Exploiting Multiple Cues in Motion Segmentation based on Background Subtraction

Ivan Huerta[a], Ariel Amato[b], Xavier Roca[b], Jordi Gonzàlez[b]

[a] *Institut de Robòtica i Informàtica Industrial (CSIC-UPC),*
Parc Tecnològic de Barcelona, Llorens i Artigas 4-6, 08028 Barcelona, Spain
Tel: 0034 93 4015751, e-mail: ihuerta@iri.upc.edu
[b] *Computer Vision Center & Department of Computer Science*
Edifici O, Campus Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain
{aamato, xavir, poal}@cvc.uab.cat

Abstract

This paper presents a novel algorithm for mobile-object segmentation from static background scenes, which is both robust and accurate under most of the common problems found in motion segmentation. In our first contribution, a case analysis of motion segmentation errors is presented taking into account the inaccuracies associated with different cues, namely colour, edge and intensity. Our second contribution is an hybrid architecture which copes with the main issues observed in the case analysis by fusing the knowledge from the aforementioned three cues and a temporal difference algorithm. On one hand, we enhance the colour and edge models to solve not only global and local illumination changes (i.e. shadows and highlights) but also the camouflage in intensity. In addition, local information is also exploited to solve the camouflage in chroma. On the other hand, the intensity cue is applied when colour and edge cues are not available because their values are beyond the dynamic range. Additionally, temporal difference scheme is included to segment motion where those three cues can not be reliably computed, for example in those background regions not visible during the training period. Lastly, our approach is extended for handling ghost detection. The proposed method obtains very accurate and robust motion segmentation results in multiple indoor and outdoor scenarios, while outperforming the most-referred state-of-art approaches.

Keywords:
Motion segmentation, shadow suppression, colour segmentation, edge

segmentation, ghost detection, background subtraction

1. Introduction

During the last decades, important research efforts in Computer Vision have been focused on developing theories, methods and systems for modelling and understanding motion perception. Motion information is the basis for a wide range of applications such as smart surveillance systems, control applications, advanced user interfaces, and motion basis diagnosis, among others [1]. A particular domain-of-interest can be found in the semantic evaluation of people behaviour in image sequences, in which different tasks are required, such as acquisition, detection, tracking, action recognition and behaviour reasoning [2], but still the basis for this high-level interpretation relies on when and where motion has been detected in an image.

Detecting moving agents is a fundamental and difficult problem because an accurate segmentation enhances the performance of the subsequent steps of human behaviour understanding [3] . But reliable (and fast) motion segmentation is hard due to the intrinsic nature of typical scenarios, where global and local illumination changes (i.e. shadows and highlights), camouflage and repetitive moving regions (like waving flags or tree leaves) should be commonly addressed, as well as other physical changes such as bootstrapping and *ghosts* [4]. Most used techniques for handling these issues are background subtraction, frame differencing, a combination of both, or optical flow [1, 5, 6, 7]. Even though many algorithms have been proposed in the literature, the detection of moving objects in complex environments is still far from being completely solved.

In this paper, a novel approach which outperforms most-known techniques used for motion segmentation is proposed. The main contributions of this paper are: (i) a novel, deep theoretical case analysis is presented for the cues most used in the literature for motion segmentation; we analyse when and why inaccuracies and errors appear due to these cues. (ii) A new hybrid architecture is presented based on such an analysis; we exploit the benefits of fusing colour, edge, and intensity cues together with temporal difference [8, 9]. (iii) New colour and edge models are proposed; in particular a novel chromatic-invariant cone model for colour segmentation, and an invariant gradient model which fuses magnitude and orientation for edge segmentation (thus avoiding false edges due to intense global illumination changes).

Regarding with the performance of our method, it can handle (i) non-physical changes (such as global or local illumination changes and camouflages), (ii) physical changes (such as bootstrapping and ghosts), and (iii) sensor dynamic range problems. In particular, *Ghost* cases are successfully detected on-the-fly without increasing the computational cost. Furthermore, our approach can detect dark camouflage cases which can be distinguished from shadows and changes of global illumination. Interestingly, real-time performance can be fulfilled because the method is highly parallelizable, mainly due to the pixel-wise nature of the proposed components.

This paper is organized as follows: next section presents a comprehensive literature review in motion segmentation and compares the main contributions of our approach w.r.t. the state of the art. Section 3 presents an analysis of the cues typically applied in motion segmentation, emphasizing those cases in which such cues can not be reliably applied. This theoretical evaluation leads to our segmentation approach which is explained in section 4: we explain how intensity, colour, edge cues and temporal difference are used together for addressing most of the identified anomaly cases. The experimental results described in section 5 demonstrates that the performance of the resulting method applied to both indoor and outdoor sequences of several, most popular databases outperforms most well-known segmentation approaches. Lastly, section 6 presents the main lines of future research based on the results obtained.

2. State of the Art

Background subtraction is the most commonly used technique for motion segmentation in static scenes [10, 11, 12, 13]. It attempts to detect moving regions in an image by subtracting the current image with a reference background model in a pixel-by-pixel manner. The background representation is created by averaging several images over time during an initialization period. Subsequently, pixels are classified as foreground if the difference between the input image and the background model is above a learnt threshold, whose calculation depends on the specific approach. Then, numerous approaches update over time the background model with new images to adapt it to dynamic scene changes.

There are a large number of different algorithms using this background subtraction scheme. Nonetheless, they differ in (i) the type of cues or structures employed to build the background representation; (ii) the procedure used for detecting the foreground region; and (iii) the updating criteria of the background model.

A naive version of the background subtraction scheme is employed by Heikkila and Silven [14], which classifies an input pixel as foreground if its value is over a predefined threshold when subtracted from the background model. This approach updates the background model in order to guarantee reliable motion detection using a first order recursive filter. However, this method is extremely sensitive to changes of dynamic scenes such as gradual illumination variation or physical changes such as *ghosts* (i.e. when an object already represented in the background model begins to move).

In order to overcome these difficulties, statistical approaches have been applied [5]. These approaches make use of statistical properties of each pixel (or regions), which are updated dynamically during all the process in order to construct the background model. It has been demonstrated that statistical approaches are more efficient when dealing with noise, illumination changes, shadows, etc.

Haritaoglu et al. in W^4 [15] apply background subtraction by computing for each pixel in the background model, during a training period, three values: its minimum and maximum intensity values, and the maximum intensity difference between consecutive frames. Background model pixels are updated using pixel-based and object-based updating conditions to be adaptive to illumination and physical changes in the scene. However, this approach is rather sensitive to shadows and lighting changes, since the only cue used is intensity.

Alternatively, Wren et al. in Pfinder [16] proposed a modelling framework in which each pixel colour value (in YUV space) is represented with a single Gaussian. Then, model parameters are recursively updated. However, a single Gaussian model cannot handle multiple backgrounds, such as waving trees. Stauffer and Grimson [17, 18] addressed this issue by using a Mixture of Gaussians (MoG) to build a background colour model for every pixel. An improvement of the MoG can be found in Zivkovic et al. [19, 20], where the parameters of a MoG model are constantly updated, while selecting simultaneously the appropriate number of components for each pixel.

Elgammal et al. [21] use a non-parametric Kernel Density Estimation (KDE) to model the background. Their representation samples an intensity values for each pixel to estimate the probability of newly observed intensity values. The background model is also updated continuously to be adaptive to background changes. In addition to colour-based information, their system incorporates region-based scene knowledge for matching nearby pixel locations. This approach can successfully handle the problem of small background motion such as tree branches. Mittal et al. [22] use adaptive KDE for modelling background in motion, and implement optical flow to detect moving regions. In this way, their approach is able to manage complex background, although the computational cost is severe. Chen et al. [23] combine pixel- and block-based approaches to model complex background; however the method is very sensitive to camouflages and shadows.

Cheng et al. in [24] proposed an on-line learning method which is able to work in real time and can be implemented in GPU which also gives similar results managing complex background. In [25] Barnich and Droogenbroeck also present a really fast method that can cope with background in motion and bootstrapping problems. The method adopts the idea of sampling the spatial neighbourhood for refining the per-pixel estimation. The model updating relies on a random process that substitutes old pixel values with new ones. However, it can not cope with camouflages and shadows. Another solution to bootstrapping problem is presented by colombari et al. in [26], where a patch-based technique exploits both spatial and temporal consistency of the static background.

Li et al. [27] and Sheikh et al. [28] use Bayesian networks to cope with dynamic backgrounds. Li et al. uses a Bayesian framework that incorporates spectral, spatial, and temporal features to characterize background appearance. Sheik et al. apply non-parametric density estimation to model the background as a single distribution, thus handling multimodal spatial uncertainties. Furthermore, they also use temporal information. The use of layers for image decomposition based on the neighbouring pixels is presented in [29] to handle dynamic backgrounds as well. Maddalena et al. [30] use neural networks to overcome the same problem. An improvement of it using self organizing maps can be found by Lopez-Rubio et al. [31] which can adapt its colour similarity measure to the characteristics of the input video. Mahadevan et al. in [32] uses a combination of the discriminant center-surround saliency framework with the modelling power of dynamic textures to solve problems with highly dynamic backgrounds and a moving camera. However, this method is not designed for high accurate segmentation but rather for detection.

Toyama et al. [4] in Wallflower use a three-component system to handle many canonical anomalies for background updating. Their work processes input images at various spatial scales, namely pixel, region, and frame levels.

Reasonably good foreground detection can be achieved when moving objects or strong illumination changes (for example when turning on/off the light in an indoor scene) are present. However, it fails when modelling small motion in the background or local illumination variations.

The aforementioned motion detection approaches generally obtain good segmentation in indoor and outdoor scenarios so some of them have been used in real-time surveillance applications for years. Nevertheless, most of them are susceptible to both local (such as shadows and highlights) and global illumination changes (like at dawn or dusk, and when the sun is suddenly covered by clouds). Towards this end, another approaches have been proposed [33], which differ in the type of cue considered.

Horprasert et al. [34] present a statistical background colour model which uses colour chrominance and brightness distortion in RGB space. Using these distortions, their approach classifies the current pixel as background (shaded, shadow or highlight) or moving foreground. An enhancement of this work was presented by Kim et al. [35] who built a cylinder region in RGB colour space to detect foreground objects. They also quantized background values for each pixel into codebooks which represent a compressed form of the background model for long image sequences. Nevertheless, anomalies in the dynamic range prevent to obtain accurate segmentation.

