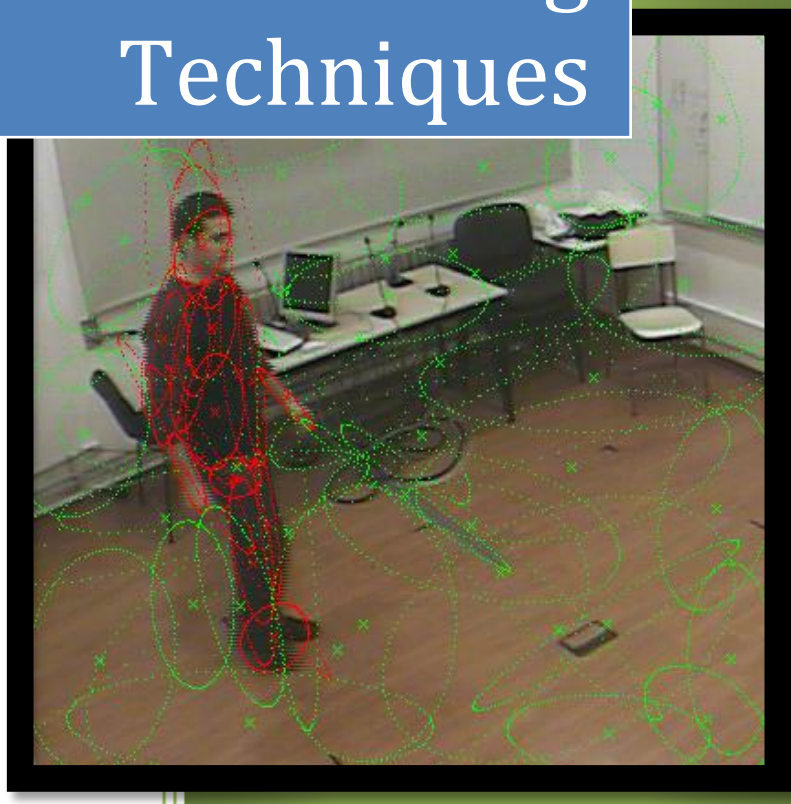
Departament de Teoria del Senyal i Comunicacions
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Master Thesis

2009

Foreground Segmentation and Tracking based on Foreground and Background Modeling Techniques



Master MERIT

TSC Teoria del Señal y las Comunicaciones

Author: Jaime Gallego Vila
Directors: Montse Pardàs, Gloria Haro
ETSETB TSC
09/02/2009

Acknowledgments

Thanks to Montse Pardàs and Gloria Haro to direct this Master Thesis with attention and dedication in all moment.

Thanks to my colleagues, in special to Marcel Alcoverro, David Bernal, Cristian Cantón, Albert Gil , Adolfo Lopez, Jordi Luque, Jordi Pont and Jordi Salvador and Carlos Segura.

Thanks to my friends.

Thanks to Mariona, for her support.

And special acknowledgments to my family to be close to me in all moments.

Than you.

Jaume Gallego Vila
February 2009

TABLE OF CONTENTS

TABLE OF FIGURES

Dedicado a mi familia...

Si buscas resultados distintos,

no hagas siempre lo mismo.

Albert Einstein (1879-1955)

1 INTRODUCTION

Foreground segmentation is a fundamental first processing stage for vision systems which monitor real-world activity being of great interest in many applications. For instance, in videoconferencing once the foreground and the background are separated, the background can be replaced by another image, which then beautifies the video and protects the user privacy. The extracted foreground objects can be compressed to facilitate efficient transmission using object-based video coding. As an advanced video editing tool, segmentation also allows people to combine multiple objects from different video and create new artistic results. In 3D multi-camera environments, robust foreground segmentation allows a correct 3-dimesional reconstruction without background artifacts while, in video surveillance tasks, foreground segmentation allows a correct object identification and tracking.

