Variational Inference for Background Subtraction in Infrared Imagery

Konstantinos Makantasis[1], Anastasios Doulamis[2], Konstantinos Loupos[3]

[1]Technical University of Crete, University campus, Kounoupidiana, 73100, Chania, Greece
kmakantasis@isc.tuc.gr
[2]National Technical University of Athens, Zografou campus, 15780, Athens, Greece
adoulam@cs.ntua.gr
[3]Institute of Communication and Computer Systems, Athens, Greece,
kloupos@iccs.gr

Abstract. We propose a Gaussian mixture model with fixed but unknown number of components for background subtraction in infrared imagery. Following a Bayesian approach, our method automatically estimates the number of components as well as their parameters, while simultaneously it avoids over/under fitting. The equations for estimating model parameters are analytically derived and thus our method does not require any sampling algorithm that is computationally and memory inefficient. The pixel density estimate is followed by an efficient and highly accurate updating mechanism, which permits our system to be automatically adapted to dynamically changing visual conditions. Experimental results and comparisons with other methods indicate the high potential of the proposed method while keeping computational cost suitable for real-time applications.

1 Introduction

Pixels values of infrared frames correspond to the relative differences in the amount of thermal energy emitted or reflected from objects in the scene. Due to this fact, infrared cameras are equally applicable for both day and night scenarios, while at the same time, compared to visual-optical cameras, are less affected by changing illumination or background texture. Furthermore, infrared imagery eliminates any privacy issues as people being depicted in the scene can not be identified. These features make infrared cameras prime candidate for persistent video surveillance systems.

Although, infrared imagery can alleviate several problems associated with visual-optical videos, it has its own unique challenges such as a) low signal-to-noise ratio (noisy data) and b) almost continuous pixel values that model objects' temperature. Both issues complicate pixel responses modeling and most of conventional computer vision techniques, that successfully used for visual-optical data, can not be applied straightforward on thermal imagery.

For many high-level vision based applications, either they use visual-optical videos [1, 2] or infrared data [3–5], the task of background subtraction constitutes a key component, as this is one of the most common methods for locating moving objects, facilitating search space reduction and visual attention modeling [6–8]. Background subtraction methods applied on visual-optical videos model the color properties of depicted

objects [9,10] and can be classified into three main categories [11]: basic modeling [12], statistical modeling [13, 14] and background estimation [15, 16].

The most used methods are the statistical ones due to their robustness to critical situations. In order to statistically represent the background, a probability distribution is used to model pixel intensities over time. Towards this direction, the work of Stauffer and Grimson [17], is one of the best known approaches. It uses a Gaussian mixture model, with fixed number of components, for a per-pixel density estimate. Similar to this approach, Makantasis *et al.* in [18] propose a Student-t mixture model, taking advantage of Student-t distribution compactness and robustness to noise and outliers. The works of [19, 20] extend the method of [17] by introducing a rule based on a user defined threshold to estimate the number of components. However, this rule is application dependent and not directly derived from the data. All these techniques present the drawback that objects' color properties are highly affected by scene illumination, making the same object to look completely different under different lighting or weather conditions.

Although, thermal imagery can provide a challenging alternative for addressing the aforementioned difficulty, there exist few works for thermal data. The authors of [21,22] exploit contour saliency to extract foreground objects. Initially, they utilize a unimodal background modeling technique to detect regions of interest and then exploit the halo effect of thermal data for extracting foreground objects. However, unimodal background modeling is not usually capable of capturing background dynamics. Baf *et al.* in [11] present a fuzzy statistical method for background subtraction to incorporate uncertainty into the mixture of Gaussians. This method requires a predefined number of components and thus is application dependent. Elguebaly and Bouguila in [23] propose a finite asymmetric generalized Gaussian mixture model for object detection. Again this method requires a predefined maximum number of components, presenting therefore limitations when this technique is applied on uncontrolled environments. Dai *et al.* in [24] propose a method for pedestrian detection and tracking using infrared imagery. This method consists of a background subtraction technique that exploits a two-layer representation (one for foreground and one for background) of infrared frame sequences. However, the assumption made is that the foreground is restricted to moving objects, a consideration which is not sufficient for dynamically changing environments.

One way to handle the aforementioned difficulties is to introduce a background model, the parameters and the structure of which are directly estimated from the data, while at the same time it takes into account the special properties of infrared imagery.

1.1 Our contribution

This work presents background modeling able to provide a per pixel density estimate, taking into account the special characteristics of infrared imagery, such as low signal-to-noise ratio. Our method exploits a Gaussian mixture model with unknown number of components. The advantage of such a model is that its own parameters and structure can be directly estimated from data distribution, allowing dynamic model adaptation to uncontrolled and changing environments.

