

Article

Background Subtraction Based on a New Fuzzy Mixture of Gaussians for Moving Object Detection

Ali Darwich ^{1,2}, Pierre-Alexandre Hébert ¹, André Bigand ^{1,*} and Yasser Mohanna ²

¹ Laboratoire d'Informatique Signal et Image de la Côte d'Opale, 62228 Calais, France; darwich-ali@hotmail.com (A.D.); hebert@univ-littoral.fr (P.-A.H.)

² Department of physics and electronics, Lebanese University, 1003 Beirut, Lebanon; yamoha@ul.edu.lb

* Correspondence: bigand@univ-littoral.fr; Tel.: +33-321465653

Received: 15 May 2018; Accepted: 28 June 2018; Published: 10 July 2018



Abstract: Moving foreground detection is a very important step for many applications such as human behavior analysis for visual surveillance, model-based action recognition, road traffic monitoring, etc. Background subtraction is a very popular approach, but it is difficult to apply given that it must overcome many obstacles, such as dynamic background changes, lighting variations, occlusions, and so on. In the presented work, we focus on this problem (foreground/background segmentation), using a type-2 fuzzy modeling to manage the uncertainty of the video process and of the data. The proposed method models the state of each pixel using an imprecise and adjustable Gaussian mixture model, which is exploited by several fuzzy classifiers to ultimately estimate the pixel class for each frame. More precisely, this decision not only takes into account the history of its evolution, but also its spatial neighborhood and its possible displacements in the previous frames. Then we compare the proposed method with other close methods, including methods based on a Gaussian mixture model or on fuzzy sets. This comparison will allow us to assess our method's performance, and to propose some perspectives to this work.

Keywords: background subtraction; gaussian mixture model; Type-2 fuzzy sets; optical flow

1. Introduction

Background extraction, also known as foreground detection, or background subtraction, is a conventional technique in the field of frame processing and computer vision, in which the foreground of a frame (often a set of moving objects) is extracted for on-line or off-line processings (object recognition, etc.). Generally the regions of a frame of interest are objects (humans, cars, texts, etc.).

Many issues prevent such a basic technique from giving good background subtraction results. Indeed, it often fails in real videos: for instance when the frame is spoiled by noise, or when a little change in the environment disturbs the motion detection—typically illumination, shadow, slight movement and so on.

This problem can be roughly solved with very basic algorithms, based on the difference between a background-frame model and the current frame [1,2], but the videos often contain disruptive elements that may defeat such methods. In particular, dynamic backgrounds, or lightning variations in videos, can greatly complicate the classification of a pixel as foreground or background.

To deal with such complex videos (most often natural scenes), background subtraction methods based on more advanced modeling tools have therefore been developed.

Most of them follow a simple flow diagram (cf. Figure 1), structured in four major steps: pre-processing, background modeling, foreground detection and data validation. The produced result consists in a sequence of foreground masks, which tend to estimate the “ground truth” foreground mask of the corresponding video frame (cf. Figure 1, fourth and first image). Detection and tracking of moving

foregrounds can be considered as lower-level vision tasks, intended to prepare the understanding of the high-level event (decision-making).

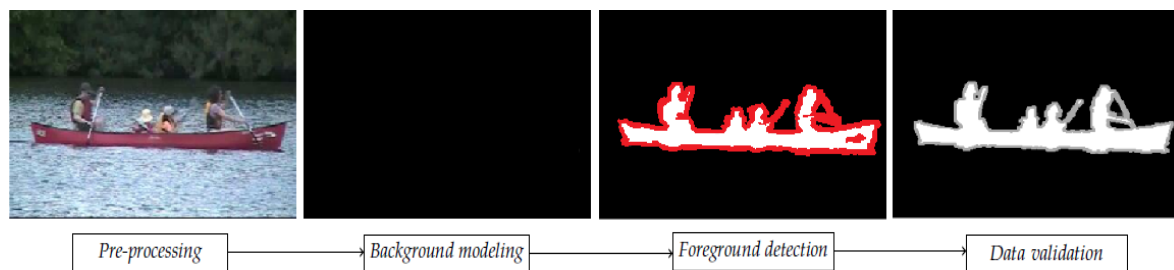


Figure 1. Diagram of a generic background subtraction algorithm illustrated with the video Canoe, from the video category Dynamic background (from the site [3]).

There are many challenges in developing a good background subtraction algorithm. First of all, it must be robust with regard to lighting changes. Then, it must avoid detecting non-stationary background elements such as oscillating leaves, rain, snow and shadows. Moreover, the internal background model must react quickly to changes in background such as starting and stopping vehicles for example. A good background subtraction algorithm must handle moving foregrounds that first blend in the background and then appear in the foreground later.

The problem of identifying moving foregrounds in a complex environment is still far from being completely solved, and new algorithms are regularly added to the literature. Today, the challenge in background subtraction mining is to develop robust, accurate and efficient approaches on complex videos.

According to Bouwmans et al. [4], three main conditions are necessary for the proper operation of background subtraction in video surveillance: the camera must be fixed, the lighting should be constant and the background should be static.

In practice, various disturbances can come to undermine these conditions, and constitute many challenges for the subtraction of patterns:

- Noise in the frame: it is often due to a source of poor quality frames (for example, webcam frames, or heavily compressed).
- Camera Jitter: video can be captured by unstable cameras, for example, because of vibrations or wind. The amplitude of the jitter can vary from one video sequence to another.
- Automatic camera settings: most modern cameras have auto-focus, automatic gain control, auto white balance, and automatic brightness control. These adjustments change the dynamics in the color levels between the different frames in the sequence.
- Lighting changes: they can be relatively progressive, as in an outdoor scene (movement of the sun), or sudden as the lighting of a lamp in an indoor scene. Illumination strongly affects the appearance of the background and causes false detection.
- Bootstrap: when the first frames of a video contain still foregrounds, the initialization of the background model may pose a problem. Bootstrap techniques can then be used.
- Dynamic background: some parts of the scenery may contain motions (fountain, cloud movement, swaying tree branches, water wave, etc.), but should be considered background. Such movements may be periodic or irregular (eg, traffic lights, undulating trees). Managing such a dynamic background is a very difficult task.
- Camouflage: intentionally or not, some foregrounds may be confused with the surrounding background. This is a critical problem in monitoring applications. And it particularly affects temporal differentiation methods.
- Opening foreground: when a displaced foreground object contains uniformly colored regions, some of its pixels may be detected as motionless, and consequently as background.
- Sleeping foreground: a foreground element that becomes immobile, may be incorporated into the background. Depending on the context, this is not necessarily desirable.

- Shadows: shadows can be detected as foreground, and can come either from background elements or moving foregrounds [5].
- Moving background “objects”: some parts of the background may move. In this case these objects should not be classified as foreground.
- Inserted background elements: a new background “object” can be inserted. Depending on the context, these objects should not be considered as part of the foreground.
- Start of moving foreground: when a “foreground” initially confused with the background starts moving, the background parts then revealed are called “ghosts”; they should be quickly assimilated to the background.
- Weather variations: the detection of moving foreground becomes a very difficult task when videos are captured in shower weather conditions, such as in winter, with for example snowstorms, snow on the ground, fog, strong winds (turbulence).

To overcome these obstacles, a lot of research has been devoted to develop new and performing algorithms. The common approach is to perform a background subtraction, which consists in modeling the background scene, so as to detect foreground objects as pixel regions not compatible with the model. This is the approach we focus on.

This paper is organized as follows: in the following section we review the most popular methods to extract backgrounds (often based on Gaussian Mixture Models, GMMs, classic GMMs and fuzzy GMMs). Section 3 introduces Gaussian Mixture Models in background modeling, and its type-2 fuzzy set extension FGMMs. Section 4 proposes a new decision under uncertainty technique using an interval comparison method, we name IV-FGMM (Interval-Valued Fuzzy Gaussian Mixture Model). Section 5 presents a comparison with methods based on GMMs and fuzzy sets FGMMs. Finally Section 6 presents the conclusion and the perspectives of this work.

2. Background

In this section we present some common approaches to background detection, considering the uncertainty added to the process. Indeed the previous listed challenges represent a high level of uncertainty for algorithms, and this is the point of view we consider.

2.1. Common Approaches

The most common techniques use a density function of the pixel values over the video frames, as a by-pixel probabilistic background modeling. Decision is then frequently obtained from a maximum likelihood-like decision method. Many algorithms using Gaussian models have been proposed [6–14]. Most of them use a single Gaussian density function per pixel. However, pixel values that may have more complex distributions and Gaussian mixtures are generally preferred.

Stauffer and Grimson early developed one of the most important Gaussian Mixture Models (GMMs)-based algorithms for real-time background subtraction [12], also called MoG (Mixture of Gaussians). He first proposed a multi-modal distribution allowing a complex background modeling, then an algorithm to update this model in real-time. Each Gaussian mode is then assumed to model either background pixel values (the most frequent values) or foreground pixel values (the less frequent ones). This model is able to fit background changes, more or less quickly according to an adjustable learning rate. Bouwmans et al. provided a review and an original classification of the numerous improvements of this initial GMM subtraction algorithm [15].

Reduction dimension and learning techniques were also investigated to extract background. Classic ACP methods were first investigated. For example, Oliver et al. [16] used subspace learning to compress the background (the so-called eigenbackground). In an other way, Lin et al. proposed to learn background model via classification [17]. Wang et al. proposed to improve target detection by coupling it with tracking [18], when Tavakkoli et al. proposed an SVM approach for foreground region detection in videos with quasi-stationary backgrounds [19]. In the same way they proposed a

support vector data description approach for background modeling in videos with quasi-stationary backgrounds [20].

2.2. Recent Approaches

A great number of improvements of the GMM techniques have been proposed. Varadarajan et al. [21] proposed a generalization of the GMM algorithm where the spatial relationship between the pixels is taken into account. Basically, the classification of the pixel does not depend only on its GMM, but also on its neighbors. Gaussian mixtures are associated with regions rather than pixels. Initialization, as the update of the parameters GMM, is significantly different from the classic method.

Another important method based on GMM and proposed by Martins [22] is the method of robust and computationally efficient method, BMOG, which greatly increases the performance of the GMM method [12]. The computational complexity in BMOG is kept low, making it applicable in real time. The solution proposed in this method combines two main contributions: the use of a more appropriate color space than the classic *RGB*, *CIE L*a*b*, as well as a procedure to dynamically adapt the GMM learning rate.

Chen et al. proposed a “sharable” GMM based background subtraction approach [23], in which GMMs are shared by neighbor pixels. Each pixel dynamically searches the best matched model in the neighborhood. This kind of space-sharing way is particularly robust to camera jitter and dynamic background. El-Gammal et al. [24] proposed a novel non-parametric background model that estimates the probability of observed pixel intensity values based on a sample of intensity values for each pixel. This model can adapt to the scene changes quickly and enables it to be very sensitive to the detection of moving targets. In the same way, another approach based on GMM was proposed by Haines and Xiang [25], built on a non-parametric density estimate (DE), which avoids over-/under-fitting. In the literature, outside of GMMs based methods, we also considered optical flow methods to take advantage of temporal continuity in our work. Many methods for background subtraction are indeed based on optical flow (whose concept was first studied in the 1940). We call “optical flow” the estimated velocity field from the variations of the brightness, its calculation is a standard low level processing in frame processing.

The two basic methods of optical flow are Horn and Schunck’s [26], and that of Lucas and Kanade [27]. They have many applications, such as fluid motion analysis in experimental physics, or in higher level processing, such as three-dimensional scene reconstruction.

Many other algorithms have been proposed since the two founding articles. In particular Farneback [28] proposed an extension of the Lucas algorithm, which allows to estimate the movement of all pixels from two consecutive frames.

This optical flow technique has been applied to our problem, in order to exploit the temporal continuity between the consecutive frames of a video. Thus Chauhan and Krishan [29] combine GMMs with the optical flow for bottom extraction.

