25	3877.75
OneStopMoveNoEnter2cor.mpg	257.00	63.00	232.00	574.50	348.50	259.00	1169.25	697.75	654.50
TwoEnterShop1cor.mpg	2487.00	1372.00	1733.00	3534.00	2479.50	1483.00	4468.25	3795.5	3420.25
Average	**802.7**	**385.6**	**356.6**	**1432.75**	**774.4**	**504.40**	**2031.98**	**1325.55**	**1156.25**

TABLE II

AS PER TABLE I, BUT USING CLUSTERED ERROR PIXELS (CEPS) AS THE ERROR MEASURE.

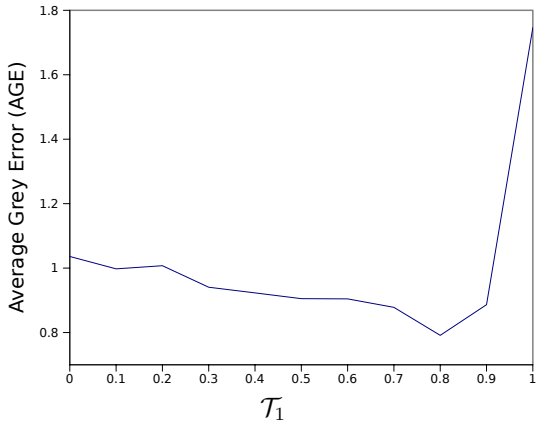


Fig. 8. Effect of $\mathcal{T}_1$ on AGE, while using a fixed value of $\mathcal{T}_2$.

Wallflower Sequence	**Relative improvement in *similarity* (Eqn. 14)**
WavingTrees	34%
ForegroundAperture	6%
LightSwitch	1%
Camouflage	20%
Bootstrap	62%
TimeOfDay	-23%
Average	**16.67%**

TABLE III
RELATIVE PERCENTAGE IMPROVEMENT IN FOREGROUND SEGMENTATION *similarity* (EQN. 14), OBTAINED ON THE WALLFLOWER DATASET, RESULTING FROM THE USE OF THE MRF-BASED PARAMETER ESTIMATION IN COMPARISON TO DIRECT PARAMETER ESTIMATION. THE SIMILARITY VALUE OF *moved object* SEQUENCE TURNS OUT TO BE ZERO (DUE TO THE ABSENCE OF TRUE POSITIVES IN ITS GROUND-TRUTH) AND IS THEREFORE NOT LISTED.

3) Sensitivity of $\mathcal{T}_1$: To find the optimum value of $\mathcal{T}_1$, we chose a random set of sequences from the CAVIAR dataset, whose true background was available a-priori and computed the averaged AGE between the true and estimated backgrounds for various values of $\mathcal{T}_1$ as indicated in Figure 8. As shown, the optimum value (minimum error) was obtained at $\mathcal{T}_1$ = 0.8.

B. Evaluation by Foreground Segmentation

In order to show the proposed method aids in better segmentation results, we objectively evaluated the performance of a segmentation algorithm (via background subtraction) on the Wallflower dataset. We note that the proposed method is primarily designed to deal with static backgrounds, while Wallflower contains both static and dynamic backgrounds. As such, Wallflower might not not optimal for evaluating the efficacy of the proposed algorithm in its intended domain, however it can nevertheless be used to provide some suggestive results as to the performance in various conditions.

For foreground object segmentation estimation, we use a Gaussian based background subtraction method where each background pixel is modeled using a Gaussian distribution. The parameters of each Gaussian (i.e., the mean and variance) are initialised either directly from a training sequence, or via the proposed MRF-based background estimation method (i.e. using labels yielding the maximum value of the posterior probability described in Eqn. (12) and their corresponding variances, respectively). The median filter and ISI [14] methods were not used since they do not define how to compute pixel variances of their estimated background.

For measurement of foreground segmentation accuracy, we use the similarity measure adopted by Maddalena and Petrosino [30], which quantifies how similar the obtained foreground mask is to the ground-truth. The measure is defined as:

$$similarity = \frac{tp}{tp + fp + fn} \tag{14}$$

where $similarity \in [0, 1]$, while tp, fp and fn are total number of true positives, false positives and false negatives (in terms of pixels), respectively. The higher the $similarity$ value, the better the segmentation result. We note that the *similiarity* measure is related to precision and recall metrics [31].

The parameter settings were the same as used for measuring the standalone performance (Section IV-A). The relative improvements in *similarity* resulting from the use of the MRF-based parameter estimation in comparison to direct parameter estimation are listed in Table III.