An important issue in the proposed Gaussian mixture modeling concerns learning the model parameters. In our method, this is addressed using a variational inference framework to associate the functional structure of the model with real data distributions

obtained from the infrared images. Then, the Expectation-Maximization (EM) algorithm is adopted to fit the outcome of variational inference to real measurements. Updating procedures are incorporated to allow dynamic model adaptation to the forthcoming infrared data. Our updating method avoids of using heuristics by considering existing knowledge accumulated from previous data distributions and then it compensates this knowledge with current measurements. Our overall strategy exploits a Bayesian framework to estimate all the parameters of the mixture model and thus avoids over/under fitting issues. To compensate computational challenges arising from the non a priori known nature of the mixture model, we utilize conjugate priors and thus we derive analytical equations for model estimation. In this way, we avoid the need of any sampling method, which are computationally and memory inefficient.

2 Gaussian mixture modeling

In this section we formulate the Bayesian framework adopted in this paper to estimate all the parameters of the proposed mixture model. In section 2.1 we briefly describe the basic theory behind Gaussian mixtures, while in section 2.2 we describe the introduction of conjugate priors that assist us in yielding analytical model estimations.

2.1 Model fundamentals

The Gaussian mixture distribution can be seen as a linear superposition of Gaussian functional components,

$$p(x|\boldsymbol{\varpi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \sum_{k=1}^{K} \varpi_k \mathcal{N}(x|\mu_k, \tau_k^{-1}) \tag{1}$$

where the parameters $\{\varpi_k\}_{k=1}^K$ must satisfy $0 \leq \varpi_k \leq 1$ for every k and $\sum_{k=1}^K \varpi_k = 1$ and K is the number of Gaussian components. In the proposed mixture modeling, variable K can take any natural value up to infinity. However, it is highly recommended to set the upper bound for K less than the cardinality of the dataset, i.e. the number of observed samples. By introducing a K-dimensional binary latent variable $\boldsymbol{z}$, such as $\sum_{k=1}^K z_k = 1$ and $p(z_k = 1) = \varpi_k$, the distribution $p(x)$ can be defined in terms of a marginal distribution $p(\boldsymbol{z})$ and a conditional distribution $p(x|\boldsymbol{z})$ as follows

$$p(x|\boldsymbol{\varpi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \sum_{\boldsymbol{z}} p(\boldsymbol{z}|\boldsymbol{\varpi}) p(x|\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\tau}) \tag{2}$$

where $p(\boldsymbol{z}|\boldsymbol{\varpi})$ and and $p(x|\boldsymbol{z})$ are in the form of

$$p(\boldsymbol{z}|\boldsymbol{\varpi}) = \prod_{k=1}^{K} \varpi_k^{z_k} \quad \text{and} \quad p(x|\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \prod_{k=1}^{K} \mathcal{N}(x|\mu_k, \tau_k^{-1})^{z_k} \tag{3}$$

where $\boldsymbol{\mu} = \{\mu_k\}_{k=1}^K$ and $\boldsymbol{\tau} = \{\tau_k\}_{k=1}^K$, correspond to the mean values and precisions of Gaussian components. By introducing latent variables and transforming the Gaussian

mixture distribution into the form of (2), we are able to exploit the EM algorithm for fitting our model to the observed data, as shown in Section 4.

If we have in our disposal a set $\boldsymbol{X} = \{x_1, ..., x_N\}$ of observed data we will also have a set $\boldsymbol{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_N\}$ of latent variables. Each $\boldsymbol{z}_n$ will be a K-dimensional binary vector, such as $\sum_{k=1}^{K} z_{nk} = 1$, and, in order to take into consideration the whole dataset of N samples, the distributions of (3) will be transformed to

$$p(\boldsymbol{Z}|\boldsymbol{\varpi}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \varpi_k^{z_{nk}} \tag{4}$$

$$p(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\tau}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \mathcal{N}(x_n|\mu_k, \tau_k^{-1})^{z_{nk}} \tag{5}$$

2.2 Conjugate priors

To avoid computational problems in estimating the parameters and the structure of the proposed Gaussian mixture, we introduce conjugate priors, over the model parameters $\boldsymbol{\mu}$, $\boldsymbol{\tau}$ and $\boldsymbol{\varpi}$, that allow us to yield analytical solutions. Introduction of priors implies the use of a Bayesian framework for the analysis.

Let us denote as $\boldsymbol{Y} = \{\boldsymbol{Z}, \boldsymbol{\varpi}, \boldsymbol{\mu}, \boldsymbol{\tau}\}$ the set which contains all model latent variables and parameters and as $q(\boldsymbol{Y})$ its distribution. Then, our goal is to estimate $q(\boldsymbol{Y})$ which maximizes model evidence $p(\boldsymbol{X})$.

$$q(\boldsymbol{Y}) : \max \ln p(\boldsymbol{X}) \tag{6}$$

where in (6) we used the logarithm of $p(\boldsymbol{X})$ for calculus purposes. For maximizing (6) we need to define the distribution over $\boldsymbol{Y}$, that is, $p(\boldsymbol{Z}|\boldsymbol{\varpi})$ from (4), $p(\boldsymbol{\varpi})$ and $p(\boldsymbol{\mu}, \boldsymbol{\tau})$.