Then Chen et al. [30] proposed a similar, more sophisticated technique, based on the following steps:

- use of the minimal weight spanning tree, to define a dissimilarity between each pair of pixels in the frame, based on the calculation of the path of maximal color continuity between two pixels. This dissimilarity is used to define a concept of neighborhood more relevant than a simple geometric distance.
- estimation of the optical flow using the fast algorithm of [31], using robust estimators M-smooth.
- computation and fusion of spatial decisions (obtained thanks to the previously calculated dissimilarity) and temporal (obtained thanks to the optical flow).

Recently Javed et al. [32] proposed a background-foreground modeling based on spatio-temporal sparse subspace clustering with success. St-Charles and Biloteau [33] proposed an efficient method (named SuBSENSE) using a spatio-temporal Local Binary Similarity Patterns (LBSP) descriptor

instead of simply relying on pixel intensities as its core component, it keeps memory usage, complexity and speed at acceptable levels for online applications. St-Charles et al. also proposed another method [34] based on local binary patterns as well as color information, also called multi-Q method (multi-objectives method). Moreover, parameters are automatically adjusted by pixel-level feedback loops. Varghese in [35] proposed an efficient method for integrated shadow detection and background subtraction, to identify background, foreground regions, and shadow in a video sequence.

More recently some improvements in dimension reduction techniques have been proposed, as the robust PCA technique (via Principal Component Pursuit) [4], the decomposition into low-rank plus additive matrices for Background/Foreground separation [36]. This make it possible to use of large-scale dataset. Robust low rank matrix decomposition with IRLS scheme (Iteratively Reweighted Least Squares) have also been cited in that case [37]. By computing tensors instead of matrices, dimension reduction techniques make it possible to detect other abnormalities (blob motion, ...). Javed et al. proposed a stochastic decomposition into low rank and sparse tensor for robust background subtraction [38], while Sobral et al. proposed an incremental and multi-feature tensor subspace learning applied for background modeling and subtraction [39].

These last years, many researchers applied deep learning to background subtraction for more robust algorithms [40–45]. Xu et al. proposed an efficient method for dynamic background based on deep auto-encoder networks [43], where the auto-encoder is an artificial neural network used for unsupervised learning of efficient codings. Zhang [42] proposed a deep learning based block-wise scene analysis method equipped with a binary spatio-temporal scene model. Based on the stacked denoising auto-encoder, the deep learning module of the proposed method aims at learning an effective deep image representation encoding the intrinsic scene information. This method can be easily applied to real-time automated video analysis in different scenes due to extremely fast running speed and low memory usage. Brunetti et al. [44] published an interesting survey of techniques for pedestrian detection and tracking using deep learning.

2.3. Fuzzy Approaches

There are many methods and theories to model uncertainties and among them fuzzy modeling (fuzzy sets, or FSs in the sequel). In general, uncertainty as a subjective phenomenon can be modeled by very different theories depending on the causes of uncertainty, the type and amount of available information, the requirements of the observer, and so on. Many models are associated with these uncertainties: probabilistic models, theory of evidence, Dempster-Shafer, fuzzy sets ... Random phenomena are very well taken into account by the probability theory (the applications of “Gaussian mixtures” in processing frames and videos are very numerous to date). Dubois and Prade [46] presented an interesting comparison of these techniques. More recently Sugeno presented a classification of the different uncertainties that can be modeled using FSs: fuzziness, incompleteness, randomness, non-specificity of information, etc. [47].

Fuzzy sets are often associated with vague knowledge modeling (“a high tree”, “a high heat”), but they are not limited to this type of uncertainty. They are quite capable of modeling other forms of inaccuracy, in well-defined situations, by choosing the most suitable model (fuzzy classic models, type-2 models, etc.), as we will present in this paper.

An important drawback of GMM methods relies in the ambiguity of the “most likely mode” (either foreground or background) associated to a given pixel value. This happens when two Gaussian modes are very close and overlap. Some fuzzy methods were introduced to take into account this ambiguity and the imprecision of mode distributions. Particularly, Bouwmans et al. proposed a background modeling method [13] based on type-2 fuzzy GMMs [48]. But as further explained, the introduced fuzziness is not fully exploited, because of a crisp mode selection.

Recently Chiranjeevi et al. proposed several methods for background subtraction using classifiers based on different pixel features (fuzzy correlograms, fuzzy statistical texture features), whose decisions

are then aggregated by fuzzy integrals [49–54]. Fuzzy concepts have been introduced at different levels of the general background subtraction process, as we recall now [55]:

- **Fuzzy Background Modeling:** many recent approaches are based on a multimodal background model. The model usually used is the Gaussian mixture [11]. The parameters are often initialized using a training sequence, which contains insufficient or noisy data. So the parameters are not well determined.

In this context, models of type-1 or type-2 Fuzzy Gaussian mixture [13,56,57] allow to take into account this inaccuracy.

- **Fuzzy Foreground Detection:** this category of methods does not explicitly introduce imprecision in their data or models - that is, in the form of fuzzy sets—but they measure the membership of pixels to the foreground class, through fuzzy degrees, obtained by normalizing the similarity measures (or dissimilarity) in $[0, 1]$. [58–62]
- **Fuzzy Background Maintenance:** the idea is to update the background model according to the fuzzy membership degrees of the pixel to the background and foreground classes, these fuzzy degrees coming from a fuzzy detection. This fuzzy adaptive background maintenance provides robust handling of lighting changes and shadows.

Kim et al. [63] and then Gutti et al. [64] presented background subtraction algorithms exploiting this technique. They both use a fuzzy color histogram (FCH), in order to attenuate the color variations generated by the background movements while highlighting moving objects.

These methods efficiently handle the background noise of scenes with dynamic (temporal) textures.

- **Fuzzy Post-Processing:** a fuzzy inference engine can be applied to the current and previous foreground masks to detach the moving foreground from the dynamic background. In [65] such a method is proposed, which uses a color feature and a median filtering [55].

2.4. Problem and Contribution

In order to take into account the global uncertainty considering background subtraction, and to propose a more robust method to dynamic changes, Darwich in [66] first proposed a method based on type-2 fuzzy GMMs. Its main idea was to consider fuzzy likelihood functions to build fuzzy decisions to achieve more robustness, and to use this fuzziness in a spatial fusion of the fuzzy detection responses in the pixel neighborhood. Indeed, the main problem considering background subtraction is to be robust to dynamic changes (when videos have hard conditions as light change, weather change, ...) and our approach is particularly devoted to that drawback using type-2 fuzzy sets to model the uncertainty. In this paper, we propose to take into account the ambiguity of the mode selection, using a type-2 fuzzy approach. Unlike most methods in literature, the mode selection and the classification are completely fuzzy modeled.

The fuzziness is preserved from its introduction in the GMM model until the final binary decision.

Moreover, uncertainty is exploited to weight the participation of new information “sources” in the decision-making process. This additional information comes from the hypothesis of spatial and temporal homogeneity of the class of a pixel in a video sequence. Thus the method we propose is a decision under uncertainty method using intervals based on type-2 FSs. We now recall the basics of a fuzzy GMM, namely FGMMs, and the method we propose (namely IV-FGMM) in the following section.

3. Toward Fuzziness in Gaussian Mixture Models for Background Subtraction

3.1. GMM for Background Subtraction

The flowchart in Figure 2 shows the different steps of the basic subtraction method based on a mixture of Gaussians. The details of each of these steps are presented below.

Distribution of data coming from several groups may be well modeled by a probabilistic mixture model, with one distribution component (or mode) per group. In the current application, values of a

pixel over time may be characterized by such a mixture model: each mode being either associated to the background or to the foreground.

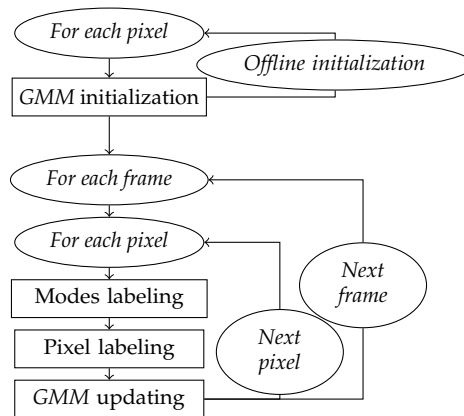


Figure 2. Algorithm of the classic Gaussian Mixture Method.

Gaussian Mixture Model (GMM) is the most popular technique to model the background and foreground state of a pixel [13]. GMM have the ability of universal approximation, because they can fit any density function, if they contain enough mixture [67].

Let I_t be the frame of the video t and p the studied pixel—of coordinates (i, j) —and x_t^p its (RGB) value in the frame I_t . The sample of values of this particular pixel in time is then designated as follows:

$$\{x_1^p, \dots, x_t^p\} = \{I_t(i, j) : 1 \leq t \leq T\}, \tag{1}$$

with T the number of frames.

The GMM associated to pixel p in RGB color space at frame t is composed of K weighted Gaussian functions:

$$f(x) = \sum_{k=1}^K w_{k,t}^p \cdot f_g(x; \mu_{k,t}^p, \Sigma_{k,t}^p), \tag{2}$$

with:

- K : the number of modes of the mixture,
- $f_g(x; \mu_{k,t}^p, \Sigma_{k,t}^p)$: Gaussian density function of the k th Gaussian mode of p , in the frame t ,
- $w_{k,t}^p$: the weight of mode k ,
- $\mu_{k,t}^p$: its center vector,
- $\Sigma_{k,t}^p$: its covariance matrix.

And with f_g the multivariate Gaussian function:

$$f_g(x; \mu_{k,t}^p, \Sigma_{k,t}^p) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_{k,t}^p|}} \exp^{-1/2(x - \mu_{k,t}^p)^T \Sigma_{k,t}^p^{-1} (x - \mu_{k,t}^p)}. \tag{3}$$

To simplify the calculation, the co-variance matrix is often assumed to be diagonal:

$$\Sigma_{k,t}^p = \sigma_{k,t}^p{}^2 I, \tag{4}$$

with I the identity matrix of size 3×3 .

This means that the R , G , B pixel levels are assumed independent with equal variances. This is probably not true, but this assumption avoids expensive matrix inversion with respect to model accuracy.

3.1.1. Step 1: GMM Initialization

This is an optional steps when the ideal is to apply the EM (expectation-maximization) algorithm on a part of the video, but we can also initialize a single mode per pixel (of weight 1), starting from the levels of the first frame.

3.1.2. Step 2: Mode Labeling

Each Gaussian mode is classified as either Background or Foreground. This critical association is obtained from an empirical rule: the more frequent and precise the mode, the more likely it models Background colors.

Specifically, the K modes are sorted according to their priority level $\frac{w_k}{\sigma_k}$. The first K_B modes are then labeled Background. The value of K_B is determined by a threshold $T_b \in [0, 1]$:

$$K_B = \arg \min \sum_{k=1}^K w_{k,t} > T_b. \tag{5}$$

3.1.3. Step 3: Pixel Labeling

The third step is to classify the pixel. In most methods, the pixel is assigned the class of the closest mode center, under the constraint:

$$\|x_t^p - \mu_{k,t}^p\| \leq k_p \sigma_{k,t}^p \tag{6}$$

where k_p is a constant coefficient, to be adapted for each video.

If none of the modes satisfy this constraint, then the lowest priority mode is replaced by a new Gaussian centered on the current intensity x_t^p , with a priori variance weight [12].