To avoid shadows, other spaces are used such as that presented in Cucchiara et al. [36]. They use the HSV space colour model to avoid local illumination problems. An extension ispresented in [37] where a more complex model is used to detect shadows and ghosts, thereby classifying the pixels as moving object, uncovered background, background, ghost, or shadow.

So colour has been demonstrated to be a suitable cue to handle problems with local and global illumination changes. Nevertheless, there are a lot of problems when colour is used, such as the change of illuminant, the nonlinear sensor response, etc. Two main approaches are employed in order to deal with these two problems, namely colour constancy and colour invariant normalisations.

Based on colour constancy methods, some approaches make use of intrinsic images to remove shadows while coping with illuminant variation. Intrinsic image decomposition separates one image into two: one which records variation in reflectance and another which represents the variation in the illumination across the image. Given that, Finlayson et al. [38] compute an invariant image which depends only on the reflectance. Hence, their approach is invariant to the changes in illuminant colour and intensity. Nonetheless, part of the colour information is lost when removing the effect of the scene illumination at each pixel in the image, thereby increasing the problem of camouflages. Other approaches such as [39] use the bluish effect from the illuminant scene plus a spatio-temporal ratio test and a dichromatic reflection model for shadow removal.

Weiss [40] also extract intrinsic images using edge cues instead of colour to obtain the reflectance image. This process requires several frames to determine the reflectance edges of the scene. However, the reflectance image also contains scene illuminations because this approach requires prominent changes in the scene, specifically for the position of shadows.

There are other approaches which use different techniques to eliminate local illuminations (e.g. shadows), such as normalised cross correlation. However, these techniques are not usually applied because of their problems with camouflages.

Statistical learning-based approaches have been developed to learn and remove cast shadows [41, 42, 43]. For example, in [42] a nonparametric framework to model surface behaviour when shadows are cast on them is presented. Physical properties of light sources and surfaces are employed in order to identify a direction in RGB space at which background surface values under cast shadows are found. However, these approaches are particularly affected by the training phase.

Edge cues are also used for motion segmentation. Jabri et al. [44] use a statistical background modelling which combines colour (in RGB space) with edges. Subsequently, background subtraction is performed by subtracting the colour and edge channels separately using confidence maps, and then combining the results to get the foreground pixels. McKenna et al. [45] also use colour and edge information to model the background. In this case, motion segmentation consists of three separate background models which are combined to obtain the foreground pixels. Javed et al. [46] present a method that uses multiple cues, based on colour and gradient information. The approach tries to handle different difficulties by using three distinct levels: pixel, region and frame level, inspired from [4]. Nevertheless, *ghosts* can not be eliminated if the background contains a high number of edges, and shadows can not be removed either.

Alternative approaches use textures for shadow removal. Leone et al. [47] use a descriptor based on the coefficients of Gabor decomposition and photometric properties. Heikkil*ä* et al. [48] apply a modified Local Binary Pattern (LBP) operator. In Yao et al. [49], textures are computed using the LBP combined with a RGB colour model. However, texture-based approaches usually fails in handling camouflages and local illumination changes. In [50] Amato et al. present a method which introduces two discriminative features based on angular and modular patterns which are formed by similarity measurement between two sets of RGB colour vectors. However, in this work issues as camouflage and ghost are not addressed.

Approaches based on temporal difference extract moving regions by making use of a pixel-by-pixel difference between consecutive frames [5]. This methodology is very adaptive to dynamic scene changes but it can not generally extract the entire pixels of moving objects, thereby causing the so-called foreground aperture problem. Temporal differencing cannot cope with *sleeping objects*, so additional techniques have been proposed for detecting motionless foreground objects. For example, Shen [51] is an example of hybrid algorithm which combines RGB, HSI colour spaces, fuzzy information and temporal difference.

Lastly, Optical Flow (OF) has been used for motion segmentation as well: flow vectors of moving objects are computed over time to detect changing regions [22]. OF techniques are able to segment moving objects in video sequences even from moving cameras. The main drawback is that these methods are computationally highly expensive and very sensitive to noise. Therefore, most of them cannot be executed in real-time without specialized hardware [5].

2.1. Main contributions of this paper

Once the advantages and drawbacks of the most referred approaches have been detailed, we next detail the main contributions of our approach w.r.t. the state of the art:

- A novel theoretical case analysis of motion segmentation problems is presented, where the performance of each cue used in the literature for segmentation (intensity, colour, and edges) is exhaustively evaluated, showing the advantages of every cue and stating when each cue can or cannot be applied. To the best of our knowledge, current state-of-the-art considers chromatic spaces only, so literature still do not address most of the problems identified in our theoretical case analysis.

- Our hybrid algorithm uses intensity, colour, edges cues and temporal difference, because each cue solves a particular problem identified in the case analysis. Cue models have been improved over the existing ones and moreover their combination is a step forward in current state-of-the-art, since:

- Our proposed chromatic-invariant cone model achieves better segmentation results than the commonly-used cylinder model [34, 35]. The invariant gradient model combines magnitudes and orientations for edge segmentation while avoiding false edges due to intense global illumination changes [45, 44].
- Some techniques are not be able to work with global and local illumination changes [17, 15, 16]. Using chromaticity only, the assessment of whether a foreground region is a shadow, a change of global illumination or a dark camouflage is not possible [34, 35]. Approaches using HSV [36, 51] also exhibit this problem. Another approaches fuse colour and edges cues without addressing shadow removal [44, 46]. We cope with all these problems without a significant increase of computational cost.
- Using the combination of cues, we are able to address the ghost problem on-the-fly, instead of requiring a predefined time period or not handling this problem [34, 35, 21, 4, 45, 44].

Summarizing, the main contributions are (i) the case analysis of segmentation problems, and (ii) the architecture derived from such analysis; (iii) improved versions of the cue models typically used for segmentation, while detailing (iv) how these cues are combined, when are these used, and why each cue solves each anomaly case. The algorithms presented in this paper for each cue can be enhanced independently or rather substituted by better ones without modifying the architecture itself.

3. A Case Analysis of motion segmentation problems

Colour Information obtained from a recording camera is based on the sensor response s^c —for Lambertain or perfect matte surfaces— and depends on three components: the illuminant spectral power distribution $L\left(\lambda\right)$, the object reflectance distribution $R\left(\lambda\right)$, and the sensor sensitivity $S^c\left(\lambda\right)$, following the equation $s^c = \int_\lambda L\left(\lambda\right) R\left(\lambda\right) S^c\left(\lambda\right) d\lambda$, where λ denotes the wavelength, and $c \in \{R, G, B\}$ the colour channel. Therefore, changes in the illumination —in both brightness and chrominance components— modify the sensor

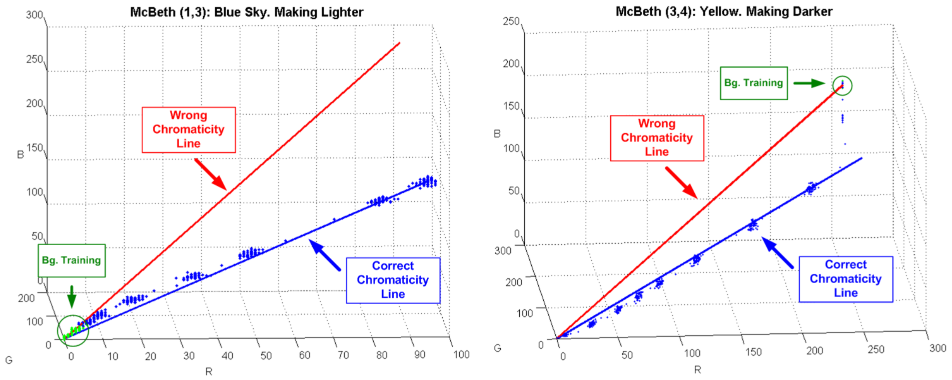


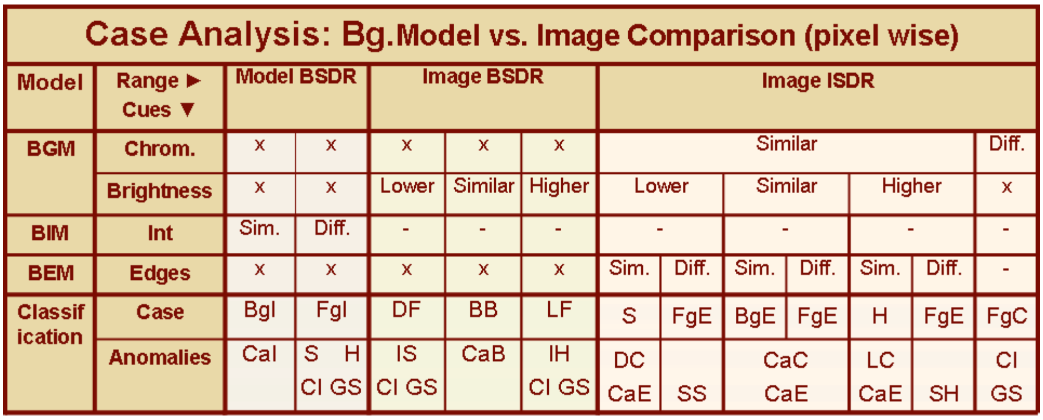
Figure 1: Experiments on a Macbeth board to test the sensor dynamic range. Background pixels are drawn in green. The red line denotes the modelled chrominance line, whereas the blue one corresponds to the correct one. First image corresponds to a blue checker which is not observed with enough light during the modelling process. In second image case, the chrominance of a yellow checker is modelled while some of the channels are saturated. Consequently, there are noticeable deviations between the inferred and correct chrominance in both cases.

response. The object reflectance may considerably depend on the both the incident-light angle and the viewing angle. It also may present strong specular components with no information about the object colour. Finally, it depends on the sensor sensitivity.

In addition, the sensor dynamic range must be taken into account. This is defined as the ratio between the maximum possible signal versus the noise signal in dark. Thus, very low or very high brightness distort the observed response. Consequently, these effects should be considered as a source of potential errors during both background modelling and image segmentation.

The measurement of the low intensity pixels are affected by quantization noise which it make unstable region to describe their chromaticity as well high intensity pixels are affected by the saturation of the sensor.

A series of experiments with a Macbeth board were designed to explore these phenomena, see Fig. 1. Experiments show as a wrong background model may be built depending on the illumination conditions during the training step of the background model (red line in Fig. 1). A Macbeth board was first illuminated with a constant light source. Then, the diaphragm was modified in a series of time steps, thereby changing the received luminance. The background was modelled during 50 frames. Then, 650 more frames were acquired while changing the aperture.