Due to the fact that $p(\boldsymbol{Z}|\boldsymbol{\varpi})$ is a Multinomial distribution, its conjugate prior is a Dirichlet distribution over the mixing coefficients $\boldsymbol{\varpi}$

$$p(\boldsymbol{\varpi}) = \frac{\Gamma(K\lambda_0)}{\Gamma(\lambda_0)^K} \prod_{k=1}^{K} \varpi_k^{\lambda_0 - 1} \tag{7}$$

where $\Gamma(\cdot)$ is the Gamma function. Parameter λ_0 has a physical interpretation; the smaller the value of this parameter is, the larger is the influence of the data rather than the prior on the posterior distribution $p(\boldsymbol{Z}|\boldsymbol{\varpi})$. In order to introduce uninformative priors and not prefer a specific component against the other, we choose to use a single parameter λ_0 for the Dirichlet distribution, instead of a vector with different values for each mixing coefficient.

Similarly, the conjugate prior of (3) takes the form of a Gaussian-Gamma distribution, because both $\boldsymbol{\mu}$ and $\boldsymbol{\tau}$ are unknown. Subsequently, the joint distribution of parameters $\boldsymbol{\mu}$ and $\boldsymbol{\tau}$ can be modeled as

$$p(\boldsymbol{\mu}, \boldsymbol{\tau}) = p(\boldsymbol{\mu}|\boldsymbol{\tau})p(\boldsymbol{\tau}) = \prod_{k=1}^{K} \mathcal{N}(\mu_k|m_0, (\beta_0\tau_k)^{-1}) Gam(\tau_k|a_0, b_0) \tag{8}$$

where $Gam(\cdot)$ denotes the Gamma distribution. In order to not express any specific preference about the form of the Gaussian components through the introduction of priors, we use uninformative priors by setting the values of hyperparameters m_0, β_0, a_0 and b_0 to appropriate values as shown in Section 4.

Having defined the parametric form of observed data, latent variables and parameters distributions, our goal is to approximate the posterior distribution $p(\boldsymbol{Y}|\boldsymbol{X})$ and the model evidence $p(\boldsymbol{X})$, where $\boldsymbol{Y} = \{\boldsymbol{Z}, \boldsymbol{\varpi}, \boldsymbol{\mu}, \boldsymbol{\tau}\}$ is the set with distribution $q(\boldsymbol{Y})$, which contains all model latent variables and parameters. Based on the Bayes rule, the logarithm of distribution $p(\boldsymbol{X})$ can be expressed as

$$\ln p(\boldsymbol{X}) = \int q(\boldsymbol{Y}) \ln \frac{p(\boldsymbol{X}, \boldsymbol{Y})}{q(\boldsymbol{Y})} d\boldsymbol{Y} - \int q(\boldsymbol{Y}) \ln \frac{p(\boldsymbol{Y}|\boldsymbol{X})}{q(\boldsymbol{Y})} d\boldsymbol{Y} \tag{9a}$$

$$= \mathcal{L}(q) + KL(q||p) \tag{9b}$$

where $KL(q||p)$ is the Kullback-Leibler divergence between $q(\boldsymbol{Y})$ and $p(\boldsymbol{Y}|\boldsymbol{X})$ distributions and $\mathcal{L}(q)$ is the lower bound of $\ln p(\boldsymbol{X})$. Since $KL(q||p)$ is a non negative quantity, equals to zero only if $q(\boldsymbol{Y})$ is equal to $p(\boldsymbol{Y}|\boldsymbol{X})$, maximization of $\ln p(\boldsymbol{X})$ is equivalent to minimizing of $KL(q||p)$. For minimizing $KL(q||p)$ and estimating $p(\boldsymbol{X})$ we exploit the EM algorithm, as shown in Section 4.