3.1.4. Step 4: Updating GMM

Stauffer [12] proposes to operate as follows:

- if a mode i is successfully selected, then the GMM parameters are updated to reinforce this mode:

$$w_{i,t+1} = (1 - \alpha)w_{i,t} + \alpha \tag{7}$$

$$\mu_{i,t+1} = (1 - \rho)\mu_{i,t} + \rho \cdot x_{t+1}^p \tag{8}$$

$$\sigma_{i,t+1}^2 = (1 - \rho)\sigma_{i,t}^2 + \rho \left\| x_{t+1}^p - \mu_{i,t+1} \right\|^2 \tag{9}$$

$$w_{j,t+1} = (1 - \alpha)w_{j,t}, \forall j \neq i, \tag{10}$$

with α a constant learning rate and $\rho = \alpha \cdot f(x_{t+1}^p, \mu_i, \sigma_i)$.

- otherwise, the last distribution is replaced by a new Gaussian mode.

Then a new frame can be processed.

3.2. Type-2 Fuzzy Sets

Ordinary fuzzy sets (or FS) are currently used in image processing [68]. These techniques consider that the spatial ambiguity among pixels has inherent vagueness rather than randomness. However, there remain some sources of uncertainty in ordinary (or precise) fuzzy sets (see [69]): in the fuzzy set design (e.g. in symbolic description), in the fact that measurements may be noisy or that the data used to calibrate the parameters of ordinary fuzzy sets may also be noisy. These uncertainties can be linked to video disturbances described in the introduction. So, since fuzzy sets were introduced by Zadeh [70], many new approaches treating imprecision and uncertainty were proposed (see [46] for a rapid discussion about some of these theories). Among these, is a well-known generalization of an ordinary fuzzy set, the interval-valued fuzzy set, first introduced by Zadeh [71].

3.2.1. Introduction to Type-2 Fuzzy Sets

Type-2 Fuzzy sets have been proposed for the purpose of modeling and minimizing the effects of uncertainties in fuzzy logic systems. Their representation of uncertainty is even more developed than that of previous models. For example, they were used in a type-2 fuzzy controller [72], and the development of efficient classifiers based on “fuzzy c-means” [73]. But their characterization and manipulation are more complex than conventional fuzzy sets, which greatly penalizes their use, especially in video.

Let X be the domain of discourse. Whereas each element of X is mapped to a single value by type-1 fuzzy set membership function, they are mapped to a (type-1) fuzzy set by type-2 fuzzy set ones [74]. The T2FS was introduced by Zadeh in 1975 and was extended (among others) by Karnik and Mendel in 1998 [75].

Karnik et al. defined a fuzzy set \tilde{A} , using a characteristic function $\mu_{\tilde{A}} : X \times [0, 1] \rightarrow [0, 1]$ with two parameters:

$$\tilde{A} = \{(x, u), \mu_{\tilde{A}}(x, u), \forall x \in X, \forall u \in J_x \subseteq [0, 1]\}. \tag{11}$$

Each value $x \in X$ is associated to a fuzzy number in the domain $[0, 1]$, rather than to a singleton—too precise—(like type-1 fuzzy sets). So, for the considered x , $\mu_{\tilde{A}}$ associates to any element u of this domain, a membership degree—now a singleton—of $[0, 1]$.

Karnik et al. further defined two simpler membership functions for breaking out type-2 fuzzy sets:

- the primary membership function: for each $x \in X$, it associates the minimal interval $J_x \subseteq [0, 1]$ containing the set of membership degrees u such that $\mu_{\tilde{A}}(x, u) > 0$; this interval is the support of the type-1 fuzzy number associated to x by \tilde{A} .
- the secondary membership function: for each $x \in X$, it is the membership function of the type-1 fuzzy set associated to x by \tilde{A} . Such a secondary membership function is drawn in Figure 3, in black color.

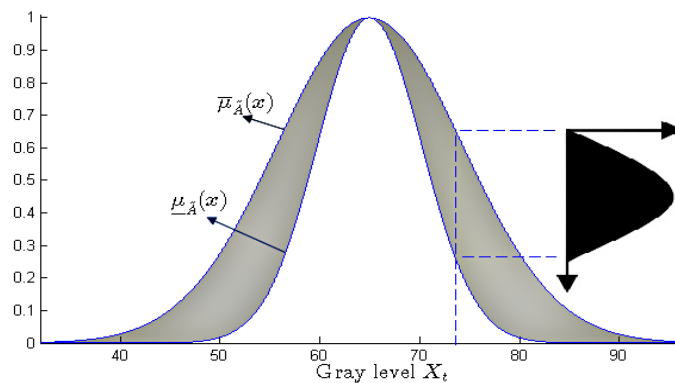


Figure 3. A type-2 fuzzy set.

Finally, Karnik et al. denote the region covering the set of J_x intervals defined on X (the primary functions of \tilde{A}) using the term footprint of uncertainty (or *FOU*, cf. Figure 3, in gray):

$$FOU(\tilde{A}) = \bigcup_{x \in X} J_x. \tag{12}$$

The surface of this footprint measures the “fuzzy amount” of the primary membership functions. It is bounded by an upper and a lower function, which are type-1 fuzzy membership functions. They account for the global imprecision of the primary membership functions.

3.2.2. Interval-Valued Fuzzy Sets

General T2FS framework may be simplified, by defining secondary membership functions as crisp intervals: such T2FSs are called Interval-Valued Fuzzy Sets (IVFSs) [46].

The Interval-Valued Fuzzy Sets have been proposed to overcome the excessive precision of the membership degrees of type-1 fuzzy subsets [76] but with less computational complexity than T2FSs.

Their principle consists in framing the membership functions of the type-1 fuzzy sets, by two membership functions: one which limits the possible membership degrees lower, the other higher.

Two membership functions are enough to model an IVFS:

- T2FS upper membership function: $\overline{f}_g(x)$;
- T2FS lower membership function: $\underline{f}_g(x)$.

This simplification of T2FSs represents a crucial advantage for a software implementation. This is why IVFSs are the most used T2FSs in frame processing [77].

Those two membership functions are frequently built from a single ordinary Gaussian function. A common procedure consists in introducing imprecision in its mean parameter [78]. We then consider that the true value μ^* is not precisely known, but that it belongs to a half- Δ -length interval centered on μ where Δ is a value $\in [1, 3]$ depending on the video :

$$\mu^* \in [\underline{\mu} = \mu - \Delta, \overline{\mu} = \mu + \Delta]. \tag{13}$$

Lower and upper membership functions are then obtained by shifting the mean value inside its domain, and retaining maximal and minimal values of the Gaussian function f_g :

$$\overline{f}(x; \mu, \sigma, \Delta) = \max_{\mu^* \in [\mu - \Delta, \mu + \Delta]} f_g(x; \mu^*, \sigma), \tag{14}$$

$$\underline{f}(x; \mu, \sigma, \Delta) = \min_{\mu^* \in [\mu - \Delta, \mu + \Delta]} f_g(x; \mu^*, \sigma). \tag{15}$$

Figure 4 illustrates this building. Let us note that the larger the mean interval, the larger the FOU (and hence the uncertainty) of the resulting T2FS.

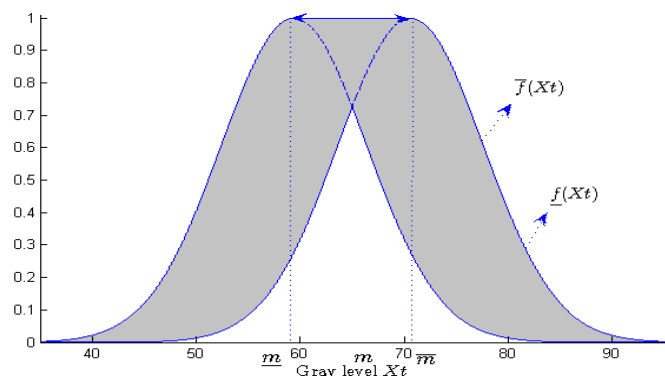


Figure 4. Gaussian interval type-2 fuzzy set (uncertain mean).

Upper and lower membership functions could be defined—in a way quite similar—by considering an imprecise variance [48]. In this paper, we consider only the previous procedure.

3.3. Background Modeling Using T2-FGMM with Uncertain Mean

Bouwman et al. introduced Type-2 Fuzzy sets to handle uncertainty in GMM mixtures used for background subtraction [13]. They applied the GMM uncertain model proposed by Zeng et al. in paper Type-2 Fuzzy Gaussian Mixture Model [48].

In fact, Fuzzy type-2 model only appears in the final decision step of the method. Previous steps consist in a classic GMM approach.

The algorithm is shown in Figure 5. In the next part, we only focus to the two modified steps (with respect to the classic GMM method), grayed out, with the steps 1 is the same initialization steps in GMM classic method.

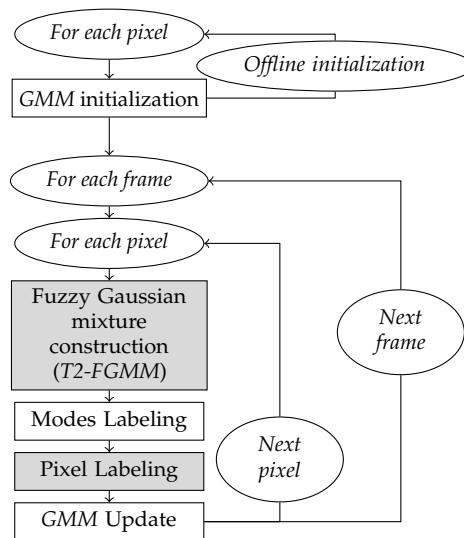


Figure 5. T2-FGMM algorithm.

3.3.1. Step 2 : Fuzzy GMM Construction

Figure 6 illustrates how the Fuzzy Gaussian mode is built from Equation (14).

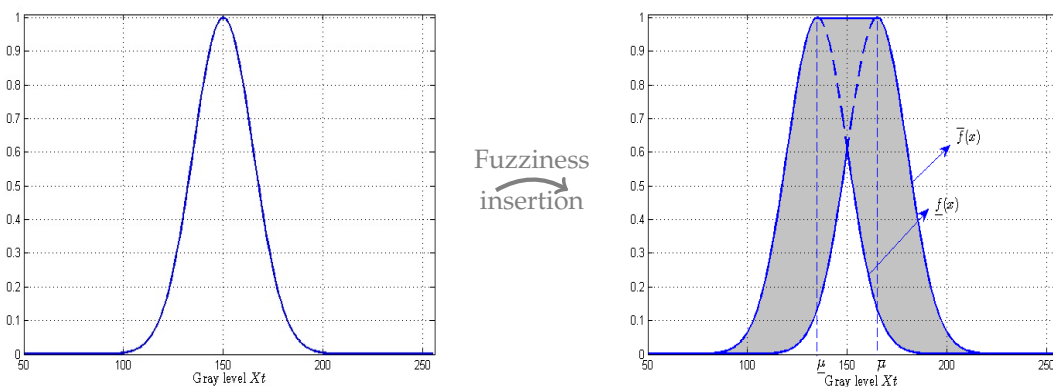


Figure 6. Fuzzy insertion into a Gaussian mode.

3.3.2. Step 3: Pixel Labeling

Most methods estimate the pixel class from the distance between its gray level and the center of the closest mode. This means that the pixel is labeled according to the first mode compatible with its value (gray-level in our case).

Type-2 fuzzy sets are introduced in this step, with two membership functions f and \bar{f} . By construction, they remain centered on the center μ of the crisp-valued Gaussian function f .

Bouwman et al. [13,56,57] based the class labeling on the length of the “log-likelihood” interval at the x level of the current pixel. The pixel is labeled with the label of the “best matching” Gaussian mode, i.e. the first ranked mode (according to $\frac{w_k}{\sigma_k}$) whose membership bounds in x make this inequality true:

$$H(x) = \left| \ln(\bar{f}(x)) - \ln(\underline{f}(x)) \right| < k_p \sigma, \tag{16}$$

with k_p a constant factor.

If this inequality cannot be asserted for any mode in x , then the last sorted mode is replaced by a new one, centered in x .