The current Master Thesis is defined in this framework: *Foreground segmentation and tracking based on foreground and background modeling techniques* with the main objective of developing segmentation and tracking methods for moving and static monocular video sequences. In the following lines, we will be expose the project kernel with three main contributions to the state of the art:

- Adaptation of the *foreground segmentation and tracking technique via SCGMM* (1) for moving monocular video sequences.
- Foreground segmentation in monocular static sequences via SCGMM-1Gaussian combined models.
- Foreground segmentation in monocular static sequences via SCGMM-1Gaussian combined models with Foreground model updating before decision..

All three methods are foreground segmentation novel techniques that we propose in this project and have been tested successfully after their implementation in C++ for the UPC image processing group software *ImagePlus*.

The work can be distributed in the following steps:

- Study of the Current State of The Art literature
- Study of previous foreground segmentation techniques of the image group
- Implementation of the *SCGMM foreground-background joint tracking* algorithm
- Developing the three novel techniques detailed above

The final results of this work are three applications to segment and track foreground objects: one to segment and track objects in monocular moving camera sequences, and two others to segment objects in a foreground-background color similarity situation.

The manuscript is organized as follows: In the next section *"Project Framework"*, we will describe the bases of the techniques explained in this project. Section two is devoted to the foreground segmentation and tracking *"State of the Art"*. In *"Our Work"*, section three, we will detail the techniques that we propose, including theoretical bases, implementation overview

and some results for each one of the methods. The manuscript will finish with *"Conclusions"* and *"Future Work"* sections where we will summarize the results obtained and we will propose future lines of development.

2 PROJECT FRAMEWORK

The Project Framework is the detection and tracking of foreground objects in static and moving video sequences.

The objective of a foreground segmentation and Tracking is to segment the scene in foreground objects and background and establish the temporal correspondence of the foreground objects. In this project we will focus on techniques that are based on a classification using a statistical model of the background and the foreground. For this reason, we will assume that the segmentation of the first frame is provided. Our objective will be to improve the models and define an appropriate updating of these models to reach a correct foreground-background segmentation minimizing False Negatives and False Positives. The tracking process makes the correspondence of the segmented objects with the objects being tracked from previous frames. Depending on the technique, the tracking can be clearly separated from the segmentation (when previous foreground information is not used for the segmentation) or can be implicit in the foreground segmentation (when we are using a priori information of the object).

Dickinson *et al.* (2) and Yu *et al.* (1) propose a joint segmentation and tracking system that works similarly to the previous workflow, based on Spatial Color Gaussian Mixture Models foreground and background modeling.

3 STATE OF THE ART

In this section several foreground segmentation and tracking methods of the literature will be revised.

This section consists of two subsections:

- **Foreground segmentation and tracking based only on background modeling.**
- **Foreground segmentation and tracking based on background and foreground modeling.**

3.1 FOREGROUND SEGMENTATION AND TRACKING BASED ONLY ON BACKGROUND MODELING

These common techniques propose to use a background probabilistic model to detect foreground regions as a background exception. Without a foreground model, these systems need a two step process to achieve a detection and identification of the object along the sequence:

- **Foreground segmentation:**
  Consists in segmenting all foreground pixels of the image to obtain foreground Connected Components for each frame. This segmentation is obtained detecting all pixels that don't belong to background model.

- **Object Tracking:**
  After the foreground segmentation step, a tracking system is used to maintain a temporal consistence of the foreground connected components between frames. This process is needed because no prior information of the objects is used to segment them. Hence, this tracking step is used to identify which segmented connected component corresponds to each object being tracked.

3.1.1 Foreground segmentation. Methods

In this section we explain some main foreground segmentation techniques to detect the foreground objects, without any prior information of these objects. M. Piccardi (3) reviews this issue while Butler *et al.* (4) refer to other methods.

3.1.1.1 Temporal Median Filter

Proposed by Lo and Velastin (5). This system proposes to use the last N frames to calculate the median for each pixel (i,j) and conform a reference background model. The system uses a buffer for the N last pixel values, to update the median for each frame.