We note that each of the Wallflower sequences addresses one specific problem, such as dynamic background, sudden and gradual illumination variations, camouflage, and bootstrapping. As mentioned earlier, the proposed method is primarily designed for static background estimation (bootstrapping). On the 'Bootstrap' sequence, characterised by severe background occlusion we register a significant improvement of over 62%. On the other sequences, the results are only suggestive and need not always yield high *similarity* values. For example, we note a degradation in the performance on 'TimeOfDay' sequence. In this sequence, there is steady increase in the lighting intensity from dark to bright, due to which identical labels were falsely treated as 'unique'. As a result, estimated background labels variance appeared to be smaller than the true variance of the background, which in turn resulted in surplus false positives. Overall, MRF based background initialisation over 6 sequences achieved an average percentage improvement in *similarity* value of 16.67%.

C. Additional Observations

We noticed (via subjective observations) that all background estimation algorithms perform reasonably well when foreground objects are always in motion (i.e., in cases where the background is visible for a longer duration when compared to the foreground). In such circumstances, a median filter is perhaps sufficient to reliably estimate the background. However, accurate estimation by the median filter and the ISI method becomes problematic if the above condition is not satisfied. This is the main area where the proposed algorithm is able to estimate the background with considerably better quality.

The proposed algorithm sometimes mis-estimates the background in cases where the true background is characterised by strong edges while the occluding foreground object is smooth (uniform intensity value) and has intensity value similar to that of the background (i.e., low contrast between the foreground and the background). Under these conditions, the energy potential of the label containing the foreground object is smaller (i.e., smoother spectral response) than that of the label corresponding to the true background.

From our experiments we found the memory footprint to store the state space of all the nodes is on average only 5% of the memory required for storing all the frames. This is in contrast to existing algorithms, which typically require the storage of all the frames before processing can begin.

We conducted additionally experiments on image sequences represented in other colour spaces, such as RGB and YUV, and evaluated the overall posterior as the sum of individual posteriors evaluated on each channel independently. The results were marginally better than those obtained using greyscale input. We conjecture that this is because the spatial continuity of structures within a scene are well represented in greyscale.

V. Main Findings and Future Work

In this paper we proposed a background estimation algorithm in an MRF framework that is able to accurately estimate the static background from cluttered surveillance videos containing image noise as well as foreground objects. The objects may not always be in motion or may occlude the background for much of the time.

The contributions include the way we define the neighbourhood system, the cliques and the formulation of clique potential which characterises the spatial continuity by analysing data in the spectral domain. Furthermore, the proposed algorithm has several advantages, such as computational efficiency and low memory requirements due to sequential processing of frames. This makes the algorithm possibly suitable for implementation on embedded systems, such as smart cameras [32], [1].

The performance of the algorithm is invariant to moderate illumination changes, as we consider only AC coefficients of the DCT in the computation of the energy potential defined by Eqn. (13). However, the similarity criteria defined by Eqns. (3) and (4) creates multiple representatives for the same visually identical block. Tackling this problem efficiently is part of further research. We also intend to extend this work to estimate background models of non-static backgrounds.

Experiments on real-life surveillance videos indicate that the algorithm obtains considerably better background estimates (both objectively and subjectively) than methods based on median filtering and finding intervals of stable intensity. Furthermore, segmentation of foreground objects on the Wallflower dataset was also improved when the proposed method was used to initialise the background model based on a single Gaussian. We note that the proposed background estimation algorithm can be combined with almost any foreground segmentation technique, such as [8], [33].

Acknowledgements

The authors thank Prof. Terry Caelli for useful discussions and suggestions. NICTA is funded by the Australian Government via the Department of Broadband, Communications and the Digital Economy, as well as the Australian Research Council through the ICT Centre of Excellence program.

References

[1] W. Wolf, B. Ozer, and T. Lv. Smart cameras as embedded systems. *Computer*, vol. 35, no. 9, pp. 48–53, 2002.

[2] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa. A system for video surveillance and monitoring. Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-00-12, May 2000.

[3] C. Sanderson and B. C. Lovell. Multi-region probabilistic histograms for robust and scalable identity inference. *International Conference on Biometrics, Lecture Notes in Computer Science (LNCS)*, vol. 5558, 2009, pp. 199–208.

[4] S. Cheung and C. Kamath. Robust techniques for background subtraction in urban traffic video. *Proceedings of SPIE*, vol. 5308, 2004, pp. 881–892.

[5] M. Piccardi. Background subtraction techniques: a review. *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, 2004, pp. 3099–3104.

[6] M. Heikkila and M. Pietikainen. A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 657 –662, april 2006.