Case Analysis: Bg.Model vs. Image Comparison (pixel wise)

Model	Range ▶ Cues ▼	Model BSDR		Image BSDR			Image ISDR						
BGM	Chrom.	x	x	x	x	x	Similar						Diff.
	Brightness	x	x	Lower	Similar	Higher	Lower		Similar		Higher		x
BIM	Int	Sim.	Diff.	-	-	-	-		-		-		-
BEM	Edges	x	x	x	x	x	Sim.	Diff.	Sim.	Diff.	Sim.	Diff.	-
Classification	Case	BgI	FgI	DF	BB	LF	S	FgE	BgE	FgE	H	FgE	FgC
	Anomalies	CaI	S H CI GS	IS CI GS	CaB	IH CI GS	DC CaE	SS	CaC CaE		LC CaE	SH	CI GS

Figure 2: Labelling: Beyond (BSDR) or Inside (ISDR) Sensor Dynamic Range; Shadow (S), Highlight (H), Background using Chrominance (BgC), Brightness (BB), Edges (BgE) or Intensity cues (BgI); Foreground using Chrominance (FgC), Edges (FgE) or Intensity (FgI), Dark Foreground (DF) and Light Foreground (LF); Camouflage using Chrominance (CaC), Edges (CaE), Brightness (CaB) or Intensity (CaI), Dark Camouflage (DC), Light Camouflage (LC), Sharp Shadows (SS), Sharp Highlight (SH), Intense Shadows (IS), Intense Highlight (IH), Change of Illuminant (CI), Gleaming Surface (GS). Cues: **'x'** it cannot be used; **'-'** it is no relevant. See text for details.

Fig. 2 shows a case analysis of the potential segmentation problems using the combination of three background models: colour, edges and intensity, and the pixel value within the sensor dynamic range.

Edges from very dark pixels with not enough brightness can be hidden since they are beyond sensor dynamic range. And a similar problem appears with very light pixels. Consequently, cases beyond sensor dynamic range should be addressed using an intensity model, because both colour and edge models are not suitable, thereby classifying the pixels as foreground (case FgI) or background (case BgI) depending on their intensity.

There could be pixels whose Bg. colour model can be computed, although the current image pixels are beyond the sensor dynamic range. Here, neither chrominance nor edge cues can be used. In such a case, the brightness component of the colour model can be used as a suitable cue, thereby classifying them as dark/light foreground (case DF/LF) or background (case BB).

Changes in illumination, despite of being local or global, sudden or gradual (such as shadows or highlights) are all supposed to entail just variations in the observed brightness, but not in the chrominance. Thus, a pixel can be considered as foreground using colour and edge models in the following situations: (i) a pixel is considered foreground using the colour model when it differs in chrominance with the model (case FgC); (ii) using the edge model when it shows a gradient change respect to the model (case FgE). Otherwise the pixel is classified as background (cases BgE, shadow (S) or highlight (H)).

Foreground pixels whose lower and higher brightness cannot be distinguished from shadows and highlights, are considered dark/light camouflage (DC/LC, respectively).

Hence, fusing the three models may overcome some of the segmentation problems such as changes in illumination conditions, camouflage in intensity and camouflage in chroma, as long as the illuminant has a plain spectral power distribution.

However, there are other *anomalies* that cannot be disambiguated with the colour, edges and intensity cues, which are not taken into account in this paper. Firstly, foreground pixels with the same chrominance, brightness, and gradient as the background model can not be segmented, so such pixels are considered camouflaged (CaC, CaB, and CaE respectively). Secondly, intense shadows and highlights (IS/IH) can be classified as DF or LF, and shadows and highlights (S/H) over zones beyond the sensor dynamic range can be considered as foreground (FgI). Thirdly, edges of sharp shadows and highlights (SS, SH) can be segmented (FgE). Finally, local and global changes in the illuminant chrominance (CI), as well as gleaming surfaces (GS) may cause false-positive segmentations. So there is still a lot of ground to cover.

4. Multicue Image Segmentation

The segmentation task is next presented following a statistical background-subtraction approach based on the case analysis as discussed before. Our approach addresses the aforementioned cases by combining background models built on three different cues, and a temporal difference technique.

Firstly, background models are built and an automatic threshold selection is computed for them. Next, image segmentation using these models is presented and finally, an approach which combines these models is detailed.

4.1. Background Modelling

The approach combines three background models and a temporal difference algorithm. A sketch of the Background-Modelling Module is shown in Fig 3. A Background Colour Model (BCM) consists of a chromatic invariant

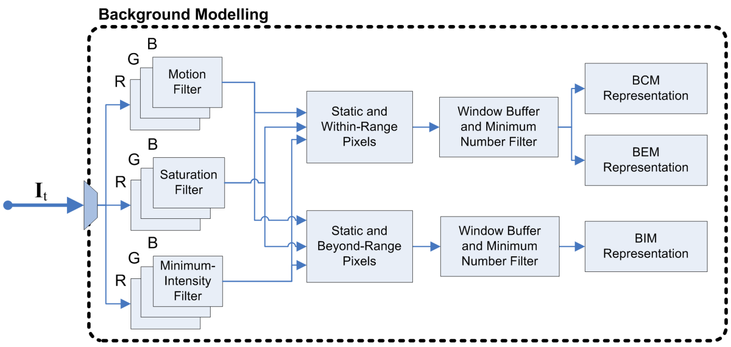


Figure 3: Background modelling approach. See text for details.

cone representation which separates the chrominance and brightness components; a Background Edge Model (BEM) makes use of invariant gradient magnitudes and orientations; a Background Intensity Model (BIM) computes the mean and standard deviation for each pixel intensity; and a Temporal Differencing (TD) algorithm evaluates the changes between three consecutive frames. We next detail the procedure.

Background is modelled on a pixel-wise basis [17, 15, 45], which provides the necessary representation accuracy. Training is carried out by using a window of T frames. A motion filter $\left|I^c_{a,t} - \widetilde{I}^c_a\right| < \max\left(2\sigma^c_a, \epsilon\right)$ is used to remove moving pixels during a training period of T frames, where $I^c_{a,t}$ and $\widetilde{I}^c_a$ are the current image value and median value of pixel ‘a‘ for each channel $c \in \{R, G, B\}$ respectively, σ^c_a is the corresponding standard deviation, and ϵ is a small positive quantity. This process is iterated until convergence.

Then, those pixels with a representative number of valid values in the training period are taken into account for building the background model. Values of colour, edge and intensity cues are computed for these pixels. On one hand, pixels whose RGB values are within the dynamic range of the sensor are used to build BCM and BEM. On the other hand, pixels whose value is beyond the sensor dynamic range are used to build BIM. Those pixels considered in motion are not used to build any background model and will be evaluated using the TD algorithm.

Background Colour Model (BCM)

The BCM is computed according to the chromatic-invariant cone representation shown in Fig. 4: first, the RGB mean $\boldsymbol{\mu}_a = \left(\mu^R_a, \mu^G_a, \mu^B_a\right)$ and

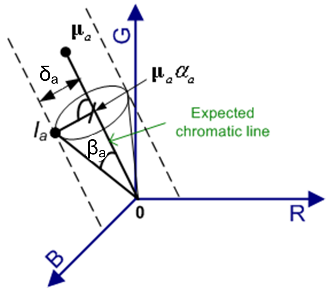


Figure 4: Colour-model representation. $\boldsymbol{\mu}_a$ represents the expected RGB colour value for a pixel a, while $\mathbf{I}_a$ is the current pixel value. The line $\overline{\mathbf{0}\boldsymbol{\mu}_a}$ shows the expected chromatic line —all colours along this line have the same chrominance, but different brightness. α_a and β_a give the current brightness and chrominance angle distortion, respectively.

the standard deviation $\boldsymbol{\sigma}_a = \left(\sigma_a^R, \sigma_a^G, \sigma_a^B\right)$ of each image pixel a during the training period $t = [1 : T_1]$ is computed.

Once each RGB component is normalised by their respective standard deviation σ_a^c, two distortion measures are established during the training period: the brightness $\alpha_{a,t}$ and the chrominance angle $\beta_{a,t}$ distortions. Brightness distortion is computed by minimising the distance between the current pixel value $\mathbf{I}_{a,t}$ and the chromatic line $\overline{\mathbf{0}\boldsymbol{\mu}_a}$. The angle between $\overline{\mathbf{0}\boldsymbol{\mu}_a}$ and $\overline{\mathbf{0}\boldsymbol{I}_a}$ is, in fact, the chromatic angle distortion. Thus, brightness and chromatic angle distortions are given by:

$$\alpha_{a,t} = \frac{\frac{I_{a,t}^R \mu_a^R}{(\sigma_a^R)^2} + \frac{I_{a,t}^G \mu_a^G}{(\sigma_a^G)^2} + \frac{I_{a,t}^B \mu_a^B}{(\sigma_a^B)^2}}{\left(\frac{\mu_a^R}{\sigma_a^R}\right)^2 + \left(\frac{\mu_a^G}{\sigma_a^G}\right)^2 + \left(\frac{\mu_a^B}{\sigma_a^B}\right)^2}, \tag{1}$$

$$\beta_{a,t} = \arcsin \frac{\sqrt{\sum_{c=R,G,B} \left(\frac{I_{a,t}^c - \alpha_{a,t}\mu_a^c}{\sigma_a^c}\right)^2}}{\sqrt{\sum_{c=R,G,B} \left(\frac{I_{a,t}^c}{\sigma_a^c}\right)^2}}. \tag{2}$$

Finally, the Root Mean Square over time of both distortions for each pixel is computed as:

$$\begin{aligned}
\bar{\alpha}_a &= RMS\left(\alpha_{a,t}-1\right) = \sqrt{\frac{1}{T_1}\sum_{t=0}^{T_1}\left(\alpha_{a,t}-1\right)^2}, &(3)\\
\bar{\beta}_a &= RMS\left(\beta_{a,t}\right) = \sqrt{\frac{1}{T_1}\sum_{t=0}^{T_1}\left(\beta_{a,t}\right)^2}, &(4)
\end{aligned}$$

where 1 is subtracted to $\alpha_{a,t}$ so that the brightness distortion becomes zero-mean: positive values identify brighter pixels, whereas negative values correspond to darker pixels. These values are used as normalising factors so that a single threshold can be set for the whole image. This 4-tuple $BCM =< \boldsymbol{\mu}_a, \boldsymbol{\sigma}_a, \bar{\alpha}_a, \bar{\beta}_a >$ constitute the finl pixel colour background model.