This choice is clearly justified by Bouwmans et al. They show that indicator $H(x)$ is a decreasing monotonous function of distance $|x - \mu|$:

$$H(x) = \begin{cases} 2 \frac{k_m |x - \mu|}{\sigma}, & \text{if } x < \underline{\mu} \text{ or } x > \bar{\mu} \\ \frac{|x - \mu|^2}{2\sigma^2} + \frac{k_m |x - \mu|}{\sigma} + k_m^2, & \text{otherwise.} \end{cases}$$

So, the larger the normalized (log-likelihood) interval, the closer the pixel-value to the center of the mode. Consequently, the larger the interval, the more likely the mode.

Let us note that, because of their prior sorting, modes with highest weights and lowest variance—likely Background modes—are tested first: if a pixel is compatible with such a mode, it is then labeled Background.

4. The Proposed IV-FGMM-ST (Interval-Valued Fuzzy GMM spatio-Temporal) Method

4.1. Originality of the Proposed Method

The algorithm proposed in this paper mainly differs from previous work through the following points:

- T2FS uncertainty is used to handle the possible ambiguity of the mode estimation.
- Fuzzy modeling is preserved until the final binary decision.
- Uncertainty is used to weight the participation of several “information sources” in a fusion classification process.

The proposed algorithm is illustrated in Figure 7. Original steps are grayed out.

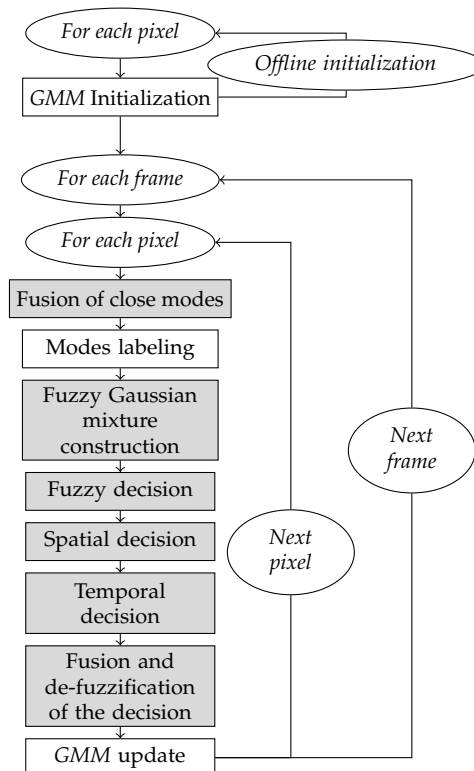


Figure 7. Algorithm of the proposed method.

4.2. Fuzzy Decision

The main goal considering the fuzzy decision is to deal with a fuzzy decision for each pixel instead of the binary decision used in most methods, in order to avoid wrong classifications that come from the ambiguity in the decision process.

Aggregation of Imprecise Intervals

Unlike the T2-FGMM method—also based on a fuzzy GMM—model, we propose to construct a “preference” fuzzy measure between the 2 possible decisions (background, Foreground), associated with a “rejection” score, designed to ensure the lack of knowledge.

The first modification lies in the Pixel labeling Step.

For each mode k , the IVFS associates to pixel p in frame t —of value x_t^p —a normalized density interval: $[f_k(x_t^p), \bar{f}_k(x_t^p)]$.

All mode intervals are aggregated using a weighted average, according to their label. Weights are defined by Zeng’s priority levels (used in the mode ranking):

$$w'_k = \rho \left(\frac{w_k}{\sigma_k} \right), \tag{17}$$

with $\rho(x) = x^2$.

More precisely, two label intervals are built by aggregation: $\mathbf{F}_t(p)$ to assess the hypothesis “pixel value matches a Foreground region”, and $\mathbf{B}_t(p)$ to assess the opposite hypothesis. A supplementary interval—called \mathbf{R} for reject—is built to represent non-significant likelihood levels, from an a priori threshold $k_r \in \mathbb{R}^+$:

$$\mathbf{F}_t(p) = \frac{1}{\sum_{k=1}^K w'_k} \left[\sum_{k=K_B+1}^K w'_k f_k(x_t^p), \sum_{k=K_B+1}^K w'_k \bar{f}_k(x_t^p) \right], \tag{18}$$

$$\mathbf{B}_t(p) = \frac{1}{\sum_{k=1}^K w'_k} \left[\sum_{k=1}^{K_B} w'_k f_k(x_t^p), \sum_{k=1}^{K_B} w'_k \bar{f}_k(x_t^p) \right], \tag{19}$$

$$\mathbf{R} = [0, k_r]. \tag{20}$$

Then the fuzzy decision between Foreground and Background alternatives is built by comparing both intervals $\mathbf{F}_t(p)$ and $\mathbf{B}_t(p)$.

For this purpose, we use a variant of the Sengupta’s acceptability index [79]. This index assesses the “grade of acceptability of the first interval to be inferior to the second interval”:

$$\mathcal{A}_{\ominus}(\mathbf{A}, \mathbf{B}) = \frac{c(\mathbf{B}) - c(\mathbf{A})}{\delta(\mathbf{B}) + \delta(\mathbf{A})}, \text{ if } c(\mathbf{A}) \leq c(\mathbf{B}) \tag{21}$$

with:

- $c(\mathbf{A})$ denotes the central value of interval \mathbf{A} .
- $\delta(\mathbf{A})$ denotes its half-width.

Note that this index only applies if the center of \mathbf{A} is lower than that of \mathbf{B} .

We propose the following (slight) variant, to eliminate this condition, and normalize the measure in $[0, 1]$ [66]:

$$\mathcal{A}'_{\ominus}(\mathbf{A}, \mathbf{B}) = \begin{cases} \mathcal{A}_{\ominus}(\mathbf{A}, \mathbf{B}), & \text{if } c(\mathbf{A}) \leq c(\mathbf{B}) \\ & \text{and } \mathcal{A}_{\ominus}(\mathbf{A}, \mathbf{B}) \leq 1 \\ 0, & \text{if } c(\mathbf{A}) > c(\mathbf{B}) \\ 1, & \text{otherwise.} \end{cases} \tag{22}$$

Finally, the fuzzy decision between the Foreground and Background assumptions of the p pixel is constructed by comparing the intervals $\mathbf{F}_t(p)$ and $\mathbf{B}_t(p)$ according to the acceptability index variant:

$$D_t(p) = \begin{cases} \frac{1}{2} [1 + \mathcal{A}'_{\ominus}(\mathbf{B}_t(p), \mathbf{F}_t(p))], & \text{if } c(\mathbf{B}_t(p)) < c(\mathbf{F}_t(p)) \\ \frac{1}{2} [1 - \mathcal{A}'_{\ominus}(\mathbf{F}_t(p), \mathbf{B}_t(p))], & \text{if } c(\mathbf{F}_t(p)) < c(\mathbf{B}_t(p)) \\ \frac{1}{2}, & \text{if } c(\mathbf{F}_t(p)) = c(\mathbf{B}_t(p)). \end{cases} \quad (23)$$

Let us note that values higher (respectively lower) than 0.5 induce a foreground (r. Background) class, while values close to 0.5 just indicate a lack of preference.

To complete this fuzzy decision measure, a confidence level $E_t(p)$ is built, by comparing both intervals to a rejection interval \mathbf{R} . The idea behind this is to make sure that at least one of the two intervals is significant enough.

$$E_t(p) = \max \left\{ \mathcal{A}'_{\ominus}(\mathbf{R}, \mathbf{F}_t(p)), \mathcal{A}'_{\ominus}(\mathbf{R}, \mathbf{B}_t(p)) \right\}. \quad (24)$$

4.3. Complementary Decisions

The previous fuzzy decision give us a first estimate of the object pixels. But in a noisy video, or with a complex (dynamic) background, some decisions will remain ambiguous, due to a lack of information. We therefore decide to reinforce the decision by considering spatial or temporal coherence.

4.3.1. Fuzzy Spatial Decision

Here, we consider the spatial continuity hypothesis: pixels neighbors are likely to belong the same class. A neighborhood is defined using a window $W_s = 3 \times 3$ (for example).

Each pixel p in the frame t is now characterized by a fuzzy decision $D_t(p) \in [0, 1]$ and a confidence level $E_t(p) \in [0, 1]$. The decision value indicates which class Foreground and Background is preferred. And the confidence value associated with this decision establishes its future contribution to spatial decision making.

The proposed procedure consists in a spatial fusion: fuzzy decisions of neighbor pixels are merged to avoid false detection on isolated pixels.

Firstly, the window W_s is centered on the current pixel p and filled with its similarities with the 8 closest neighbors. The similarity between the p pixel and its neighbor p' is defined using a Gaussian kernel:

$$S(p, p') = e^{-\left(\frac{x_t^p - x_t^{p'}}{\sigma_s}\right)^2}, \quad (25)$$

with σ_s a tuning parameter.

The similarities are then weighted by the confidence degrees $E_t(p')$ of all neighbors p' of p .

The aggregated fuzzy decision of p is defined as a weighted average Figure 8:

$$D_t^s(p) = \frac{1}{\sum_{p' \in \mathcal{N}(p)} e_{p,p'}} \sum_{p' \in \mathcal{N}(p)} e_{p,p'} \cdot D_t(p), \quad (26)$$

with:

- $\mathcal{N}(p)$: the neighboring pixels of p , defined by the neighborhood window W_s ;
- $e_{p,p'} = S(p, p') \cdot E_t(p')$: the degree of contribution of pixel p' to the spatial decision of the pixel p Figure 9.

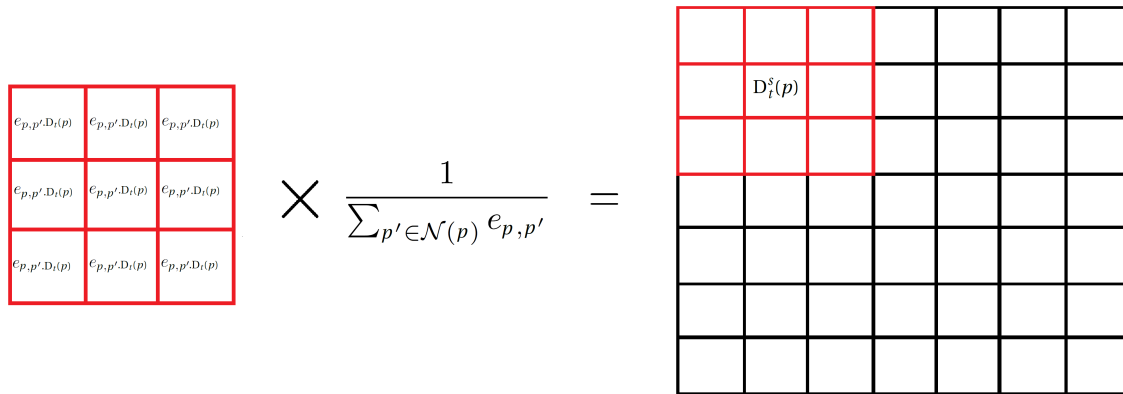


Figure 8. Spatial Fusion: aggregation of initial fuzzy decisions.

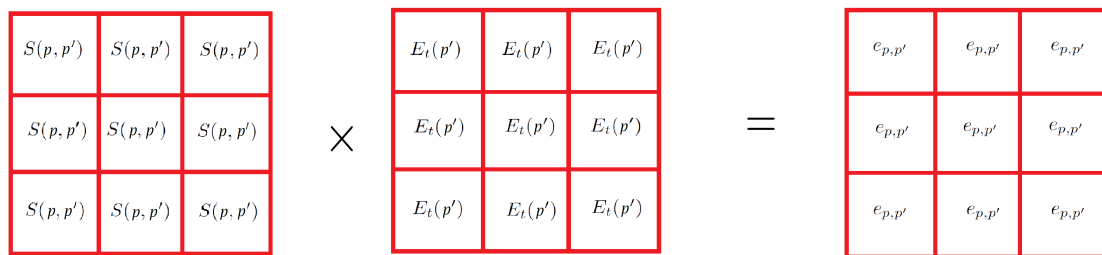


Figure 9. Spatial fusion: weight of initial fuzzy decisions.