By making the assumption, based on variational inference, that the distribution $q(\boldsymbol{Y})$ can be factorized over M disjoint sets such as $q(\boldsymbol{Y}) = \prod_{i=1}^{M} q_i(\boldsymbol{Y}_i)$, as shown in [25], the optimal solution $q_j^*(Y_j)$ corresponds to the minimization of $KL(q||p)$ is

$$\ln q_j^*(\boldsymbol{Y}_j) = \mathbb{E}_{i \neq j}[\ln p(\boldsymbol{X}, \boldsymbol{Y})] + \mathcal{C} \tag{10}$$

where $\mathbb{E}_{i \neq j}[\ln p(\boldsymbol{X}, \boldsymbol{Y})]$ is the expectation of the joint distribution over all variables $\boldsymbol{Y}_j$ for $j \neq i$ and $\mathcal{C}$ is a constant. $P(\boldsymbol{X}, \boldsymbol{Y})$ is modeled through (11). In the following, we present the analytical solution for the optimal distributions $q_j^*(Y_j)$ for model parameters and latent variables, i.e. the optimal distributions $q^*(\boldsymbol{Z})$, $q^*(\boldsymbol{\varpi})$, $q^*(\boldsymbol{\tau})$ and $q^*(\boldsymbol{\mu}|\boldsymbol{\tau})$.

3 Optimal model parameter distributions

According to (4), (5), (7) and (8), the joint distribution of all random variables can be factorized as follows

$$p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\varpi}, \boldsymbol{\mu}, \boldsymbol{\tau}) = p(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\tau}) p(\boldsymbol{Z}|\boldsymbol{\varpi}) p(\boldsymbol{\varpi}) p(\boldsymbol{\mu}|\boldsymbol{\tau}) p(\boldsymbol{\tau}) \tag{11}$$

where $\boldsymbol{X}$ are the observed variables.

Using (10) and the factorized form of (11) the distribution of the optimized factor $q^*(\boldsymbol{Z})$ is given by a Multinomial distribution of the form

$$q^*(\boldsymbol{Z}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left(\frac{\rho_{nk}}{\sum_{j=1}^{K} \rho_{nj}} \right)^{z_{nk}} = \prod_{n=1}^{N} \prod_{k=1}^{K} r_{nk}^{z_{nk}} \tag{12}$$

as ρ_{nk} we have denote the quantity

$$\rho_{nk} = \exp\left(\mathbb{E}\big[\ln \varpi_k\big] + \frac{1}{2}\mathbb{E}\big[\ln \tau_k\big] - \frac{1}{2}\ln 2\pi - \frac{1}{2}\mathbb{E}_{\boldsymbol{\mu},\boldsymbol{\tau}}\big[(x_n - \mu_k)^2 \tau_k\big] \right) \tag{13}$$

The expected value $\mathbb{E}[z_{nk}]$ of $q^*(\boldsymbol{Z})$ is equal to r_{nk}. Using (11) and (10) the distribution of the optimized factor $q^*(\boldsymbol{\varpi})$ is given a Dirichlet distribution of the form

$$q^*(\boldsymbol{\varpi}) = \frac{\Gamma(\sum_{i=1}^{K}\lambda_i)}{\prod_{j=1}^{K}\Gamma(\lambda_j)}\prod_{k=1}^{K}\varpi_k^{\lambda_k-1} \tag{14}$$

λ_k is equal to $N_k + \lambda_0$, where $N_k = \sum_{n=1}^{N} r_{nk}$ represents the proportion of data that belong to the k-th component. Similarly, the distribution of the optimized factor $q^*(\mu_k, \tau_k)$ is given by a Gaussian distribution of the form

$$q^*(\mu_k|\tau_k) = \mathcal{N}(\mu_k|m_k, (\beta_k\tau)^{-1}) \tag{15}$$

where the parameters m_k and β_k are given by

$$\beta_k = \beta_0 + N_k \tag{16a}$$

$$m_k = \frac{1}{\beta_k}\Big(\beta_0 m_0 + N_k\bar{x}_k\Big) \tag{16b}$$

where $\bar{x}_k$ is equal to $\frac{1}{N_k}\sum_{n=1}^{N} r_{nk}x_n$ represents the centroid of the data that belong to the k-th component. After the estimation of $q^*(\mu_k|\tau_k)$, distribution of the optimized factor $q^*(\tau_k)$ is given by a Gamma distribution of the following form

$$q^*(\tau_k) = Gam(\tau_k|a_k, b_k) \tag{17}$$

while the parameters a_k and b_k are given by the following relations

$$a_k = a_0 + \frac{N_k}{2} \tag{18a}$$

$$b_k = b_0 + \frac{1}{2}\left(N_k\sigma_k + \frac{\beta_0 N_k}{\beta_0 + N_k}\big(\bar{x}_k - m_0\big)^2\right) \tag{18b}$$

where $\sigma_k = \frac{1}{N_k}\sum_{n=1}^{N}(x_n - \bar{x}_k)^2$.

4 Distribution parameters optimization

After the approximation of random variables distributions, we will use the EM algorithm in order to find optimal values for model parameters, i.e. maximize (9). In order to use the EM algorithm, we have to initialize priors hyperparameters λ_0, a_0, b_0, m_0 and β_0 and the model parameters ϖ_k, μ_k, τ_k, β_k, a_k, b_k and λ_k (see Section 3).