Background Edge Model (BEM)

The BEM is built as follows: first the Sobel edge operator is applied to each colour channel in horizontal and vertical directions. This yields toboth horizontal $G^c_{x,a,t} = S_x * I^c_{a,t}$ and vertical $G^c_{y,a,t} = S_y * I^c_{a,t}$ gradient image for each frame during the training period $t = [1 : T]$, where $c \in \{R, G, B\}$ denotes the colour channel.

Then, each background pixel gradient is represented using the gradient mean $\mu_{Gx,a} = (\mu^R_{Gx,a}, \mu^G_{Gx,a}, \mu^B_{Gx,a})$ and $\mu_{Gy,a} = (\mu^R_{Gy,a}, \mu^G_{Gy,a}, \mu^B_{Gy,a})$, and gradient standard deviation $\sigma_{Gx,a} = (\sigma^R_{Gx,a}, \sigma^G_{Gx,a}, \sigma^B_{Gx,a})$ and $\sigma_{Gy,a} = (\sigma^R_{Gy,a}, \sigma^G_{Gy,a}, \sigma^B_{Gy,a})$ computed from all the training frames for each channel.

Then, the magnitudes of the gradient mean μ_G and standard deviation σ_G are computed to build BEM. The orientation of the gradient (μ_θ and σ_θ) is also computed to avoid the false edges created by illumination changes.

$$\mu^c_{G,a} = \sqrt{(\mu^c_{Gx,a})^2 + (\mu^c_{Gy,a})^2}; \quad \sigma^c_{G,a} = \sqrt{(\sigma^c_{Gx,a})^2 + (\sigma^c_{Gy,a})^2}, \tag{5}$$

$$\mu^c_{\theta,a} = \arctan\left(\frac{\mu^c_{Gy,a}}{\mu^c_{Gx,a}}\right); \qquad \sigma^c_{\theta,a} = \arctan\left(\frac{\sigma^c_{Gy,a}}{\sigma^c_{Gx,a}}\right), \tag{6}$$

where $c \in \{R, G, B\}$ denotes the colour channel. Thus, $BEM =< \mu^c_{G,a}, \sigma^c_{G,a}, \mu^c_{\theta,a}, \sigma^c_{\theta,a} >$.

Background Intensity Model (BIM)

Finally, the BIM consist of a 2-tuple given by the mean pixel intensity, μ_a^I and its standard deviation σ_a^I. BIM is computed for those motionless pixels which have a representative number of values beyond sensor dynamic range. So, $BIM =< \mu_a^I, \sigma_a^I >$.

4.2. Parameters Analysis

The thresholds employed for the segmentation task are automatically computed for each model based on statistical inference from the experimental results, as shown next.

Background Colour Model (BCM)

Building BCM is completed by the automatic threshold computation described in Horprasert et al. [34]. First, the normalised distortions are calculated for each pixel:

$$\breve{\alpha}_{a,t} = \frac{\alpha_{a,t}}{\bar{\alpha}_a}; \ \breve{\beta}_{a,t} = \frac{\beta_{a,t}}{\bar{\beta}_a}. \tag{7}$$

This process is repeated during a temporal window of T_2 frames to avoid errors due to an insufficient number of samples. Subsequently, the histograms of both accumulated measures $\breve{\alpha}_{a,t}$ and $\breve{\beta}_{a,t}$ are computed by taking into account all pixel distortions during T_2. The parameters involved in this step are: (i) the chrominance angle distortion threshold, τ_β, controls the limit of the chroma change; (ii) A lower, $\tau_{\alpha 1}$, and a higher, $\tau_{\alpha 2}$, brightness thresholds are needed to define the brightness range, and will be used later to detect shadows and highlights; and (iii) a dark, τ_D, and a light, τ_L, thresholds are used for detecting those pixels beyond the sensor dynamic range (τ_D stands for high intensity values and τ_L for those ones with low intensity).

From the experiments of different sequences, a stable detection has been achieved using the following range of values: $\tau_\beta = \kappa_\beta \max\left(\breve{\beta}_a\right)$, $\kappa_\beta = [1, 2]$, $\tau_{\alpha 1}$ is set at 0.99 of the $\breve{\alpha}_a$ accumulated histogram, $\tau_{\alpha 2}$ is set at $(1 - 0.99)$ of the $\breve{\alpha}_a$ accumulated histogram, $\tau_D = 4\tau_{\alpha 1}$, and $\tau_L = 4\tau_{\alpha 2}$.

Background Edge Model (BEM)

The BEM uses three thresholds for edge pixel segmentation. A minimum magnitude gradient threshold (τ_e) is learnt to decide if an edge can be compared using its oriented gradient. An oriented gradient threshold (τ_θ) and a maximum magnitude gradient threshold (τ_G) are learnt for pixel segmentation. The thresholds are computed as $\tau^c_{e,a} = max(3\sigma^c_{G,a}, \epsilon)$, $\tau^c_{\theta,a} = max(1.5\sigma^c_{\theta,a}, \overline{\sigma}^c_\theta)$, and $\tau^c_{G,a} = max(5\sigma^c_{G,a}, \overline{\sigma}^c_G)$, where these weighting factors are selected empirically to achieve a stable detection in all tested sequences, and $\overline{\sigma}^c_\theta$ and $\overline{\sigma}^c_G$ are the average standard deviation computed over the entire image.

Background Intensity Model (BIM)

The threshold used for pixel segmentation according to BIM is computed as $\tau^I_a = \max\left(7\sigma^I_a, \tau_m\right)$, where τ_m is the lower bound of the sensor dynamic range. Based on the experiments, the sensor dynamic range is determined by τ_m, and τ_n, which are set to25 and 250, respectively.

Temporal Differencing (TD)

Finally, the threshold for temporal differencing segmentation is automatically computed as follows: the histogram of accumulated measures ($h\sigma$) is computed by taking into account the standard deviation of each pixel vaue during the first three frames. The threshold is finally expressed as $\tau_T = 2\max\left([0.98h\sigma], \overline{\sigma}_T\right)$ where $[0.98h\sigma]$ represents the maximum possible value for avoiding outliers, and $\overline{\sigma}_T$ is the averaged standard deviation computed over the entire image.

4.3. Image Segmentation

The segmentation task is done in two steps. The first step obtains the foreground regions for each backgroud model, and the second step combines the segmenttion results to cope withthe camouflage in chroma. A sketch of the Image-Segmentation Module is shown in Fig 5.

Thus, in the first step four general cases are considered, and a different model is applied in each one:

- BCM and BEM are applied to those pixels whose current values are within the sensor dynamic range, and for which BCM and BEM can be built;

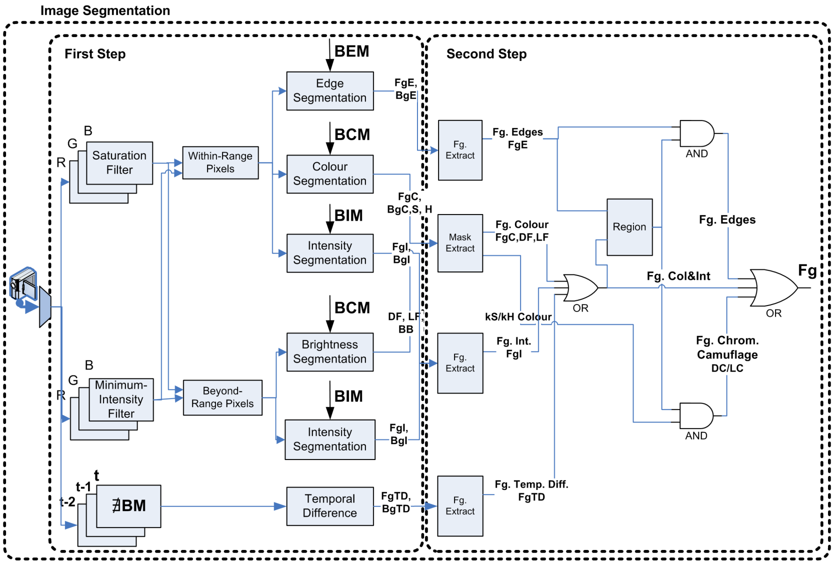


Figure 5: Image segmentation approach. As a result of applying background models to the current frame, pixels are classified in the first step according to the sensor dynamic range using BCM as foreground (FgC), background (BgC), shadow (S), and highlight (H); according to the BEM as foreground (FgE) and background (BgE); using the BCM on pixels beyond the sensor dynamic range, as dark foreground (DF), light foreground (LF), and background (BB); according to the BIM as foreground (FgI) and background (BgI); and according to the TD as foreground (FgTD) and background (BgTD). In a second step, pixels inside the region enclosed by the foregrounds from the first step are combined with a thresholded S and H mask in order to segment the foregrounds dark (DC) and light (LC) camouflage.

- the brightness component of BCM is applied to segment those pixels whose current values are beyond this range;
- BIM is applied to those pixels which do not have enough values within the dynamic sensor range during the modelling process.
- and, TD is applied to those pixels whose background was not visible during the training period and there is no background model available.

As a result, a segmentation map $\mathbf{M}_{a,t}$ is computed at each time. Thus, pixels under the first condition are classified using BCM as background (BgC), highlight (H), shadow (S), or foreground (FgC); and using BEM as background (BgE), foreground (FgE). Those pixels under the second condition are classified as background (BB), or dark foreground (DF) and light foreground (LF); those under the third one as background (BgI) or foreground (FgI); and those under the last one as background (BgTD) or foreground (FgTD). The process is summarized in Algorithm 1.

Edge segmentation is achieved based on the following premises:

- Illumination changes canmodify gradient magnitude but not gradient orientation.
- Gradient orientation is not feasible where there are no edges.
- An edge can appear in any place where there were no edges before.

Assuming the first two premises, the oriented gradients can be compared instead of the gradient magnitudes for those pixels which have a minimum magnitude, in order to avoid false edges due to illumination changes:

$$F_\theta = \big((\tau^c_{e,a} < V^c_{G,a,t}) \wedge (\tau^c_{e,a} < \mu^c_{G,a})\big) \wedge (\tau^c_{\theta,a} < |V^c_{\theta,a,t} - \mu^c_{\theta,a}|), \tag{8}$$

For those pixels satisfying the third premise, their gradient magnitudes are compared instead of their orientation magnitudes:

$$F_G = \big(\neg\big((\tau^c_{e,a} < V^c_{G,a,t}) \wedge (\tau^c_{e,a} < \mu^c_{G,a})\big)\big) \wedge (\tau^c_{G,a} < |V^c_{G,a,t} - \mu^c_{G,a}|), \tag{9}$$

where the $V^c_{\theta,a,t}$ and $V^c_{G,a,t}$ are the gradient orientation and magnitude for every pixel in the current image, respectively.