[7] M. Vargas, M. Milla, L. Toral, and F. Barrero. An Enhanced Background Estimation Algorithm for Vehicle Detection in Urban Traffic Scenes. *IEEE Transactions on Vehicular Technology*, vol. 59, no. 99, pp. 3694–3709, 2010.

[8] T. Matsuyama, T. Wada, H. Habe, and K. Tanahashi. Background subtraction under varying illumination. *Systems and Computers in Japan*, vol. 37, no. 4, p. 77, 2006.

[9] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. *International Conference on Computer Vision (ICCV)*, vol. 1, 1999, pp. 255–261.

[10] V. Reddy, C. Sanderson, and B. C. Lovell. An efficient and robust sequential algorithm for background estimation in video surveillance. *IEEE International Conference on Image Processing (ICIP), Egypt*, 2009.

[11] B. Lo and S. Velastin. Automatic congestion detection system for underground platforms. *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing*, 2001, pp. 158–161.

[12] W. Long and Y. Yang. Stationary background generation: An alternative to the difference of two images. *Pattern Recognition*, vol. 23, no. 12, pp. 1351–1359, 1990.

[13] A. Bevilacqua. A novel background initialization method in visual surveillance. *IAPR Workshop on Machine Vision Applications*, Nara, Japan, 2002, pp. 614–617.

[14] H. Wang and D. Suter. A Novel Robust Statistical Method for Background Initialization and Visual Surveillance. *ACCV 2006, Lecture Notes in Computer Science*, vol. 3851/2006, pp. 328–337, 2006.

[15] K. Kim, T. Chalidabhongse, D. Harwood, and L. Davis. Real-time foreground–background segmentation using codebook model. *Real-Time Imaging*, vol. 11, no. 3, pp. 172–185, 2005.

[16] C. Chiu, M. Ku, and L. Liang. A Robust Object Segmentation System Using a Probability-Based Background Extraction Algorithm. *IEEE Transactions on circuits and systems for video technology*, vol. 20, no. 4, pp. 518–528, 2010.

[17] D. Farin, P. de With, and W. Effelsberg. Robust background estimation for complex video sequences. *IEEE International Conference on Image Processing (ICIP)*, vol. 1, 2003, pp. 145–148.

[18] A. Colombari, A. Fusiello, and V. Murino. Background Initialization in Cluttered Sequences. *CVPRW*, Washington DC, USA, 2006, pp. 197–202.

[19] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. Jain, "A background model initialization algorithm for video surveillance. *International Conference on Computer Vision (ICCV)*, vol. 1, 2001, pp. 733–740.

[20] S. Cohen. Background estimation as a labeling problem. *International Conference on Computer Vision (ICCV)*, vol. 2, 2005, pp. 1034–1041.

[21] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *International Conference on Computer Vision (ICCV)*, vol. 1, 1999, pp. 377–384.

[22] X. Xu and T. Huang. A Loopy Belief Propagation approach for robust background estimation. *CVPR*, 2008, pp. 1–7.

[23] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 721–741, 1984.

[24] J. Besag. On the statistical analysis of dirty images. *Journal of Royal Statistics Society*, vol. 48, pp. 259–302, 1986.

[25] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 32, no. 2, pp. 192–236, 1974.

[26] N. Ahmed, T. Natarajan, and K. Rao. Discrete Cosine Transfom. *Transactions on Computers*, vol. 100, no. 23, pp. 90–93, 1974.

[27] W. Wang, J. Yang, and W. Gao. Modeling background and segmenting moving objects from compressed video. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, p. 670, 2008.

[28] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 2008.

[29] C. Sanderson. Armadillo: An open source C++ linear algebra library for fast prototyping and computationally intensive experiments. NICTA, Tech. Rep., 2010, http://arma.sourceforge.net

[30] L. Maddalena and A. Petrosino. A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications. *IEEE Transactions on Image Processing*, vol. 17, pp. 1168–1177, 2008.

[31] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. *International conference on Machine learning (ICML)*. ACM, 2006, pp. 233–240.

[32] Y. Mustafah, A. Bigdeli, A. Azman, and B. Lovell. Smart cameras enabling automated face recognition in the crowd for intelligent surveillance system. *Recent Advances in Security Technology (RNSA)*, 2007, pp. 310 – 318.

[33] V. Reddy, C. Sanderson, A. Sanin, and B. C. Lovell. Adaptive patch-based background modelling for improved foreground object segmentation and tracking. *Advanced Video and Signal Based Surveillance (AVSS)*, 2010, pp. 172–179.