The parameter λ_0 can be interpreted as the effective prior number of observations associated with each component. We introduce an uninformative prior for $\boldsymbol{\varpi}$ by setting the parameter λ_0 equal to N/K (the same number of observations is associated to each component). Parameters a_0 and b_0 were set to the value of 10^{-3}. When the values for a_0 and b_0 are close to zero the results of updating equations (18a) and (18b) are primarily affected by the data and not by the prior. The mean values of the components are described by conditional Normal distribution with means m_0 and precisions $\beta_0\tau_k$.

We introduce an uninformative prior by setting the value for m_0 to the mean of the observed data and $\beta_0 = \frac{b_0}{a_0 v_0}$, where v_0 is the variance of the observed data.

The convergence of EM algorithm is facilitated by initializing the parameters ϖ_k, μ_k, τ_k and β_k using the k-means. To utilize k-means, the number of clusters, i.e. the Gaussian components, should be a priori known. Although the number of components is unknown, it should be less or equal to the number of observed data. If we have no clue about the number of classes K_{max} we can set it to equal N. If we denote as $\hat{N}_k$ the number of observation that belong to k-th cluster, then we can set the value of parameter μ_k to equal the centroid of k-th cluster, the parameter ϖ_k to equal the proportion of observations for the k-th cluster, the parameter τ_k to equal $\hat{v}_k^{-1}$, where v_k stands for the variance of the k-th cluster and the parameter β_k to equal $\hat{N}_k^{-1}$. Then, we can exploit the formula for the expected value of a Gamma distribution to initialize the parameters a_k and b_k to values τ_k and one respectively. Finally, the initialization of ϖ_k allows us to initialize the parameter λ_k, which can be interpreted as the effective number of observations associated with each Gaussian component, to the value $N\varpi_k$.

After the initialization of model parameters and priors hyperparameters, the EM algorithm can be used to minimize $KL(q||p)$ of (9). During the E step, r_{nk} is calculated given the initial/current values of all the parameters of the model. Using (12) r_{nk} is

$$r_{nk} \propto \tilde{\varpi_k} \tilde{\tau_k}^{1/2} \exp\left(-\frac{a_k}{2b_k}(x_n - m_k)^2 - \frac{1}{2\beta_k} \right) \tag{19}$$

Due to the fact that $q^*(\boldsymbol{\varpi})$ is a Dirichlet distribution and $q^*(\tau_k)$ is a Gamma distribution, $\tilde{\varpi_k}$ and $\tilde{\tau_k}$ will be given by

$$\ln \tilde{\varpi_k} \equiv \mathbb{E}\big[\ln \varpi_k\big] = \Psi(\lambda_k) - \Psi\left(\sum_{k=1}^{K} \lambda_k\right) \tag{20a}$$

$$\ln \tilde{\tau_k} \equiv \mathbb{E}\big[\ln \tau_k\big] = \Psi(a_k) - \ln b_k \tag{20b}$$

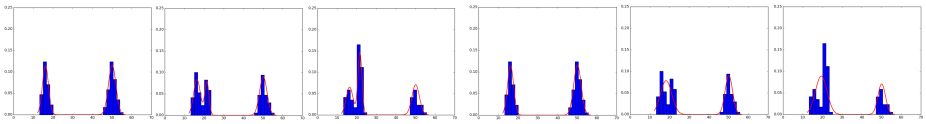
where $\Psi(\cdot)$ is the digamma function.

During the M step, we keep fixed the value of r_{nk} (calculated during the E step), and we re-calculate the values for model parameters using (14), (16) and (18). The steps E and M are repeated sequentially untill the values for model parameters are not changing anymore. As shown in [26] convergence of EM algorithm is guaranteed because bound is convex with respect to each of the factors $q(\boldsymbol{Z})$, $q(\boldsymbol{\varpi})$, $q(\boldsymbol{\mu}|\boldsymbol{\tau})$ and $q(\boldsymbol{\tau})$.

During model training the mixing coefficient for some of the components takes value very close to zero. Components with mixing coefficient less than $1/N$ are removed (we require each component to model at least one observed sample) and thus after training the model has automatically determined the right number of components.

5 Online updating mechanism

Having described how our model fits to N observed data, in this section we present the mechanism that permits our model to automatically adapt to new visual conditions. Contrary to [19] where heuristic rules are used during the adaptation of the model, we exploit statistics based on the observed data.



(a) Proposed method (b) Model presented in [19]

Fig. 1. Updating to new observed data.

Let us denote as x_{new} a new observed sample. Then, there are two cases; either this sample is successfully modeled by our trained model, or not. To estimate if a new sample is successfully modeled, we find its closest component the Mahalanobis distance, since this is reliable distance measure between a point and a distribution.