4.4. Camouflage in Chroma (case DC/LC)

Despite of the act that edge segmentation is less sensitive to global illumination changes than colour and intensity cue, problems like noise, foreground aperture and camouflage prevent of an accurate segmentation of the foreground objects. Therefore, handling dark and light camouflage problems by using only edges is not feasible. In these cases, the brightness component of the colour model should be used to solve the foreground aperture anomaly by filling the foreground object.

Thus, in a second step, the region enclosed by the foreground pixels segmented in the first step are combined with the thresholded shadows and highlights in order to solve the foreground camouflage in chroma while avoiding the global and local illumination problems, thereby segmenting foreground pixels as dark (DC) and light camouflage (LC).

Algorithm 1 Image Segmentation.

```
• if BCM and BEM exist for the current pixel ('a'), then:
    – if the current pixel ($I^c_{a,t}$) is within the sensor dynamic range ($\tau_m < I^c_{a,t} < \tau_n$), then:
        ∗ if it has a different chrominance ($\breve{\beta}_{a,t} > \tau_\beta$) or different gradient ($F_\theta \vee F_G$), then foreground (FgC, FgE),
        ∗ else if it has lower brightness ($\breve{\alpha}_{a,t} < \tau_{\alpha 1}$) and is outside the enclosed foreground region ($\notin Region(Fg)$), then shadow (S),
        ∗ else if it has lower brightness and is inside the enclosed foreground region ($\in Region(Fg)$), then dark camouflage (DC),
        ∗ else if it has higher brightness ($\breve{\alpha}_{a,t} > \tau_{\alpha 2}$) and is outside the enclosed foreground region ($\notin Region(Fg)$), then highlight (H),
        ∗ else if it has higher brightness and is inside the enclosed foreground region ($\in Region(Fg)$), then light camouflage (LC),
        ∗ otherwise, original background (BgC, BgE).
    – else
        ∗ if it has lower brightness ($\breve{\alpha}_{a,t} < \tau_D$), then dark foreground (DF),
        ∗ else if it has higher brightness ($\breve{\alpha}_{a,t} > \tau_L$), then light foreground (LF),
        ∗ otherwise, original background (BB).
• else if BIM exists, then:
    – if it has lower or higher intensity ($\left|I^I_{a,t} - \mu^I\right| > \tau^I_a$), then foreground (FgI),
    – otherwise, original background (BgI).
• otherwise, no background was visible during the training period and temporal-differencing algorithm is applied
    – if it has different intensity over three frames ($\sigma_{a,t} > \tau_T$), then foreground (FgTD),
    – otherwise, original background (BgTD).
```

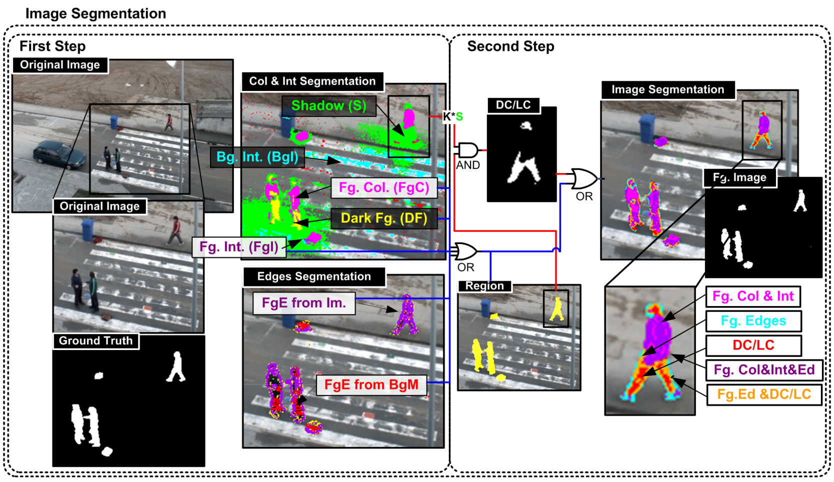


Figure 6: Colour and intensity segmentation, plus edge segmentation figures showing the foreground pixels and S/H masks. DC/LC figure shows how dark and light camouflage pixels are correctly segmented using a thresholded S/H combined with the foreground region.

In this second step, shadows (S) and highlights (H) are also detected due to DC/LC. Furthermore, to avoid noise generated from the edge cues, the foreground edges obtained for the BEM are filtering using the region created to cope with DC/LC problem.

An example of image segmentation where the camouflage in chroma is detected can be seen in Fig. 6, where the agent near the crosswalk appears with his jeans dark camouflaged with the road. The whole process is summarised in Algorithm 1.

4.5. Ghost Detection

Segmented regions are evaluated to assess whether they contain a ghost or a foreground region based on two premises:

- A ghost corresponds to an object which was included in the background model.
- A ghost cannot be in motion. Therefore, the detected region does not exhibit any motion.

Firstly, the boundary and the area of the detected region are compared with the foreground edges and the region enclosed by these edges. Thus,

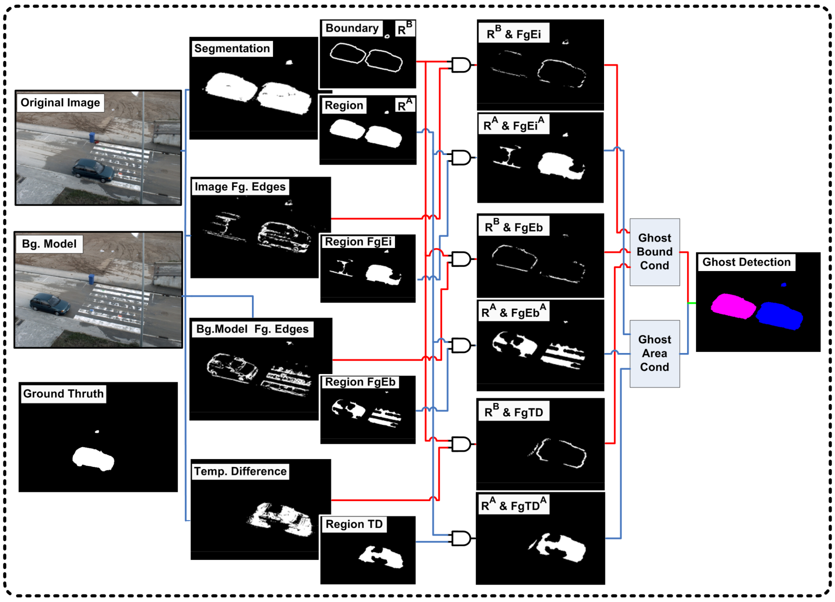


Figure 7: The boundary and area of the detected region are compared with the foreground edges and the region enclosed by these edges. This way the probability that a detected region belongs to the background model or to the current image is computed. Then, the boundary and the area of the detected region are also compared with the foreground agens obtained from the temporal difference algorithm. Thus, the probability that the detected region is in motion canbe obtained. Then, a region is considered a ghost based on the probabilities obtained in the first step. Ghost is shown in magenta colour.

foreground segmentation is compared with the foreground edges obtained from the edge cue to infer the probability that a detected region belongs to the background model or to the current image. Then, the boundary and the area of the detected region are also compared with the foreground obtained from TD to infer the probability that the detected region is in motion. Finally, a region is considered as a ghost based on this probability.

A sketch of the ghost detection approach can be seen in Fig. 7, where the images show how the ghost detection works in a real sequence, images are from the Hermes_Outdoor_Cam1 sequence.

5. Experimental Results

Our approach has been tested in several indoor and outdoor sequences under uncontrolled environments, where multiple segmentation problems ap-

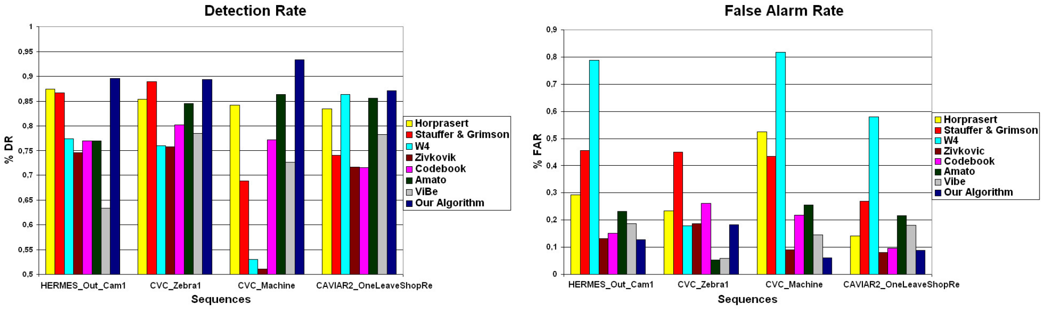


Figure 8: Detection rate and false alarm rate results. First sequence is from HERMES Database, second and third sequences are from CVC Database, and fourth sequence is from CAVIAR Database. Our approach has been compared with different approaches [15, 17, 34, 20, 35, 50, 25] using a manually segmented ground-truth. Our algorithm obtains the best DR while maintaining the lowest FAR in all the sequences evaluated.

pear. Most of the sequences are taken from well-known public databases. Successful segmentation results have been achieved for all of these sequences.

In order to evaluate the performance of the proposed approach in a quantitative way, ground-truth masks have been manually generated. The sequences segmented are *Hermes_Outdoor_Cam1* from the HERMES database[1] (1612 frames @15 fps, 1392 x 1040 PX), *CVC_Zebra1* sequence from CVC database[2] (1343 frames @20 fps, 720 x 576 PX), *CVC_Machine* sequence from CVC database (797 frames @29 fps, 640 x 480 PX), *OneLeaveShopReenter1cor* from the CAVIAR database[3] (389 frames @ 25 fps, 384 x 288 PX) used in PETS 2004, and Hall_Monitor from the NEMESIS database[4] (300 frames, 352x240 PX). Furthermore, approaches from other authors [15, 17, 34, 35, 20, 52, 53, 50, 25] have been used for performance comparison.