4.3.2. Fuzzy Temporal Decision

We propose to integrate an optical flow calculation in the method, to exploit the temporal continuity of the decisions. To summarize, the pixel velocity field is estimated, based on the hypothesis of brightness conservation (that is to say that the brightness of a physical point of the frame—likely to move—does not change over time).

In our method, we use an efficient and robust optical flow estimation algorithm, designed by Gunnar Farneback [28].

We propose the following algorithm, to enrich the class information of pixel p of the frame t :

- Computation of the optical flow, for each frame $t \neq 1$ (because the first frame has no antecedent frame), so as to associate to each pixel p its antecedent p' in the previous frame.
- Computation of the temporal confidence and decision degrees of the pixel, according to its antecedent:

$$\begin{aligned} E_t^{temp}(p) &= E_{f_{t-1}}(p'), \\ D_t^{temp}(p) &= D_{f_{t-1}}(p'), \end{aligned} \tag{27}$$

with:

- $E_{f_{t-1}}(p')$, the final confidence degree of the antecedent pixel of p in frame $t - 1$.
- $D_{f_{t-1}}(p')$, the final decision degree of the antecedent pixel of p in frame $t - 1$.

4.4. Merging Decisions

First spatial and temporal fuzzy decisions are merged to a fuzzy final decision, which is then “defuzzified”.

4.4.1. Final Fuzzy Decision

The final—spatio—temporal—decision is defined as a weighted average:

$$\begin{aligned} D_f(p) &= \frac{D_t^s(p) \cdot E_t^s(p) + E_t^{temp}(p) \cdot D_t^{temp}(p)}{E_t^s(p) + E_t^{temp}(p)}, \\ E_f(p) &= \frac{1}{2} \left(E_t^s(p) + E_t^{temp}(p) \right), \end{aligned} \tag{28}$$

where $D_f(p)$ and $E_f(p)$ respectively denote the decision and the final fuzzy confidence of pixel p .

4.5. Defuzzification of the Fuzzy Final Decision

The binary decision is obtained by rounding $D_f(p)$ towards its nearest integer if the final confidence level is high enough, else set to 1 (Foreground) otherwise:

$$D_t^b(p) = \begin{cases} 0 \text{ (Foreground)} & \text{if } D_f(p) \leq 0.5 \text{ and } E_f(p) > k_e \\ 1 \text{ (Background)} & \text{if } D_f(p) > 0.5 \text{ and } E_f(p) > k_e \\ 1 & \text{if } E_f(p) \leq k_e. \end{cases} \tag{29}$$

After this final decision step, a binary mask frame can be fulfilled, to mark the locations of the Foreground pixels.

4.6. Other Step Changes (with Respect to T2-FGMM Method)

4.6.1. GMM Update

In T2-FGMM-like methods, updating the GMM model consists in reinforcing the unique mode associated to the pixel (the most “likely” one).

In our case, the final decision comes from the aggregation of several complementary decisions, conceived from different GMM models (according to the pixels or to the frame), and also from several modes (according to several Background and Foreground intervals).

However, we propose to apply such a single mode reinforcement. It first involves identifying the mode to reinforce. In practice, we look for the most likely mode k^* of the same class (Foreground or Background):

$$k^* = \begin{cases} \arg \max_{k \in \{1, \dots, K_B\}} f_k(x_t^p), & \text{if pixel } p \text{ in frame } t \text{ is finally ranked Background,} \\ \arg \max_{k \in \{K_B+1, \dots, K\}} f_k(x_t^p), & \text{otherwise.} \end{cases}$$

The selection fails when the likelihood of the optimal mode is below the R rejection threshold. In this case, a new Gaussian mode replaces the lowest priority one (lowest $\frac{w_k}{\sigma_k}$), centered on the pixel level x_t^p , with a weight initialized to β .

Once the update is carried out, the algorithm resumes at the beginning of the loop, to the step of merging the close modes (cf. Figure 7).

4.6.2. Fusion of Close Modes

This step was proposed to remedy to overlapping Gaussian modes. The stage is divided into two parts:

- Computation of similarity $S(a, b)$ between each pair of modes (a, b) . Two modes are similar, if the center of one mode has a high (crisp) likelihood degree to the other mode:

$$S(a, b) = \max \left(f_g \left(\mu_{a,t}^p; \mu_{b,t}^p, \Sigma_{b,t}^p \right), f_g \left(\mu_{b,t}^p; \mu_{a,t}^p, \Sigma_{a,t}^p \right) \right). \tag{30}$$

- Merging modes: each pair of Gaussian modes whose similarity is strong enough ($S(a, b) > k_s$) is merged into a single new mode, following the common formula [80]:

$$\begin{aligned}
 w &= w_a + w_b; \\
 \mu &= \frac{w_a * \mu_a + w_b * \mu_b}{w_a + w_b}; \\
 \sigma^2 &= \frac{w_a * \sigma_a^2 + w_b * \sigma_b^2}{w_a + w_b}.
 \end{aligned}
 \tag{31}$$

5. Validation Steps and Comparison to Methods Based on GMM and Fuzzy GMM

5.1. Base of Reference Videos

In this work we use the complete database from the site [81], which provides realistic and hard-to-process videos, captured by camera. They are representative of indoor and outdoor scenes captured in surveillance, smart environment and video database scenarios. It is arranged in several video categories: thermal camera, shadows, intermittent foreground motion, camera jitter, dynamic background, bad weather, low frame rate, PTZ camera, night, baseline, turbulence. All quantitative evaluations of the proposed method were computed on [82], University of Littoral’s computing server (Université du Littoral Côte d’Opale: Dunkerque, France).

5.2. Performance Indicators

Each pixel is subject to a decision, either right or wrong, which can be classified in one of these 4 categories (see Table 1):

Table 1. Ranking of decisions.

Reality	Decision	
Type	Foreground	Background
Foreground	TP	FN
Background	FP	TN

Once cumulated over all the video frames, each class number assesses a performance of the Foreground/Background recognition procedure:

- *TP*: True Positives number (i.e., right Foreground decisions).
- *FP*: False Positive number (i.e., wrong Foreground decisions).
- *TN*: True Negative number (i.e., right Background decisions).
- *FN*: False Negatives number (i.e., wrong Background decisions).

More synthetic performance indicators are defined as combinations of these numbers (see Table 2).

Table 2. Performance indicators.

Parameter	Formula
Recall (or detection rate)	$R_e = \frac{TP}{TP + FN}$
Precision	$P_r = \frac{TP}{TP + FP}$
Specificity	$S_p = \frac{TN}{TN + FP}$
Ratio of erroneous classifications	$PBC = \frac{FN + FP}{TP + FN + FP + TN}$
F-measure	$F\text{-measure} = \frac{2 \cdot P_r \cdot R_e}{P_r + R_e}$

These usual parameters have the following meanings:

- R_e : proportion of pixels in the background correctly identified.
- P_r : proportion of foreground pixels correctly identified.
- F -measure: geometric mean of Recall and Precision; it takes its values between 0 and 1, 1 being associated with maximal performance.
- PBC : proportion of decision errors.
- S_p : proportion of pixels in the background false identified.

5.3. Validation of the Method Steps

To validate the different steps of the method, we worked on a single reference video: the video Canoe. From the video category Dynamic background (from the site [3]), it shows the passage of a boat on a river, with changing reflections. This is a small video (dimensions 320×240), with 30 frames per second, and 1189 frames in total.

We have selected it because it is a typical video of its category, which does not normally offer any particular difficulty to the subtraction algorithms. The parameters of our method have been optimized to best adapt to this video. The list of parameters is recalled in the Table 3.

Table 3. Parameters of the method.

Parameter	Value
K	Number of modes (Gaussian)
α	Learning rate
β	Weight of new modes
T_b	Background threshold
k_m	Quantity of footprint
k_r	Upper bound of the rejection interval
k_e	Threshold of the degree of confidence
k_s	Likelihood threshold
σ_n	The standard deviation of the new modes
W_s	Size of the neighborhood window

We optimized these parameters in order to ensure the best object detection, which is evaluated by the-synthetic-indicator of F -measure.

Below is the procedure followed:

- Generation of a set of parameters sets (2443 sets, obtained by combining heuristically chosen parameter values). Each set of parameters assigns a unique combination of values to all parameters.
- Selection of the best parameters set, according to the complete variant (spatio-temporal) method: this set is obtained by maximizing the F -measure (cf. Table 4).
- Launching calculations with the optimized parameter set, with several variants of our IV-FGMM method (obtained by activating/deactivating certain steps).

Table 4. Optimized parameters set for canoe video.

Parameter	Value
K	3
α	0.01
β	0.001
T_b	0.85
k_m	2
k_r	0.07
k_e	0.25
k_s	0.75
σ_n	36
W_s	5

5.3.1. Results Analysis

Table 5 shows the performances of the different variants of our method on the video Canoe, obtained with the optimized parameter set.

Table 5. Performance of each variant, video Canoe.

Method	TP%	FP%	FN%	TN%	Specificity	PBC	Recall	Precision	F-Measure
Binary	23.770	34.294	2.431	939.504	0.965	3.678	0.907	0.410	0.564
Binary/temporal	23.469	18.861	2.741	954.929	0.981	2.163	0.895	0.554	0.685
Binary/spatial	20.145	0.171	3.565	976.118	1	0.375	0.849	0.992	0.915
Binary/spatio-temporal	20.038	0.146	3.794	976.023	1	0.393	0.841	0.993	0.911
Fuzzy	22.651	30.915	2.646	943.787	0.968	3.348	0.895	0.423	0.574
Fuzzy/temporal	23.084	20.461	2.551	953.903	0.979	2.296	0.900	0.530	0.667
Fuzzy/spatial	22.172	0.383	2.416	975.029	1	0.280	0.902	0.983	0.941
Fuzzy/spatio-temporal	2.2233	0.260	2.419	975.087	1	0.268	0.902	0.988	0.943

5.3.2. Interest of Spatial and Fuzzy Parts of the Method

We can easily note that the decision process is mainly improved (F -measure of 0.911 for the binary/spatial variant against 0.564 for the binary variant) when using the integration of spatial information.

To realize—more qualitatively—the importance of the different steps of the method, we show now some classification results on an image of the sequence: the image 965, where we can observe the passage of the canoe (cf. the 2 upper images of the Figure 10, showing the Figure 10a and its expected ground Truth Figure 10)

The images in the Figure 10 make it possible to visualize the results of the detection of the objects, according to the various method variants studied. The advantage of the (spatial/fuzzy) merge step can be observed: it eliminates many classification errors.

The temporal contribution is less obvious (0.943 for the fuzzy/spatio-temporal variant). It would probably be a little more complex to exploit the optical flow, to obtain a significant gain.