The closest component c to the new sample is the one that presents the minimum Mahalanobis distance D_k

$$c = \arg\min_k D_k = \arg\min_k \sqrt{(x_{new} - \mu_k)^2 \tau_k} \tag{21}$$

The probability of the new sample to belong to c is

$$p(x_{new}|\mu_c, \tau_c) = \mathcal{N}(x_{new}|\mu_c, \tau_c^{-1}) \tag{22}$$

where μ_c and τ_c stand for the closest component mean value and precision respectively.

Let us denote as Ω the initially observed dataset. Then, we can assume that the probability to observe the new sample x_{new} is given by

$$p(x_{new}|e) = \frac{N_e}{N} \mathcal{U}(x_{new}|x_{new} - e, x_{new} + e) \tag{23}$$

where $N_e = \left|\{x_i \in \Omega : x_{new} - e \leq x_i \leq x_{new} + e\}\right|$ and $\mathcal{U}(x_{new}|x_{new} - e, x_{new} + e)$ is a Uniform distribution with lower and upper bounds to equal $x_{new} - e$ and $x_{new} + e$ respectively. Equation (23) suggests that the probability to observe x_{new} is related to the proportion of data that have already been observed around x_{new}. By increasing the neighborhood around x_{new}, i.e. increasing the value of e, the quantity $\mathcal{U}(x_{new}|x_{new} - e, x_{new} + e)$ is decreasing, while the value of N_e is increasing. Upon the arrival of a new sample, we can estimate the optimal range ϵ around it that maximizes (23) as

$$\epsilon = \arg\max_e p(x_{new}|e) \tag{24}$$

Then, if $p(x_{new}|\mu_c, \tau_c) \geq p(x_{new}|\epsilon)$ the new observed sample can sufficiently represented by our model. Otherwise, a new Gaussian component must be created.

For model updating, we need to define the binary variable o, called ownership and associated with the Gaussian components, as

$$o_k = \begin{cases} 1, & \text{if } k = c \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

where c represents the index of the closest component and k is the index of k-th component. When the new observed sample is successfully modeled, the parameters for the Gaussian components are updated using the *following the leader* [27] approach

$$\varpi_k \leftarrow \varpi_k + \frac{1}{N}\left(o_k - \varpi_k\right) \tag{26a}$$

$$\mu_k \leftarrow \mu_k + o_k\left(\frac{x_{new} - \mu_k}{\varpi_k N + 1}\right) \tag{26b}$$

$$\sigma_k^2 \leftarrow \sigma_k^2 + o_k\left(\frac{\varpi_k N (x_{new} - \mu_k)^2}{(\varpi_k N + 1)^2} - \frac{\sigma_k^2}{\varpi_k N + 1}\right) \tag{26c}$$

where σ_k^2 is equal to τ_k^{-1}.

When the new observed sample cannot be modeled by the existing components, a new component is created with mixing coefficient ϖ_{new}, mean value μ_{new} and standard deviation σ_{new}, the parameters of which are given as

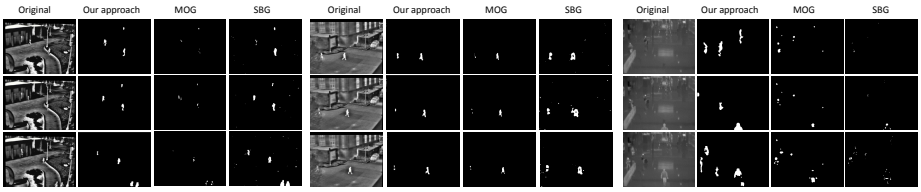
$$\varpi_{new} = \frac{1}{N} \ , \ \mu_{new} = x_{new} \ , \ \sigma_{new}^2 = \frac{(2\epsilon)^2 - 1}{12} \tag{27}$$

From (27), we see that the mixing coefficient for the new component is equal to $1/N$ since it models only one sample (the new observed one), its mean value equals the value of the new sample and its variance the variance of the Uniform distribution, whose the lower and upper bounds are $x_{new} - \epsilon$ and $x_{new} + \epsilon$ respectively. When a new component is created the values for the parameters for all the other components remain unchanged. After each adaptation of the system to new observed samples, either they modeled or not, the mixing coefficients of the components are normalized to sum to one, while components with mixing coefficients less than $1/N$ are removed.

Figure 1 presents the adaptation of our model and the model presented in [19] to new observed data. To evaluate the quality of the adaptation of the models, we used a toy dataset with 100 observations, which were generated from two Normal distributions with mean values 16 and 50 and standard deviations 1.5 and 2.0 respectively. The initially trained models are presented in the left column. Then, we generated 25 new samples form a Normal distribution with mean value 21 and standard deviation 1.0. Our model creates a new component and successfully fits the data. On the contrary, the model of [19] is not able to capture the statistical relations of the new observations and fails to separate the data generated from distributions with mean values 16 and 21 (middle column). The quality of our adaptation mechanism becomes more clear in the right column, which presents the adaptation of both models after 50 new observations.