Two standard metrics were considered to evaluate quantitatively the performance of our proposed method. Detection Rate (DR) (also called True Positive Rate) and False Alarm Rate (FAR) [33, 12]:

$$DR = \frac{TP}{TP + FN}; \; FAR = \frac{FP}{TP + FP}, \tag{10}$$

[1] http://www.hermes-project.eu

[2] http://iselab.cvc.uab.es

[3] http://homepages.inf.ed.ac.uk/rbf/CAVIAR

[4] http://www.ics.forth.gr/cvrl/demos/NEMESIS/hall_monitor.mpg

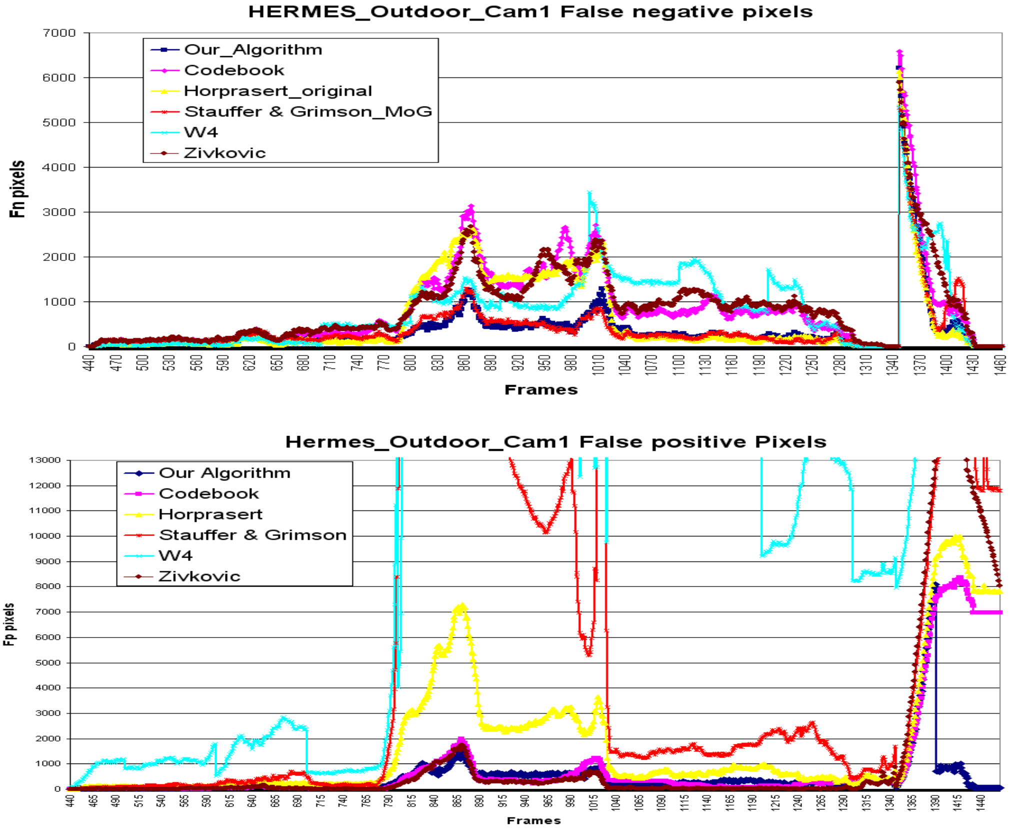


Figure 9: False negatives (top) and false positives (bottom) computed by comparing [15] (in cyan), [17] (in red), [34] (in yellow), [35] (in magenta), [20] using the shadow detector of [33] (in brown) and our approach (in blue) for the HERMES database. Our approach obtains the best results. See text for details.

where DR is the ratio of correctly detected pixels to the ground truth data, and FAR is the ratio of misclassified pixels to the total number of detected pixels. TP, FP and FN correspond to the true positive, false positive, and false negative pixels, respectively, when comparing the segmentation results with the ground truth data.

Fig. 8 shows the results of the segmentation process using DR and FAR. Results show that our algorithm obtains the best DR with the lowest FAR in all the evaluated sequences. Figs. 9, 10 and 11 show how our approach obtains the best results.

Fig. 9 shows the results to compare our approach with other approaches

[15, 17, 34, 35, 20] on the *Hermes_Outdoor_Cam1* sequence. Top graph of Fig. 9 shows the number of false negative pixels segmented using the different approaches, and the bottom one shows the number of false positives. Frames 790 to 1040 correspond to a gradual illumination change. Also, two cars appear into the scene and several persons are crossing the road through a crosswalk. Therefore, multiple motion segmentation difficulties appears in this sequence: (i) global illumination changes —the scene get darker during an instant—, (ii) local illumination changes —shadows due to agents and vehicles—, (iii) camouflage —trousers of an agent when appearing in the scene—, (iv) dark and light camouflage —dark camouflage of the trousers of an when crossing the crosswalk and light camouflage of a white car with the grey road—, and (v) ghost problem —a parked car suddenlybegins to move—.

In the aforementioned sequence, W^4 (cyan line) segments the illumination change as foreground, and also any shadows of cars and agents. The Stauffer and Grimson algorithm (red line) cannot always cope with the illumination change and also classifies the shadows as foreground. The Horprasert et al. approach (yellow line) cannot tackle the light camouflage (white car with grey road) and the Codebook technique (magenta line) is not able to differentiate between the illumination change and camouflage in chroma. The Zivkovic et al. approach (brown line) segments the illumination changes and all the shadows like the Stauffer and Grimson approach but, since their technique includes a shadow detection module [33], the system has also problems when distinguishing illumination changes and camouflage in chroma. The results show that our approach is robust to these problems and obtains the best segmentation performance. An exemplar frame (number 864) is shown in Fig. 10.(a), where light camouflage (a white car over a grey road), and soft illumination changes are present.

Frames 1340 to 1460 in Fig. 9 correspond to a car parked which begins to move (ghost). Our approach is the only one among all the evaluated approaches that detects the ghost problem as soon as it occurs without need of a background updating. This fact can be observed in the false positive graph of Fig. 9. An exemplar frame (number 1411) showing the ghost problem can be seen in Fig. 10.(b).

Fig. 10 shows significant frames comparing our approach with state-of-the-art approaches. First row shows the original image, second row is the ground truth, and from third to seventh the segmentation results are shown for [15], [17], [34], [35] and [20], respectively. Last row shows the results

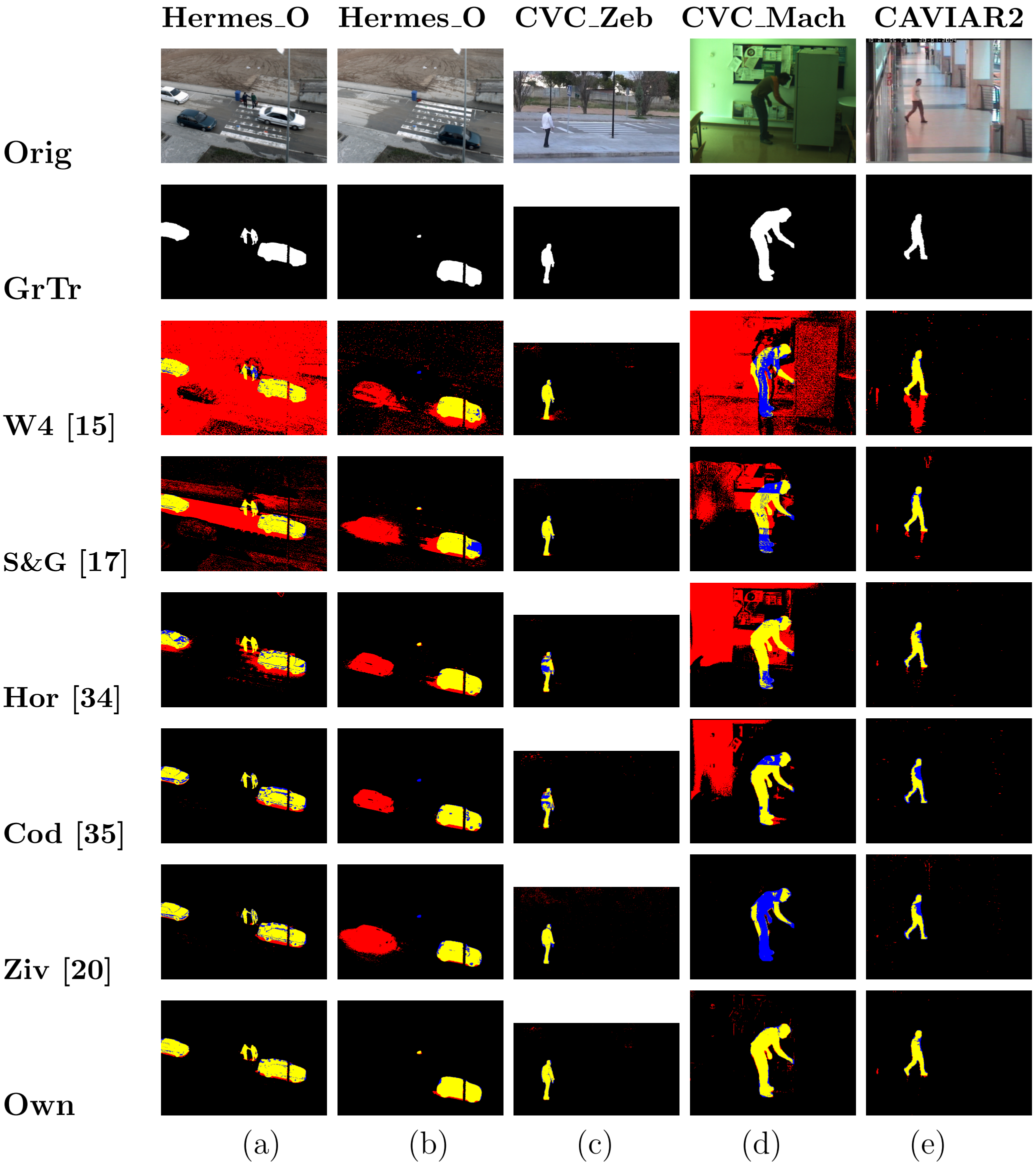


Figure 10: Foreground segmentation comparison using a hand-segmented ground truth. First and second column are from HERMES Database, third and fourth columns are from CVC Database, and fifth column is from CAVIAR Database. First row is the original image, second row is the ground truth, third row is the segmentation results from W^4 approach [15], fourth row from Stauffer and Grimson [17], fifth row from Horprasert et al. [34], sixth row from Codebook approach [35], seventh row from Zivkovic et al. [20] using a shadow detector [33], and eighth row from own approach. Segmentation results are coloured in yellow for TP pixels, blue for FN pixels, and red for FP pixels. Our algorithm obtains more number of TP along with less number of FP and FN.

using our approach. In this figure it can be seen that our framework obtains more TP and less FP and FN because its ability of tackling global and local illumination problems, the problems beyond the dynamic range, the chroma and intensity camouflage problem, and both bootstrapping and ghosts.