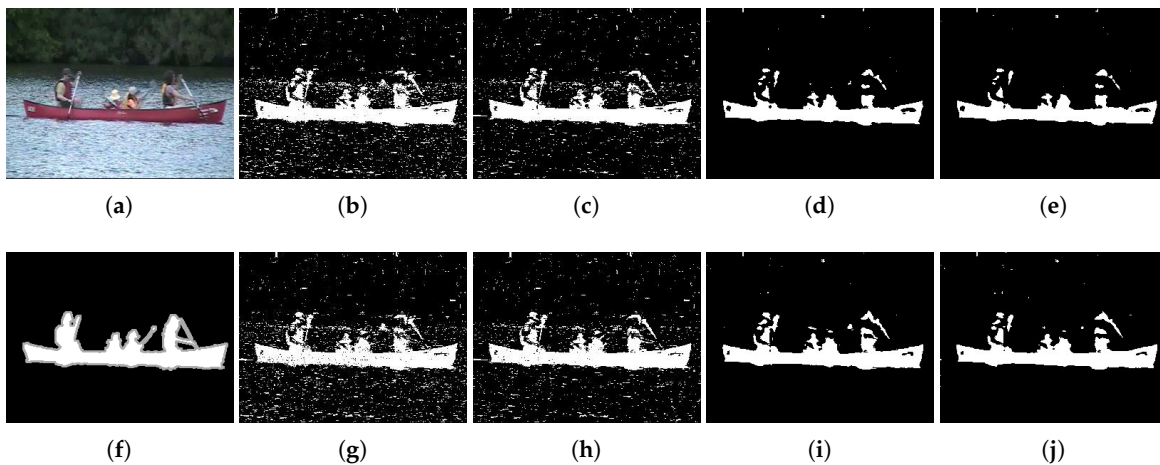


Figure 10. Image results of our method according to the used steps. (a) Initial image; (b) Binary (F -measure = 0.564); (c) Bin./Temporal (F -measure = 0.685); (d) Bin./Spatial (F -measure = 0.915); (e) Bin./Spat./Tempo. (F -measure = 0.911); (f) Ground truth; (g) Fuzzy (F -measure = 0.574); (h) Fuzzy/Tempo. (F -measure = 0.667); (i) Fuzzy/Spat. (F -measure = 0.9); (j) Fuzzy/Spat./Tempo. (F -measure = 0.943)

5.4. Optimizing the Parameters in the Video Category “Dynamic Background”

To evaluate our method, we have optimized again the parameter set for the the most complete version of our method (i.e. the fuzzy spatio-temporal), over the whole dynamic background video set, by maximizing the F -measure (F_m in the tables).

In that, we did not exactly apply the official protocol of the site ChangeDetection: it would have been necessary to optimize the average F -measure on the videos of all the categories. But given the large number of parameter combinations to test, we had to restrict ourselves to the category we were most interested in (dynamic background).

The optimized parameter set is presented in Table 6.

Table 6. Optimized parameter set on the video dynamic background category.

Parameter	Value
K	3
α	0.01
β	0.04
T_b	0.85
k_m	2.5
k_r	0.05
k_e	0.25
k_s	0.75
σ_n	36
W_s	5

The set of the selected methods used for comparison is presented in Table 7 below.

Table 7. Methods used in the comparative study.

Method	Reference
GMM based method	
GMM-Zivkovic (GMM-Z)/CDNET	Zivkovic [2004] [83]
SharedModel/CDNET	Chen et al. [2015] [23]
KDE-El-Gammal(KDE-G)/CDNET	El-Gammal et al. [2000] [24]
Haines(DP-GMM)/Other	Haines [2000] [25]
GMM-Stauffer-Grimson (GMM-SG)/CDNET	Stauffer-Grimson [1999] [12]
RMoG (Region-based MoG)/CDNET	Varadarajan et al. [2013] [21]
BMOG /CDNET	Martins et al. [2017] [22]
Fuzzy based method	
T2-FGMM-UM/BGS	El Baf et al. [2008] [56]
T2-FGMM-UV [2008]/BGS	El Baf et al. [2008] [56]
T2-FGMM-UMRF/BGS	Zhao et al. [2012] [84]
T2-FGMM-VMRF [2012]/BGS	Zhao et al. [2012] [84]
Fuzzy Sugeno Integral (FSI)/BGS	Zhang and Xu [2006] [61]
Fuzzy Choquet Integral (FCI)/BGS	El Baf et al. [2008] [62]
Fuzzy Gaussian (FG)/BGS	Sigari et al. [2008] [58]
Deep learning method	
CNN	Babae et al. [2018] [41]
Local binary pattern	
LOBSTER/CDNET	St-Charles and Biloteau [2014] [33]
SuBSENSE [2015] /CDNET	St-Charles et al. [2015] [34]
Shadow detection	
SBBS/CDNET	Varghese and Sreelekha [2017] [35]

5.5. Results Comparison

The results of methods GMM-GS, GMM-Z, KDE-G, RMoG, and BMOG directly come from [81] from the site [81].

For the deep learning method CNN, the average F-measure score for the “dynamic background” category is taken from the article [41].

For the other methods, we used the BGSLibrary [85] library, with the default settings for each method.

An exception is made considering the T2-FGMM method (cf. Section 3.3): the scores being surprisingly low, we tried to optimize them. We were expecting performances closer to ours, because of the similarities between the two methods. The parameters were obtained after maximizing the F-measure on all videos of the “dynamic background” category (on a set of heuristically defined parameter sets). The combinations obtained for all T2-FGMM variants are given in Table 8.

Table 8. Optimized parameters for T2-FGMM methods.

Method	Optimized Parameters			
	k_m	k_v	α	k_p
T2-FGMM-UM	1.9	-	0.001	6
T2-FGMM-UV	-	0.95	0.001	40
T2-FGMM-UMRF	1.9	-	0.001	5
T2-FGMM-VMRF	-	0.95	0.001	40

The Table 9 gathers all the results useful for comparing the methods.

Table 9. *F*-measures of all videos.

Method	Fall	Canoé	Overpass	Fontaine1	Fontaine2	Boats	F_m
GMM based method	–	–	–	–	–	–	–
GMM-Z [83]	0.423	0.885	0.867	0.081	0.791	0.747	0.632
RMoG [21]	0.673	0.935	0.901	0.203	0.865	0.832	0.735
BMOG [22]	0.691	0.950	0.962	0.381	0.932	0.838	0.792
SharedModel	0.893	0.878	0.824	0.780	0.936	0.620	0.822
KDE-G [24]	0.308	0.882	0.824	0.105	0.823	0.632	0.596
DP-GMM [41]	–	–	–	–	–	–	0.813
GMM+SG [12]	0.435	0.881	0.871	0.076	0.803	0.728	0.633
Fuzzy based method	–	–	–	–	–	–	–
IV-FGMM-ST	0.637	0.920	0.891	0.075	0.939	0.570	0.672
T2-FGMM-UM [56]	0.065	0.129	0.419	0.031	0.485	0.053	0.197
T2-FGMM-UV [56]	0.015	0.208	0.384	0.056	0.529	0.011	0.200
T2-FGMM-UMRF [84]	0.065	0.194	0.390	0.021	0.479	0.058	0.201
T2-FGMM-VMRF [84]	0.015	0.203	0.374	0.051	0.502	0.009	0.192
FSI [61]	0.225	0.820	0.474	0.068	0.508	0.379	0.201
FCI [62]	0.215	0.836	0.403	0.051	0.533	0.429	0.411
FG [58]	0.072	0.131	0.092	0.005	0.010	0.018	0.056
Deep learning method	–	–	–	–	–	–	–
CNN [41]	–	–	–	–	–	–	0.876
Local binary pattern method	–	–	–	–	–	–	–
SuBSENSE [34]	0.866	0.792	0.857	0.753	0.944	0.693	0.817
LOBSTER [33]	–	–	–	–	–	–	0.567
Shadow detection method	–	–	–	–	–	–	–
SBBS [35]	0.877	0.936	0.909	0.728	0.933	0.492	0.812

5.6. Analysis of Results

Table 9 first shows that our method gives comparable, if not better, results to the other methods, including those based on a GMM model (fuzzy or not fuzzy one), except for the methods BMOG and RMoG.

Compared to our method, RMoG makes a more extensive use of the pixel neighborhood: first in the initialization step, then in the update phase. As BMOG, it then uses a more appropriate color space than the classic RGB—CIE L^*a^*b —as well as a dynamic learning rate mechanism. In the SharedModel method, Chen et al. use all matched models in a $N \times N$ region to find an optimal model. In our case, an exhaustive search for all the matched models in a $N \times N$ region around the center pixel using the model of maximum probability for the foreground and background model is performed. The DP-GMM method has a good *F*-measure score compared to our method, but it's a non-parametric DE method that considers an enrichment step. In this method, a Markov random field is built considering a node for each pixel, connected using a four-way neighbourhood. It is a binary labeling problem, where each pixel in each frame either belongs to the foreground or to the background. The uncertainty is taken into account in a different way in our method (fuzzy methodology), and we use different steps to make the method robust using adjustable parameters. SBBS method has a good result when we modeled the background at pixel level with a collection of previously observed background pixel values, and we use a ghost suppression mechanism with median filtering. SuBSENSE method also has a good result when St-Charles et al. use spatio-temporal binary features as well as color information to detect changes.

In this evaluation CNN achieves the maximal averaged *F*-measure (0.876). It is a very efficient method, but it is a supervised method which needs a huge learning database, whereas all other methods are unsupervised. The comparison is consequently not “fair”.

Apart from these four methods, which exploit information of a different nature, our method obtains very good results.

In particular, it achieves one of the best scores with video Fountaine2 (0.939), higher than the other methods. It is a relatively complex video, because of the movement of water particles coming from the large fountain.

We note here that video Fountaine1 is a difficult challenge for all methods. Its unusual complexity is due to the changing pace of its four fountains, the very small size of the foreground, and its color, very close to that of the background.

6. Comparison of Calculation Time for the Different Methods

Given the algorithmic complexity of some methods, we performed our comparison on a “lightened” video. We built it by extracting a sequence of consecutive images from the video fountaine1 (between 50 and 5016, with 30 frames per second), and then reducing them to 160×128 .

We then tested all GMM based methods (except RMoG, BMOG, SuBSENSE, DP-GMM, see the following comment) on this video sequence, with a computer equipped with a *Intel CPU (R) Core (TM) i7 – 4500U CPU@1.80 Ghz 2.40 Ghz*, with 16 GB of RAM.

Considering our method, we compared several versions:

- With parallelization on 4 cores (their name is suffixed by -P), and without parallelization (without suffix).
- The fuzzy decision method (IV-FGMM, IV-FGMM-P), with temporal decision only (IV-FGMM-T, IV-FGMM-TP), with spatial decision alone (IV-FGMM-S, IV-FGMM-SP), and the full space-time version (IV-FGMM-ST, IV-FGMM-ST-P).

We note that our method ranks 7th, which is very satisfying for a prototyping algorithm Figure 11.

According to their authors, BMOG and SuBSENSE methods allow a calculation close to real time thus they do not appear on the comparison chart. We do not consider the DP-GMM method that has been implemented on a GPU.

The methods T2-FGMM are fast, more than our different variants, which have similar complexities (approximately equal slopes). Our method is indeed not economic for calculations, especially for intervals. And it becomes particularly expensive when integrating spatio-temporal fusion, so we need to optimize them in the future.

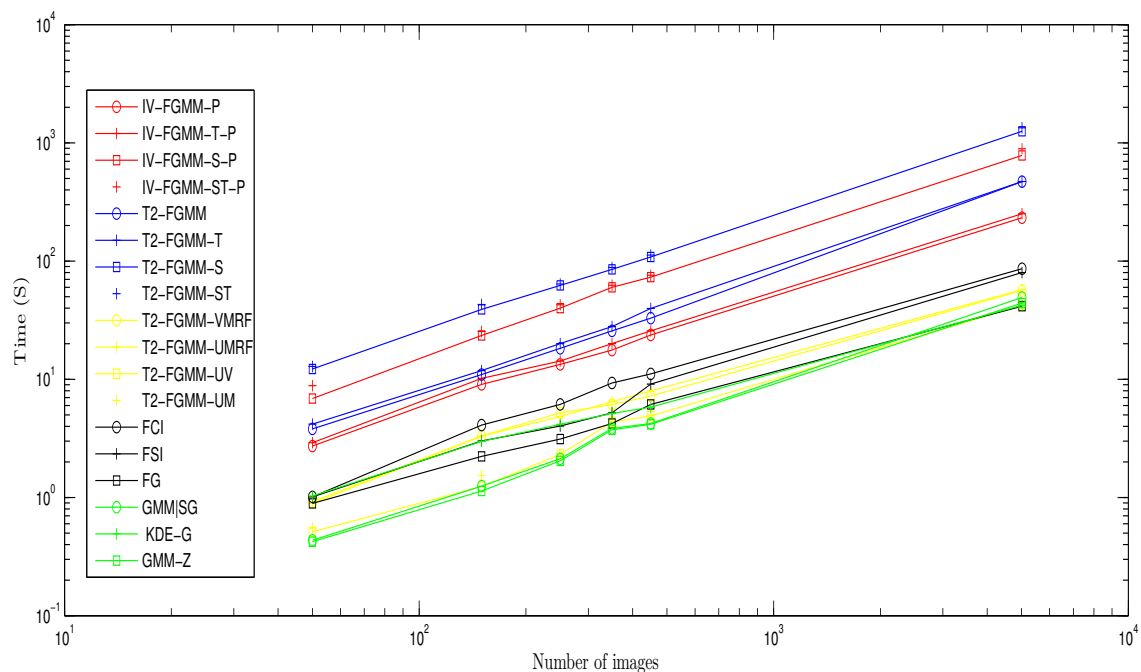


Figure 11. Calculation time according to the number of images in the video sequence.