At the beginning of the sequence, the system learns the first N frames, in a period of time called "Training Time" to find the *initial reference background model*, by means of ordering the N pixel values from minor to major and taking the pixels placed in N/2 position to conform this model. After the training period, for each new frame, each pixel input value will be compared with its corresponding pixel background model value. If the pixel in value under analysis is within certain allowed limits, it will be considered that the pixel matches the background model and its input value will be included in the pixel buffer (LIFO queue). Otherwise, if the pixel value is outside these limits, it will be classified as foreground, and no update will be done.

In Figure 1 an example of reference background model is shown. The main disadvantages of this method are that it needs a buffer of size N for each pixel, and that it doesn't present a rigorous statistical base.

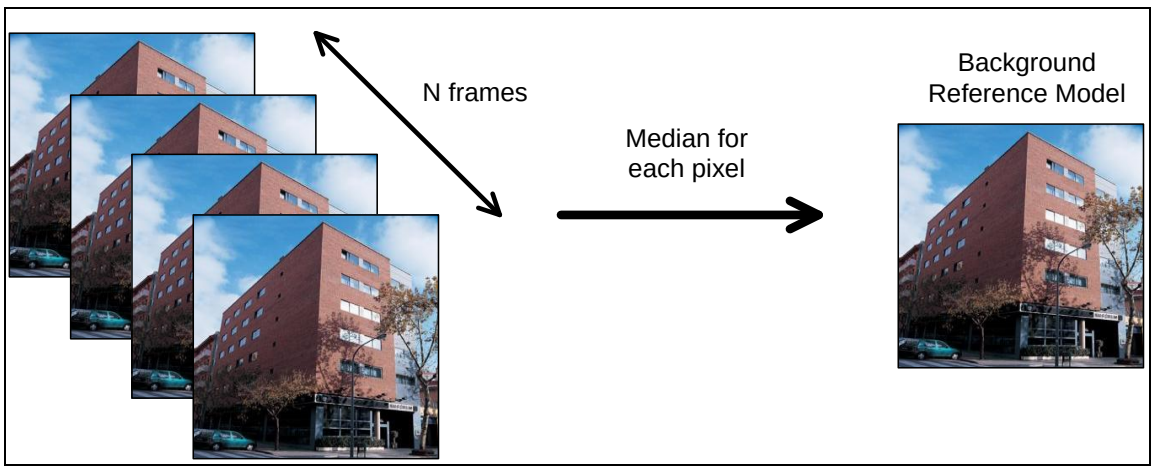


Figure 1 Example of Temporal Median Filter background model

3.1.1.2 Running Gaussian average

Wren et al (6), propose to model the background by analyzing each pixel (i,j) of the image. The background model consists in the probabilistic modeling of each pixel value via Gaussian probability function (p.d.f.), characterized by its mean μ and variance σ^2. In Figure 2 it is shown an image with the system idea, where each pixel appear modeled with a Gaussian distribution.

Mean and variance for each frame t ($\mu_t, {\sigma_t}^2$) are updated as follows:

$$\mu_t = \rho I_t + (1-\rho)\mu_{t-1}$$

Equation 3-1

$$\sigma_t^2 = \rho \cdot d^2 + (1-\rho)\sigma_{t-1}^2$$

Equation 3-2

$$d = (I_t - \mu_t)^2$$

Equation 3-3

Where I_t is the value of the pixel under analysis in the current frame; μ_t, σ_t^2 are, respectively, the mean and variance of the Gaussian distribution, ρ is a weight that defines the updating velocity (commonly $\rho = 0.01$) and d is the Euclidean distance between the Gaussian mean and the pixel value.

This updating step allows a background model evolution, making it robust to soft illumination changes, a common situation in outdoor scenarios.

For each frame, the pixel value I_t is classified as foreground according to Equation 3-4:

$$|I_t - \mu_t| > k\sigma_t$$

Equation 3-4

Where k is the threshold parameter (usually 2.5).