In the *CVC_Zebra1* sequence, vehicles and people appear, some agents walking beside street lamps and trees. W^4 segments the shadows as foreground and the updating process fails. The Stauffer and Grimson algorithm can not cope with shadows and gradual illumination changes. The technique of Horprasert et al. can not address light camouflage detection (white shirt with light-grey road) and suffers the saturation case of the sky with gradual illumination changes. Codebook cannot also cope with the light camouflage problem and saturation simultaneously. The Zivkovic et al. approach fails in the cases of illumination changes, camouflages and *sleeping persons* [4] (i.e. foreground pixels are segmented as background because the updating system wrongly incorporates foreground motionless objects to the background model). Our algorithm is robust to all of these problems. Fig. 10.(c) presents one frame where the light camouflage problem as described above is observed (white shirt with grey road).

In the *CVC_Machine* sequence an agent enters into the scene and interacts with a vending machine, see Fig. 10.(d). This scene presents strong illumination changes, and there is a saturation effect in the wall. Dark and light camouflages are also present in this scene (agent in front of the wall). Our algorithm can satisfactorily manage strong illumination changes, saturation problems, and dark and light camouflage while avoiding the sleeping persons effect. Zivkovic et al. can manage the strong illumination change using the updating system, but it fails when dealing with the sleeping persons anomaly.

In the sequence *OneLeaveShopReenter1cor*, the two agents are correctly segmented, see Fig. 10.(e). The colour distribution of the background is very similar to that of the agents, thus including strong clutter. Furthermore, several oriented lighting sources with different illuminants are present: these lights dramatically affect an agent appearance depending on its position and orientation (bluish effect at the right of the corridor, and reddish at the left). A significant frame of this sequence can be seen in the Fig. 10.(e), where dark camouflage and shadows are correctly corrected using our approach.

Fig. 11 shows frames from Hall_Monitor sequence (top row) comparing Wang et al. approach [52] (second row), Huang et al. approach [53] (third row) and our approach (bottom row). The sequence exhibits challenging

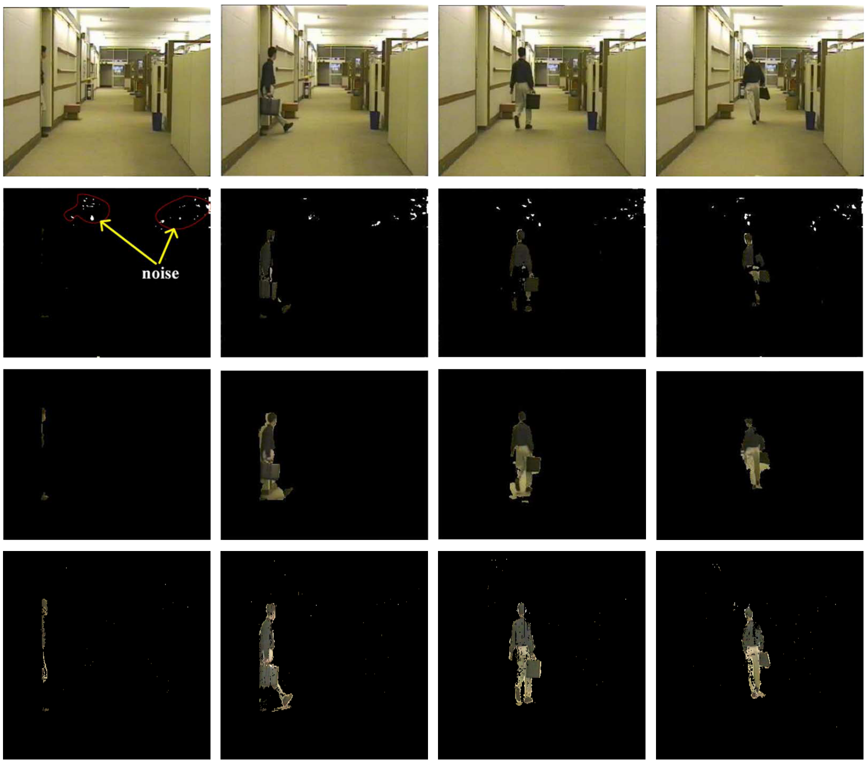


Figure 11: First row shows the original frames from the Hall_Monitor sequence of the NEMESIS dataset, second row shows the detection results of Wang et al. [52], and third row shows the detection results from Huang et al. [53]. These images have been obtained directly from [53]. Fourth row shows the segmentation results of our proposed approach without applying any morphological operation.

aspects due to noise, shadows, and camouflage. Wang et al. can not correctly handle noise and camouflage. Huang et al. is able to manage problems with noise, but shadows can not be properly removed. Also, by using their approach, regions corresponding to background are segmented as foreground, such as the region around the legs. Instead, our approach can cope with all the aforementioned issues.

Fig. 12 shows selected frames with the results of our approach in differ-

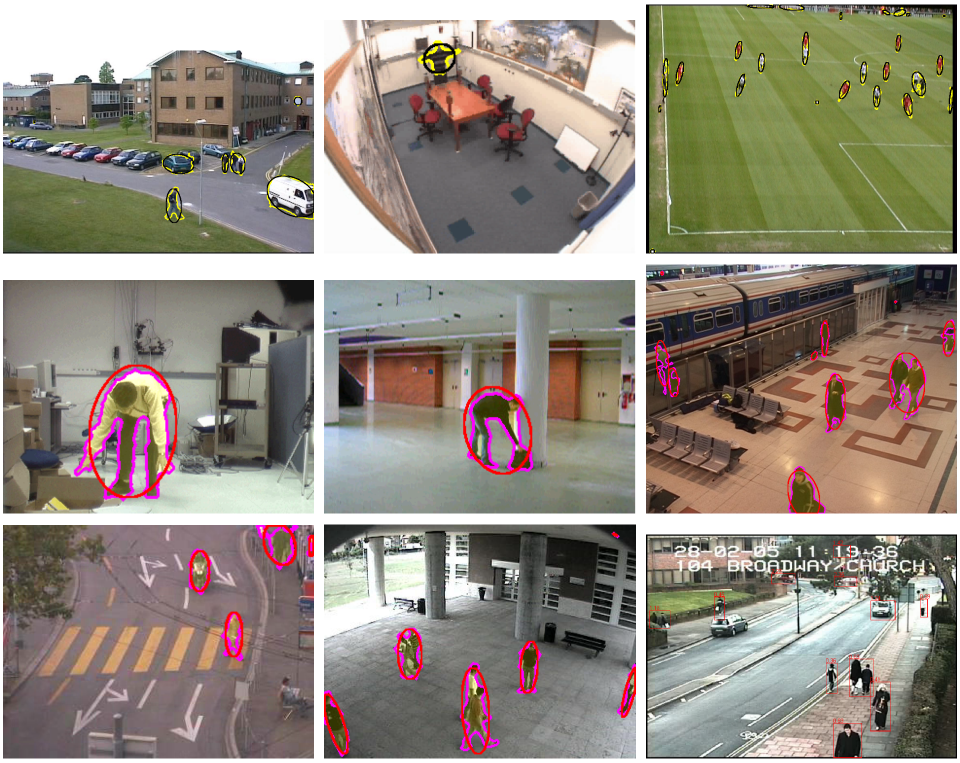


Figure 12: Foreground detection results applying our approach to different datasets. Images from left to right: PETS2001, ATON, VS-PETS, UMIACS, MODLAB, PETS2006, ETHZ, VSSN06, and CLEAR06 databases. All the agents and vehicles are correctly detected showing robustness to global and local illumination problems, problems beyond the dynamic range, and problems with camouflage in chroma.

ent datasets: PETS[5], ATON[6], VS-PETS[7], UMIACS[8], MODLAB [9], PETS 2006[10], ETHZ[11], VSSN06[12], and CLEAR06[13] database.

[5] ftp://ftp.pets.rdg.ac.uk/pub/PETS2001

[6] http://cvrr.ucsd.edu/aton/

[7] ftp://ftp.pets.rdg.ac.uk/pub/VS-PETS/

[8] http://www.umiacs.umd.edu/users/

[9] http://www.na.icar.cnr.it/ maddalena.l/MODLab/MODseq.html

[10] http://pets2006.net/

[11] http://www.vision.ee.ethz.ch/datasets/

[12] http://imagelab.ing.unimore.it/vssn06/

[13] http://clear-evaluation.org/clear06/

Lastly, some remarks on real-time requirements are next discussed. Significant speed improvements of our presented technique can be achieved because of the pixel-based nature of the approach, making the algorithm parallelizable. Since the current system is implemented in *Matlab* without a careful code optimisation, subsequent implementations of bottleneck modules in C++ should yield speed improvements over 10-100 times the computational time of specific, most time-consuming functions. This would allow the system to process the previously described sequences in near real time.

6. Conclusions

The approach described in this paper combines colour, intensity and edge cues, and a temporal differencing technique in a collaborative architecture, in which each module is devoted to a specific task. The proposed framework is built upon a case analysis of motion segmentation problems associated with the use of different cues. Consequently, this theoretical study has allowed us to define when each cue can be used to address different segmentation failures.

The background models built for each cue have been improved with respect to the current state of the art. A novel chromatic invariant cone model is proposed. Also, combining invariant gradient orientation with its magnitude allows our system to detect false edges due to intense global illumination changes.

The proposed hybrid approach can cope with different colour problems as dark and light foreground. Furthermore, we solve problems with the dynamic range (in cases of saturation and lack of colour) using intensity cues. Our approach also tackles camouflage in intensity and chroma, together with global and local (shadows and highlights) illumination changes. In addition, problems like bootstrapping and ghosts are handled.

Experiments on complex indoor and outdoor scenarios have yielded robust and accurate results, thereby demonstrating the ability of the system to deal with unconstrained and dynamic scenes.