7. Conclusions

In this work, we proposed an original unsupervised method of background subtraction, based on a type-2 fuzzy Gaussian mixture, particularly robust to the dynamic changes of the background. The method proceeds per pixel, first computing and then aggregating the responses of different complementary fuzzy classifiers, each of them using a piece of information from the pixel: its color, the classes of its neighborhood, and its pixel class in previous frames.

The use of interval-valued fuzzy sets, and the way the fuzziness is preserved all along the classification process, tend to limit the errors coming from the model and from the estimation of its parameters. The method is so particularly devoted to videos with dynamic background, lighting changes or noisy.

We compared the proposed method to powerful recent algorithms of the literature. We focused on the “dynamic background” catalog of the benchmark database [81] from the site [81]. The method proved to be very efficient compared to most of the tested methods. This attests its relevance to process dynamic background change in videos.

Future work will consist in enriching the proposed fusion process with new relevant features, especially with textures and colors (and with another color space than *RGB*). Temporal fusion also needs to be improved because its current contribution to the method performance is minor.

Another important future work would be to deal with a more extensive use of fuzzy set: from the initialization to the end of the process, instead of dealing only with the main (decision-making) step.

Regarding the uncertainty handled in the background model, we turned to *IV* fuzzy sets, for their computation simplicity. We could prefer the richer formalism of Pythagorean numbers, and see if the method can then gain in performance.

Finally, recent deep learning techniques are another promising research way. Recently, several authors [40,86] successfully started to use them for background modeling, and for all categories of videos. For example, one could consider integrating uncertainty management into these techniques, using fuzzy sets. This may be a promising association to deal with most of the challenges of real videos in the future.

Author Contributions: Conceptualization, A.D. and P.-A.H.; Methodology, P.-A.H., A.B.; Software, A.D.; Validation, A.D., P.-A.H. and A.B.; Formal Analysis, P.-A.H.; Investigation, A.B.; Writing—Original Draft Preparation, A.D.; Writing—Review and Editing, A.D.; Visualization, A.B.; Supervision, P.-A.H.; Project Administration, A.B., Y.M.

Funding: This research received no external funding.

Acknowledgments: We would like to warmly thank T.Bouwmans for his ongoing support, helpful advice and interest for our work. The experiments were carried out using the CALCULCO computing platform, supported by SCoSI/ULCO (Univ. Littoral) [82]. we would like to thank the authors of the dataset and the publication of the scores of 19 state-of-the-art methods [3,81].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Apewokin, S.; Valentine, B.; Wills, L.; Wills, S.; Gentile, A. Multimodal mean Adaptive Backgrounding for Embedded real-time video surveillance. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–6.
2. Toyama, K.; Krumm, J.; Brumitt, B.; Meyers, B. Wallflower: Principles and Practice of Background Maintenance. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 1, pp. 255–261.
3. Goyette, N.; Jodoin, P.-M.; Porikli, F.; Konrad, J.; Ishwar, P. Changedetection.net: A new change detection benchmark dataset. In Proceedings of the IEEE Workshop on Change Detection (CDW-2012) at CVPR-2012, Providence, RI, USA, 16–21 June 2012.
4. Bouwmans, T.; Zahzah, E.H. Robust PCA via Principal Component Pursuit: A Review for a Comparative Evaluation in Video Surveillance. *Comput. Vis. Image Underst.* **2014**, *122*, 22–34. [[CrossRef](#)]

5. Amato, A.; Huerta, I.; Mozerov, M.G.; Roca, F.X.; Gonzalez, J. Moving Cast Shadows Detection Methods for Video Surveillance applications. In *Wide Area Surveillance*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 23–47.
6. François, A.R.; Medioni, G.G. Adaptive Color Background Modeling for Real-Time Segmentation of Video Streams. In Proceedings of the International Conference on Imaging Science, Systems, and Technology, Las Vegas, NV, USA, 28 June–1 July 1999; Volume 1, pp. 227–232.
7. Guerra, W.I.; García-Reyes, E. A Novel Approach to Robust Background Subtraction. In Proceedings of the 14th Iberoamerican Conference on Pattern Recognition, Jalisco, Mexico, 15–18 November 2009; pp. 69–76.
8. Piccardi, M. Background Subtraction Techniques: A review. In Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics, The Hague, The Netherlands, 10–13 October 2004; Volume 4, pp. 3099–3104.
9. Elhabian, S.Y.; El-Sayed, K.M.; Ahmed, S.H. Moving Object Detection in Spatial Domain using Background Removal Techniques-State-of-Art. *Recent Pat. Comput. Sci.* **2008**, *1*, 32–54. [[CrossRef](#)]
10. Sobral, A.; Vacavant, A. A Comprehensive review of Background Subtraction Algorithms Evaluated with Synthetic and Real Videos. *Comput. Vis. Image Underst.* **2014**, *122*, 4–21. [[CrossRef](#)]
11. Wren, C.R.; Azarbayejani, A.; Darrell, T.; Pentland, A.P. Pfunder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 780–785. [[CrossRef](#)]
12. Stauffer, C.; Grimson, W.E.L. Adaptive Background Mixture Models for Real-Time Tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 2, pp. 246–252.
13. Bouwmans, T.; El Baf, F. Modeling of Dynamic Backgrounds by Type-2 Fuzzy Gaussian Mixture Models. *MASAUM J. Basic Appl. Sci.* **2010**, *1*, 265–276.
14. Shimada, A.; Nagahara, H.; Taniguchi, R. Background Modeling Based on Bidirectional Analysis. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1979–1986.
15. Bouwmans, T.; El Baf, F.; Vachon, B. Background Modeling using Mixture of Gaussians for Foreground Detection—a Survey. *Recent Pat. Comput. Sci.* **2008**, *1*, 219–237. [[CrossRef](#)]
16. Oliver, N.; Rosario, B.; Pentland, A. A Bayesian computer Vision System for Modeling Human Interactions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 831–843. [[CrossRef](#)]
17. Lin, H.; Liu, T.; Chuang, J. Learning a Scene Background Model via Classification. *IEEE Trans. Signal Process.* **2009**, *57*, 1641–1654.
18. Wang, J.; Bebis, G.; Nicolescu, M.; Nicolescu, M.; Miller, M. Improving Target Detection by Coupling it with Tracking. *Mach. Vis. Appl.* **2008**, *20*, 205–223. [[CrossRef](#)]
19. Tavakkoli, A.; Nicolescu, M.; Bebis, G. A Novelty Detection Approach for Foreground Region Detection in Videos with Quasi-stationary Backgrounds. In Proceedings of the International Symposium on Visual Computing (ISVC'06), Lake Tahoe, NV, USA, 6–8 November 2006; pp. 40–49.
20. Tavakkoli, A.; Nicolescu, M.; Bebis, G.; Nicolescu, M. A Support Vector Data Description Approach for Background Modeling in Videos with Quasi-Stationary Backgrounds. *Int. J. Artif. Intell. Tools* **2008**, *17*, 635–658. [[CrossRef](#)]
21. Varadarajan, S.; Miller, P.; Zhou, H. Spatial Mixture of Gaussians for Dynamic Background Modelling. In Proceedings of the 2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance, Krakow, Poland, 27–30 August 2013; pp. 63–68. [[CrossRef](#)]
22. Martins, I.; Carvalho, P.; Corte-Real, L.; Alba-Castro, J.L. BMOG: Boosted Gaussian Mixture Model with Controlled Complexity. In Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis, Faro, Portugal, 20–23 June 2017; pp. 50–57.
23. Chen, Y.; Wang, J.; Lu, H. Learning Sharable Models for Robust Background Subtraction. In Proceedings of the 2015 IEEE International Conference on Multimedia and Expo (ICME), Turin, Italy, 29 June–3 July 2015; pp. 1–6.
24. Elgammal, A.; Harwood, D.; Davis, L. Non-parametric Model for Background Subtraction. In Proceedings of the European Conference on Computer Vision, Dublin, Ireland, 26 June–1 July 2000; pp. 751–767.
25. Haines, T.S.; Xiang, T. Background subtraction with dirichletprocess mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 670–683. [[CrossRef](#)] [[PubMed](#)]
26. Horn, B.K.; Schunck, B.G. Determining Optical Flow. *Artif. intell.* **1981**, *17*, 185–203. [[CrossRef](#)]