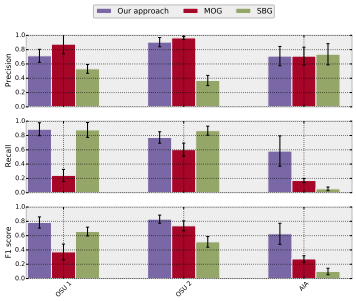
6 Background subtraction

In this section we utilize our model for background subtraction. We initially capture N frames used to create a history of infrared responses for each pixel. These histories act as observed data and used to train one model for each pixel. To classify a pixel of a new captured frame as background or foreground, we compute the probability its value to be represented by the mixture model. If this value is larger than a threshold the pixel is classified as background, otherwise it is classified as foreground. The threshold is defined in relation to the parameters of the mixture, in order to be dynamically adapted.



(a) OSU 1 dataset (a) OSU 2 dataset (a) AIA dataset

Fig. 2. Visual results for all datasets.



(a) Precision, recall and F1 score

			OSU 1	OSU 2	AIA
Our approach	Precision	Worst	0.49	0.75	0.42
		Best	0.88	0.97	0.84
	Recall	Worst	0.54	0.65	**0.25**
		Best	**0.96**	0.95	**0.98**
	F1 score	Worst	**0.60**	**0.74**	**0.31**
		Best	**0.88**	**0.94**	**0.87**
MOG	Precision	Worst	**0.52**	**0.92**	**0.44**
		Best	**0.99**	**0.99**	0.84
	Recall	Worst	0.12	0.46	0.12
		Best	0.39	0.78	0.21
	F1 score	Worst	0.20	0.62	0.19
		Best	0.56	0.87	0.33
SBG	Precision	Worst	0.41	0.23	0.42
		Best	0.66	0.75	**0.90**
	Recall	Worst	**0.56**	**0.78**	0.02
		Best	**0.96**	**0.99**	0.10
	F1 score	Worst	0.53	0.36	0.05
		Best	0.78	0.59	0.16

(b) Best/worst case for precision, recall and F1 score

Fig. 3. Algorithms performance per dataset.

7 Experimental results

For evaluating our algorithm, we used the Ohio State University (OSU) thermal datasets and an dataset captured at Athens International Airport (AIA) during eVACUATE fp7 project OSU datasets contain frames that have been captured using a thermal camera and have been converted to grayscale images. In contrast, the AIA dataset contains raw thermal frames whose pixel values correspond to the real temperature of objects. OSU datasets [21, 22] are widely used for benchmarking algorithms for pedestrian detection and tracking in infrared imagery. Videos were captured under different illumination and weather conditions. AIA dataset was captured using a Flir A315 camera at different Airside Corridors and the Departure Level. Totally, 10 video sequences were captured, with frame dimensions 320×240 pixels of total duration 32051 frames, at 7.5fps.

We compared our method with method presented by Zivkovic in [19] (MOG), which is one of the most robust and widely used background subtraction technique, and with the method for extracting the regions of interest presented in [22] (SBG). To conduct the comparison we utilized the objective metrics of *recall*, *precision* and *F1 score* on a pixel wise manner. Figure 2 visually present the performance of the three methods. Our method outperforms both MOG and SBG on all datasets. While MOG and SBG perform satisfactory on grayscale frames of OSU datasets, their performance collapses when they applied on AIA dataset, which contains actual thermal responses. Regarding OSU datasets, although MOG algorithm presents high precision it yields very low

recall values, i.e. the pixels that have been classified as foreground are indeed belong to the foreground class, but a lot of pixels that in fact belong to background have been misclassified. SBG algorithm seems to suffer by the opposite problem. Regarding AIA dataset, our method significantly outperforms both methods. In particular, MOG and SBG algorithms present high precision, but their recall values are under 0.2. Figure 3(a) presents average precision, recall and F1 score per dataset and per algorithm for all frames examined to give an objective evaluation. In Figure 3(b) presents the best and worst case in terms of precision, recall and F1 score among all frames examined.

Regarding computational cost, the main load of our algorithm is in the implementation of EM optimization. In all experiments conducted, the EM optimization converges within 10 iterations. Practically, the time required to apply our method is similar to the time requirements of Zivkovic's method making it suitable for real-time applications.

8 Conclusions

This paper presents a background subtraction method applicable to thermal imagery, based on Gaussian mixtures with unknown number of components. We analytically derive the solutions that describe the parameters of the model and we use the EM optimization to estimate their values, avoiding this way sampling algorithms and high computational cost. Due to its low computational cost and the real-time operation, our method is suitable for real-world applications.