For future work, a proper updating process should be included in the approach to incorporate motionless objects to the background model. Furthermore, the use of a pixel-updating process can help to reduce false positives obtained by using the intensity mask due to drastic illumination changes. Furthermore, colour invariant normalisation or colour constancy techniques can be used to improve the colour model. The edge model can be enhanced by avoiding false edges due to local intense illumination changes. Further, edge linking or B-spline techniques can be used to avoid the partial loss of foreground borders due to camouflage, thereby improving the edge mask. A statistical-based decision approach that combines the use of multiples cues could be useful to reduce some of the parameters described in this article. Lastly, the discrimination between the agents and the local environments can be enhanced by using new cues (such as texture) or tracking.

Acknowledgments

This work has been supported by the Spanish Research Programs Consolider-Ingenio 2010:MIPRCV (CSD200700018); Avanza I+D ViCoMo (TSI-020400-2009-133); along with the Spanish projects TIN2009-14501-C02-01, TIN2009-14501-C02-02, and DIP2010-17112.

References

[1] D. Gavrila, The visual analysis of human movement: A survey, CVIU 73 (1) (1999) 82–98.

[2] J. Gonzàlez, D. Rowe, J. Varona, F. Roca, Understanding dynamic scenes based on human sequence evaluation, Image and Vision Computing 27 (10) (2009) 1433–1444.

[3] J. Varona, J. Gonzàlez, I. Rius, J. Villanueva, On importance of detection for video surveillance applications, Optical Engineering 47 (8) (2008) 1–8.

[4] K. Toyama, J.Krumm, B.Brumitt, B.Meyers, Wallflower: Principles and practice of background maintenance, in: Proc. ICCV'99, Vol. 1, Kerkyra, Greece, 1999, pp. 255–261.

[5] L. Wang, W. Hu, T. Tan, Recent developments in human motion analysis, Pattern Recognition 36 (3) (2003) 585–601.

[6] T. B. Moeslund, A. Hilton, V. Kruger, A survey of advances in vision-based human motion capture and analysis, CVIU 104 (2006) 90–126.

[7] R. Radke, S.Andra, O. Al-Kofahi, B.Roysam, Image change detection algorithms: a systematic survey, IEEE TIP 14 (3) (2005) 294–307.

[8] I. Huerta, D. Rowe, M. Mozerov, J. Gonzàlez, Improving background subtraction based on a casuistry of colour-motion segmentation problems, in: Ibpria'07, Vol. 2, Springer LNCS, Girona, Spain, 2007, pp. 475–482.

[9] I. Huerta, A. Amato, J. Gonzàlez, J. Villanueva, Fusing edge cues to handle colour problems in image segmentation., in: Proc. AMDO'08, Vol. 5098, Springer LNCS, Andratx, Mallorca, Spain, 2008, pp. 279–288.

[10] A. McIvor, Background subtraction techniques, in: In Proc. of Image and Vision Computing, Auckland, New Zealand, 2000.

[11] M. Piccardi, Background subtraction techniques: a review, in: IEEE International Conference on Systems, Man and Cybernetics, Vol. 4, The Hague, Netherlands, 2004, pp. 3099 – 3104.

[12] M. Karaman, L. Goldmann, D. Yu, T. Sikora, Comparison of static background segmentation methods, in: VCIP '05, Beijing, China, 2005.

[13] S. Brutzer, B. Hferlin, G. Heidemann, Evaluation of background subtraction techniques for video surveillance, in: IEEE CVPR'11, 2011, pp. 1937–1944.

[14] J. Heikkila, O. Silven, A real-time system for monitoring of cyclists and pedestrians, in: Proceedings of the Second IEEE Workshop on Visual Surveillance, IEEE Computer Society, Washington, DC, USA, 1999, pp. 74–81.

[15] I. Haritaoglu, D. Harwood, L. Davis, W4: Real-time surveillance of people and their activities, IEEE TPAMI 22 (8) (2000) 809–830.

[16] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland, Pfinder: Real-time tracking of the human body, IEEE TPAMI 19 (7) (1997) 780–785.

[17] C. Stauffer, W. Grimson, Adaptive background mixture models for real-time tracking, in: IEEE CVPR'99, Vol. 1, Ft. Collins, CO, USA, 1999, pp. 22–29.

[18] C. Stauffer, W. Eric, L. Grimson, Learning patterns of activity using real-time tracking, IEEE TPAMI 22 (8) (2000) 747–757.

[19] Z. Zivkovic, Improved adaptive gaussian mixture model for background subtraction, in: Proc. ICPR'04, Vol. 2, 2004, pp. 23–26.

[20] Z. Zivkovic, F. Heijden, Efficient adaptive density estimation per image pixel for the task of background subtraction, Pattern Recognition Letters 27 (7) (2006) 773–780.

[21] A. Elgammal, D. Harwood, L. S. Davis, Nonparametric background model for background subtraction, in: ECCV'00, Dublin, 2000, pp. 751–767.

[22] A. Mittal, N. Paragios, Motion-based background subtraction using adaptive kernel density estimation, in: Proc. CVPR'04, Vol. 2, Washington DC, USA, 2004, pp. 302–309.

[23] Y. Chen, C. Chen, C. Huang, Y. Hung, Efficient hierarchical method for background subtraction, Pattern Recognition 40 (10) (2007) 2706–2715.

[24] L. Cheng, M. Gong, D. Schuurmans, T. Caelli, Real-time discriminative background subtraction, IEEE TIP 20 (5) (2011) 1401–1414.

[25] O. Barnich, M. V. Droogenbroeck, Vibe: A universal background subtraction algorithm for video sequences, IEEE TIP 20 (6) (2011) 1709–1724.

[26] A. Colombari, A. Fusiello, V. Murino, Patch-based background initialization in heavily cluttered video, IEEE TIP 19 (4) (2010) 926–933.

[27] L. Li, W. Huang, I. Y.-H. Gu, Q. Tian, Statistical modeling of complex backgrounds for foreground object detection, IEEE TIP 13 (11) (2004) 1459–1472.

[28] Y. Sheikh, M. Shah, Bayesian modeling of dynamic scenes for object detection, IEEE TPAMI 27 (11) (2005) 1778–1792.

[29] K. A. Patwardhan, G. Sapiro, V. Morellas, Robust foreground detection in video using pixel layers, IEEE TPAMI 30 (4) (2008) 746–751.

[30] L. Maddalena, A. Petrosino, A self-organizing approach to background subtraction for visual surveillance applications, IEEE TIP 17 (7) (2008) 1168–1177.

[31] E. Lopez-Rubio, R. Luque-Baena, E. Dominguez, Foreground detection in video sequences with probabilistic self-organizing maps, International Journal of Neural Systems 21 (3) (2011) 225–246.

[32] V. Mahadevan, N. Vasconcelos, Spatiotemporal saliency in dynamic scenes, IEEE TPAMI 32 (1) (2010) 171–177.

[33] A. Prati, I. Mikic, M. Trivedi, R. Cucchiara, Detecting moving shadows: Algorithms and evaluation, IEEE TPAMI 25 (7) (2003) 918–923.

[34] T. Horprasert, D.Harwood, L.S.Davis, A statistical approach for real-time robust background subtraction and shadow detection, in: IEEE Frame-Rate Applications Workshop, Kerkyra, Greece, 1999.

[35] K. Kim, T. Chalidabhongse, D. Harwood, L. Davis, Real-time foreground-background segmentation using codebook model, Real-Time Imaging 11 (3) (2005) 172–185.

[36] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, S. Sirotti, Improving shadow suppression in moving object detection with hsv color information, in: Proceedings. IEEE Intelligent Transportation Systems, Oakland, USA, 2001, pp. 334–339.

[37] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, Detecting moving objects, ghosts, and shadows in video streams, IEEE TPAMI 25 (10) (2003) 1337–1342.

[38] G. Finlayson, S. Hordley, C. Lu, M. Drew, On the removal of shadows from images, IEEE TPAMI 28 (1) (2006) 59–68.

[39] S. Nadimi, B. Bhanu, Physical models for moving shadow and object detection in video, IEEE TPAMI 26 (8) (2004) 1079–1087.

[40] Y. Weiss, Deriving intrinsic images from image sequences, in: Proc. ICCV'01, Vol. 02, Vancouver, Canada, 2001, pp. 68–75.

[41] N. Martel-Brisson, A. Zaccarin, Learning and removing cast shadows through a multidistribution approach, IEEE TPAMI 29 (7) (2007) 1133–1146.

[42] N. Martel-Brisson, A. Zaccarin, Kernel-based learning of cast shadows from a physical model of light sources and surfaces for low-level segmentation, in: IEEE CVPR'08, 2008, pp. 1–8.

[43] J. Huang, C. Chen, Moving cast shadow detection using physics-based features, in: IEEE CVPR'09, 2009, pp. 2310–2317.

[44] H. Jabri, Z.Duric, A.Rosenfeld, Detection and location of people in video images using adaptive fusion of color and edge information, in: 15th ICPR, Vol. 4, Barcelona, Spain, 2000, pp. 627–630.

[45] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, H. Wechsler, Tracking groups of people, CVIU 80 (1) (2000) 42–56.

[46] O. Javed, K. Shafique, M. Shah, A hierarchical approach to robust background subtraction using color and gradient information, in: Proc. of the Workshop on Motion and Video Computing (MOTION'02), Orlando, 2002, p. 22.

[47] A. Leone, C. Distante, Shadow detection for moving objects based on texture analysis, Pattern Recognition 40 (4) (2007) 1222–1233.

[48] M. Heikkila, M. Pietikainen, A texture-based method for modeling the background and detecting moving objects, IEEE TPAMI 28 (4) (2006) 657–662.

[49] J. Yao, J. Odobez, Multi-layer background subtraction based on color and texture, in: IEEE CVPR'07, Minneapolis, Minnesota, USA, 2007, pp. 17–22.

[50] A. Amato, M. Mozerov, X. Roca, J. Gonzàlez, Robust real-time background subtraction based on local neighborhood patterns, EURASIP Journal on Advances in Signal Processing (2010) 1–7.

[51] J. Shen, Motion detection in color image sequence and shadow elimination, Visual Communications and Image Processing 5308 (2004) 731–740.

[52] L. Wang, T. Tan, H. Ning, W. Hu, Silhouette analysis-based gait recognition for human identification, IEEE TPAMI 25 (12) (2003) 1505–1518.

[53] S. Huang, L. Fu, P. Hsiao, Region-level motion-based background modeling and subtraction using mrfs, IEEE TIP 16 (5) (2007) 1446–1456.