27. Lucas, B.D.; Kanade, T. An Iterative Image Registration Technique with an Application to Stereo Vision. In Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, BC, Canada, 24–28 August 1981.
28. Farneback, G. Two-frame Motion Estimation Based on Polynomial Expansion. In Proceedings of the 13th Scandinavian Conference, Halmstad, Sweden, 29 June–2 July 2003; pp. 363–370.
29. Chauhan, A.K.; Krishan, P. Moving Object Tracking using Gaussian Mixture Model and Optical Flow. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2013**, *3*, 243–246.
30. Chen, M.; Yang, Q.; Li, Q.; Wang, G.; Yang, M.H. Spatiotemporal Background Subtraction using Minimum Spanning Tree and Optical Flow. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 521–534.
31. Bao, L.; Yang, Q.; Jin, H. Fast Edge-preserving Patchmatch for Large Displacement Optical Flow. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3534–3541.
32. Javed, S.; Mahmood, A.; Bouwmans, T.; Jung, S.K. Background-Foreground Modeling Based on spatio-temporal Sparse Subspace Clustering. *IEEE Trans. Image Process.* **2017**, *26*, 5840–5854. [[CrossRef](#)] [[PubMed](#)]
33. St-Charles, P.L.; Bilodeau, G.A. Improving Background Subtraction using Local Binary Similarity Patterns. In Proceedings of the 2014 IEEE Winter Conference on Applications of Computer Vision (WACV), Steamboat Springs, CO, USA, 24–26 March 2014; pp. 509–515.
34. St-Charles, P.L.; Bilodeau, G.A.; Bergevin, R. Subsense: A Universal Change Detection Method with Local Adaptive Sensitivity. *IEEE Trans. Image Process.* **2015**, *24*, 359–373. [[CrossRef](#)] [[PubMed](#)]
35. Varghese, A.; Sreelekha, G. Sample-based Integrated Background Subtraction and Shadow Detection. *IPSI Trans. Comput. Vis. Appl.* **2017**, *9*, 25. [[CrossRef](#)]
36. Bouwmans, T.; Sobral, A.; Javed, S.; Jung, S.K.; Zahzah, E.-H. Decomposition into Low-rank plus Additive Matrices for Background/Foreground Separation: A Review for a Comparative Evaluation with a Large-Scale Dataset. *Comput. Sci. Rev.* **2017**, *23*, 1–71. [[CrossRef](#)]
37. Guyon, C.; Bouwmans, T.; Bouwmans, E.Z. Moving Object Detection via Robust Low Rank Matrix Decomposition with IRLS scheme. In Proceedings of the Proceedings of International Symposium on Visual Computing (ISVC'12), Crete, Greece, 16–18 July 2012; pp. 665–674.
38. Javed, S.; Bouwmans, T.; Jung, K.S. Stochastic Decomposition into Low Rank and Sparse Tensor for Robust Background Subtraction. In Proceedings of the 6th International Conference on Imaging for Crime Prevention and Detection (ICDP-15), London, UK, 15–17 July 2015.
39. Sobral, A.; Baker, C.G.; Bouwmans, T.; Zahzah, E. Incremental and Multi-feature Tensor Subspace Learning applied for Background Modeling and Subtraction. In Proceedings of the Proceedings of International Conference on Image Analysis and Recognition (ICIAR'14), Vilamoura, Portugal, 22–24 October 2014.
40. Braham, M.; Van Droogenbroeck, M. Deep Background Subtraction with Scene-specific Convolutional Neural Networks. In Proceedings of the 2016 International Conference on Systems, Signals and Image Processing (IWSSIP), Bratislava, Slovakia, 23–25 May 2016; pp. 1–4.
41. Babae, M.; Dinh, D.T.; Rigoll, G. A deep convolutional neural network for video sequence background subtraction. *Pattern Recognit.* **2018**, *76*, 635–649. [[CrossRef](#)]
42. Zhang, Y.; Li, X.; Zhang, Z.; Wu, F.; Zhao, L. Deep learning driven blockwise moving object detection with binary scene modeling. *Neurocomputing* **2015**, *168*, 454–463. [[CrossRef](#)]
43. Xu, P.; Ye, M.; Li, X.; Liu, Q.; Yang, Y.; Ding, J. Dynamic background learning through deep auto-encoder networks. In Proceedings of the 22nd ACM international conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 107–116.
44. Brunetti, A.; Buongiorno, D.; Trotta, G.F.; Bevilacqua, V. Computer Vision and Deep Learning Techniques for Pedestrian Detection and Tracking: A Survey. *Neurocomputing* **2018**, *300*, 17–33. [[CrossRef](#)]
45. Yang, L.; Tian, S.; Yu, L.; Ye, F.; Qian, J.; Qian, Y. Deep Learning for Extracting Water Body from Landsat Imagery. *Int. J. Innov. Comput. Inf. Control* **2015**, *11*, 1913–1929.
46. Dubois, D.; Prade, H. Interval-valued Fuzzy Sets, Possibility Theory and Imprecise Probability. In Proceedings of the 4th Conference of the European Society for Fuzzy Logic and Technology, Barcelona, Spain, 7–9 September 2005; pp. 314–319.
47. Sugeno, M. Exploring Categories of Uncertainty—Toward Structure of Uncertainty. Presented at the Séminaire Donnees et APprentissage Artificiel, Valenciennes, France, 30 May 2013.

48. Zeng, J.; Xie, L.; Liu, Z.Q. Type-2 Fuzzy Gaussian Mixture Models. *Pattern Recognit.* **2008**, *41*, 3636–3643. [[CrossRef](#)]
49. Chiranjeevi, P.; Sengupta, S. New Fuzzy Texture Features for Robust Detection of Moving Objects. *IEEE Signal Process. Lett.* **2012**, *19*, 603–606. [[CrossRef](#)]
50. Chiranjeevi, P.; Sengupta, S. Detection of Moving Objects using Multi-channel Kernel Fuzzy Correlogram Based Background Subtraction. *IEEE Trans. Cybern.* **2014**, *44*, 870–881. [[CrossRef](#)] [[PubMed](#)]
51. Chiranjeevi, P.; Sengupta, S. Neighborhood Supported Model Level Fuzzy Aggregation for Moving Object Segmentation. *IEEE Trans. Image Process.* **2014**, *23*, 645–657. [[CrossRef](#)] [[PubMed](#)]
52. Pojala, C.; Sengupta, S. Detection of Moving Objects using Fuzzy Correlogram Based Background Subtraction. In Proceedings of the 2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, Malaysia, 16–18 November 2011.
53. Chiranjeevi, P.; Sengupta, S. Interval-Valued Model Level Fuzzy Aggregation-Based Background Subtraction. *IEEE Trans. Cybern.* **2017**, *47*, 2544–2555. [[CrossRef](#)] [[PubMed](#)]
54. Chiranjeevi, P.; Sengupta, S. Rough-set-theoretic Fuzzy Cues-based Object Tracking Under Improved Particle Filter Framework. *IEEE Trans. Fuzzy Syst.* **2016**, *24*, 695–707. [[CrossRef](#)]
55. Bouwmans, T. Background Subtraction for Visual Surveillance: A Fuzzy Approach. *Handb. Soft Comput. Video Surveill.* **2012**, *5*, 103–138.
56. El Baf, F.; Bouwmans, T.; Vachon, B. Type-2 Fuzzy Mixture of Gaussian Model: Application to Background, Modeling. In Proceedings of the International Symposium on Visual Computing, Las Vegas, NV, USA, 1–3 December 2008; pp. 772–781.
57. Baf, F.E.; Bouwmans, T.; Vachon, B. Fuzzy Statistical Modeling of Dynamic Backgrounds for Moving Object Detection in Infrared Videos. In Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Miami, FL, USA, 20–25 June 2009; pp. 60–65. [[CrossRef](#)]
58. Sigari, M.H.; Mozayani, N.; Pourreza, H. Fuzzy Running Average and Fuzzy Background Subtraction: Concepts and Application. *Int. J. Comput. Sci. Netw. Secur.* **2008**, *8*, 138–143.
59. Sigari, M. Fuzzy Background Modeling/Subtraction and its Application in Vehicle Detection. In Proceedings of the World Congress on Engineering and Computer Science (WCECS 2008), San Francisco, CA, USA, 22–24 October 2008.
60. Rosell-Ortega, J.; Garcia-Andreu, G.; Rodas-Jorda, A.; Atienza-Vanacloig, V. A Combined Self-configuring Method for Object Tracking in Colour Video. In Proceedings of the 2010 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey, 23–26 August 2010; pp. 2081–2084.
61. Zhang, H.; Xu, D. Fusing Color and Texture Features for Background Model. In Proceedings of the 3th International Conference on Fuzzy Systems and Knowledge Discovery, Xian, China, 24–28 September 2006; pp. 887–893.
62. El Baf, F.; Bouwmans, T.; Vachon, B. Fuzzy Integral for Moving Object Detection. In Proceedings of the 2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 1729–1736.
63. Kim, W.; Kim, C. Background Subtraction for Dynamic Texture Scenes using Fuzzy Color Histograms. *IEEE Signal Process. Lett.* **2012**, *19*, 127–130. [[CrossRef](#)]
64. Gutti, V.; Shankar, C. A Novel Approach to Background Subtraction using Fuzzy Color Histogram. *J. Adv. Eng. Technol.* **2014**, *3*, 231–239.
65. Manjula, D.; Sivabalakrishnan, M. Adaptive Background Subtraction in Dynamic Environments using Fuzzy Logic. *Int. J. Video Image Process. Netw. Secur.* **2010**, *10*, 13–16.
66. Darwich, A.; Hébert, P.A.; Mohanna, Y.; Bigand, A. Background Subtraction under Uncertainty using a Type-2 Fuzzy Set Gaussian Mixture Model. In Proceedings of the The Fourth International Conference on Computer Science, Computer Engineering, and Education Technologies (CSCEET2017), Beirut, Lebanon, 26–28 April 2017; pp. 1–6.
67. McLachlan, G.J.; Basford, K.E. Mixture Models. Inference and Applications to Clustering. In *Statistics: Textbooks and Monographs*; Marcel Dekker: New York, NY, USA, 1988; Volume 1.
68. Bigand, A.; Colot, O. Fuzzy Filter Based on Interval-Valued Fuzzy Sets for Image Filtering. *Fuzzy Sets Syst.* **2010**, *161*, 96–117. [[CrossRef](#)]
69. Mendel, J.M.; John, R.B. Type-2 Fuzzy Made Simple. *IEEE Trans. Fuzzy Syst.* **2002**, *10*, 117–127. [[CrossRef](#)]
70. Zadeh, L.A. Fuzzy Sets. *Inf. Control* **1965**, *8*, 338–353. [[CrossRef](#)]

71. Zadeh, L.A. The Concept of Linguistic Variable and its Application to Approximate Reasoning. *Inf. Sci.* **1975**, *8*, 199–249. [[CrossRef](#)]
72. Boukezzoula, R.; Galichet, S.; Foulloy, L. Sur les Systèmes Flous de Type-2 en Contrôle! In Proceedings of the 25^{èmes} Rencontres Francophones Sur la Logique Floue et ses Applications (LFA'2016), La Rochelle, France, 15–16 November 2016.
73. Hwang, C.; Rhee, F.C.H. Uncertain Fuzzy Clustering: Interval Type-2 Fuzzy Approach to C-means. *Fuzzy Syst. IEEE Trans.* **2007**, *15*, 107–120. [[CrossRef](#)]
74. Hosseini, R.; Dehmeshki, J.; Barman, S.; Mazinani, M.; Qanadli, S. A Genetic Type-2 Fuzzy Logic System for Pattern Recognition in Computer Aided Detection Systems. In Proceedings of the 2010 IEEE International Conference on Fuzzy Systems (FUZZ), Barcelona, Spain, 18–23 July 2010; pp. 1–7.
75. Karnik, N.N.; Mendel, J.M.; Liang, Q. Type-2 Fuzzy Logic Systems. *Fuzzy Syst. IEEE Trans.* **1999**, *7*, 643–658. [[CrossRef](#)]
76. Hisdal, E. The IF THEN ELSE Statement and Interval-Valued Fuzzy Sets of Higher Type. *Int. J. Man-Mach. Stud.* **1981**, *15*, 385–455. [[CrossRef](#)]
77. Bigand, A.; Colot, O. Membership Function Construction for Interval-Valued Fuzzy Sets with Application to Gaussian Noise Reduction. *Fuzzy Sets Syst.* **2016**, *286*, 66–85. [[CrossRef](#)]
78. Zeng, J.; Liu, Z.Q. Type-2 Fuzzy Sets for Handling Uncertainty in Pattern Recognition. In Proceedings of the 2006 IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, 16–21 July 2006; pp. 1247–1252.
79. Sengupta, A.; Pal, T.K. On Comparing Interval Numbers. *Eur. J. Oper. Res.* **2000**, *127*, 28–43. [[CrossRef](#)]
80. Shimada, A.; Arita, D.; Taniguchi, R.i. Dynamic Control of Adaptive Mixture-of-Gaussians Background Model. In Proceedings of the 2006 IEEE International Conference on Video and Signal Based Surveillance, Sydney, Australia, 22–24 November 2006; pp. 5–5.
81. Wang, Y.; Jodoin, P.-M.; Porikli, F.; Konrad, J.; Benezeth, Y.; Ishwar, P. CDnet 2014: An Expanded Change Detection Benchmark Dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Columbus, OH, USA, 24–27 June 2014.
82. *CALCULCO*, version 2017-05-11; University of the Littoral Opal Coast: Dunkerque, France, 2017.
83. Zivkovic, Z. Improved Adaptive Gaussian Mixture Model for Background Subtraction. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26–26 August 2004; Volume 2, pp. 28–31.
84. Zhao, Z.; Bouwmans, T.; Zhang, X.; Fang, Y. A Fuzzy Background Modeling Approach for Motion Detection in Dynamic Backgrounds. In *Multimedia and signal processing*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 177–185.
85. Sobral, A. BGSLibrary: An OpenCV C++ Background Subtraction Library. In Proceedings of the IX Workshop de Visão Computacional (WVC'2013), Rio de Janeiro, Brazil, 3–5 June 2013.
86. Wang, Y.; Luo, Z.; Jodoin, P.M. Interactive Deep Learning Method for Segmenting Moving Objects. *Pattern Recognit. Lett.* **2017**, *96*, 66–75. [[CrossRef](#)]