Acknowledgements

This research was funded from European Unions FP7 under grant agreement n.313161, eVACUATE Project (www.evacuate.eu)

References

1. Porikli, F.: Achieving real-time object detection and tracking under extreme conditions. Journal of Real-Time Image Processing **1** (2006) 33–40
2. Tuzel, O., Porikli, F., Meer, P.: Pedestrian detection via classification on riemannian manifolds. IEEE Trans. on Pattern Analysis and Machine Intelligence **30** (2008) 1713–1727
3. Jungling, K., Arens, M.: Feature based person detection beyond the visible spectrum. In: IEEE Computer Vision and Pattern Recognition Workshops, 2009. CVPR. (2009) 30–37
4. Latecki, L., Miezianko, R., Pokrajac, D.: Tracking motion objects in infrared videos. In: IEEE Conf. on Advanced Video and Signal Based Surveillance, 2005. AVSS. (2005) 99–104
5. Wang, W., Zhang, J., Shen, C.: Improved human detection and classification in thermal images. In: 2010 17th IEEE Int. Conf. on Image Processing (ICIP). (2010) 2313–2316
6. Doulamis, N.D.: Coupled multi-object tracking and labeling for vehicle trajectory estimation and matching. Multimedia Tools and Applications **50** (2010) 173–198
7. Kosmopoulos, D.I., Doulamis, N.D., Voulodimos, A.S.: Bayesian filter based behavior recognition in workflows allowing for user feedback. Computer Vision and Image Understanding **116** (2012) 422–434

8. Voulodimos, A.S., Doulamis, N.D., Kosmopoulos, D.I., Varvarigou, T.A.: Improving multi-camera activity recognition by employing neural network based readjustment. Applied Artificial Intelligence **26** (2012) 97–118
9. Brutzer, S., Hoferlin, B., Heidemann, G.: Evaluation of background subtraction techniques for video surveillance. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2011) 1937–1944
10. Herrero, S., Bescs, J.: Background subtraction techniques: Systematic evaluation and comparative analysis. In: the 11th International Conference on Advanced Concepts for Intelligent Vision Systems. ACIVS '09, Berlin, Heidelberg, Springer-Verlag (2009) 33–42
11. El Baf, F., Bouwmans, T., Vachon, B.: Fuzzy statistical modeling of dynamic backgrounds for moving object detection in infrared videos. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. (2009) 60–65
12. Zheng, J., Wang, Y., Nihan, N., Hallenbeck, M.: Extracting roadway background image: Mode-based approach. Transportation Research Record: Journal of the Transportation Research Board **1944** (2006) 82–88
13. Elgammal, A., Harwood, D., Davis, L.: Non-parametric model for background subtraction. In Vernon, D., ed.: Computer Vision ECCV 2000. Number 1843 in Lecture Notes in Computer Science. Springer Berlin Heidelberg (2000) 751–767
14. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfinder: real-time tracking of the human body. IEEE Trans. on Pattern Analysis and Machine Intelligence **19** (1997) 780–785
15. Messelodi, S., Modena, C.M., Segata, N., Zanin, M.: A kalman filter based background updating algorithm robust to sharp illumination changes. In Roli, F., Vitulano, S., eds.: Image Analysis and Processing ICIAP 2005. Number 3617 in Lecture Notes in Computer Science. Springer Berlin Heidelberg (2005) 163–170
16. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: principles and practice of background maintenance. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999. Volume 1. (1999) 255–261 vol.1
17. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: Computer Vision and Pattern Recognition. IEEE Conf. on. Volume 2. (1999) –252 Vol. 2
18. Makantasis, K., Doulamis, A., Matsatsinis, N.: Student-t background modeling for persons' fall detection through visual cues. In: 2012 13th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS). (2012) 1–4
19. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. Volume 2. (2004) 28–31 Vol.2
20. Zivkovic, Z., van der Heijden, F.: Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters **27** (2006) 773–780
21. Davis, J., Sharma, V.: Fusion-based background-subtraction using contour saliency. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, 2005. CVPR Workshops. (2005) 11–11
22. Davis, J.W., Sharma, V.: Background-subtraction in thermal imagery using contour saliency. International Journal of Computer Vision **71** (2007) 161–181
23. Elguebaly, T., Bouguila, N.: Finite asymmetric generalized gaussian mixture models learning for infrared object detection. Comp. Vision and Image Understanding **117** (2013) 1659–1671
24. Dai, C., Zheng, Y., Li, X.: Pedestrian detection and tracking in infrared imagery using shape and appearance. Computer Vision and Image Understanding **106** (2007) 288–299
25. Bishop, C.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer (2007)
26. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
27. Dasgupta, S., Hsu, D.: On-line estimation with the multivariate gaussian distribution. In: 20th Annual Conf. on Learning Theory. COLT'07, Springer-Verlag (2007) 278–292