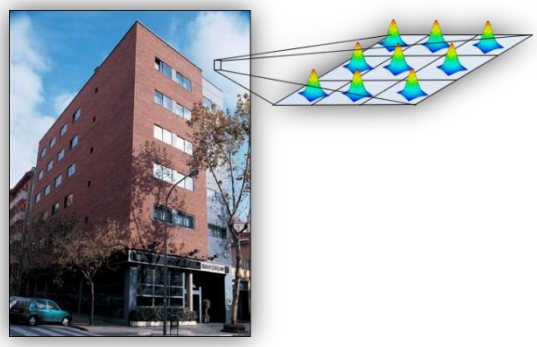


Figure 2 Running Gaussian Average graphic idea

When the inequality is satisfied, the pixel I_t is considered foreground. Otherwise, it is considered background.

Koller *et al.* (7) emphasize that the updating process has to be done only if the pixel is considered background, replacing Equation 3-4 by Equation 3-5.

$$\mu_t = M\mu_{t-1} + (1 - M)(\alpha I_t + (1-\alpha)\mu_{t-1})$$

Equation 3-5

Where M=1 if I_t is considered as foreground, and M=0 otherwise.

This method presents the main advantages of less memory and computational cost requirements: only two parameters per pixel are stored (mean and variance (μ_t, σ^2))

3.1.1.3 Gaussian Mixture Model

This method appears as an extension from the previous one to improve the results in dynamic background scenarios (moving tree leafs, flags on the wind…). The system proposes to use a Gaussian Mixture Model to model the p.d.f. of each pixel (i,j), using a different Gaussian distribution for each pixel to create the background model.

Stauffer and Grimson (8), present this method where it is described the probability of observing a pixel value x in the time t as a Gaussian Mixture Model is defined as:

$$P(x_t) = \sum_{i=1}^{k} \omega_{i,t} \cdot \eta\,(x_t, \mu_{i,t}, \Sigma_{i,t})$$

Equation 3-6

Where k is the number of gaussian distributions used to model each pixel, $\omega_{i,t}$ is the weight of the Gaussian (how much information it represents), $\mu_{i,t}$ is the mean of the Gaussian, $\Sigma_{i,t}$ is the matrix variance of the Gaussian simplified as $I \cdot \sigma_{i,t}^2$ (incorrelated components (r, g, b) and the same variance for each color), $\eta\,(x_t, \mu_{i,t},\ \Sigma i, t)$ is the Gaussian function.

The number of Gaussian distributions commonly used to model each pixel is three or five. These Gaussians are correctly combined thanks to the weight factor $\omega_{i,t}$, which is modified (increased or decreased), for each Gaussian, according to the number of times the input pixel matches the Gaussian distribution.

The weights are normalized via Equation 3-7:

$$\omega_{i,t} = \frac{\omega_{i,t}}{\sum_{i=1}^{k} \omega_{i,t}}$$

Equation 3-7

The background is modeled by the B Gaussian distributions with highest weight $\omega_{i,t}$ and lowest variance, according to the next inequality:

$$B = \arg\min_{\mathrm{b}} \left(\sum_{i=1}^{b} \omega_{i,t} > T \right)$$

Equation 3-8

Where T is the decision threshold (commonly 0.6) and B is the minimum number of Gaussian distributions to include in the summation (sorted by ω/σ), in order to verify the inequality.

The background is usually more static and appears with more frequency, this is the reason why it is modeled by the first B Gaussian distributions, which are those that have been used more times and at the same time are more compact.

To decide if the input pixel value matches any Gaussian distribution, the following expression is evaluated:

$$|X_t - \mu_{i,t}| > \varphi\, \sigma_{i,t}$$

Equation 3-9

Where X_t is the pixel value (r, g, b) in the frame t, φ is a constant parameter threshold (commonly $\varphi = 2.5$), $\mu_{i,t}$ is the i^{th} Gaussian mean and $\sigma_{i,t}$ is its standard deviation.

- If the inequality is true for all Gaussian distributions, the pixel belongs to the foreground because it doesn't match the probabilistic background model. A graphic example can be observed in Figure 3

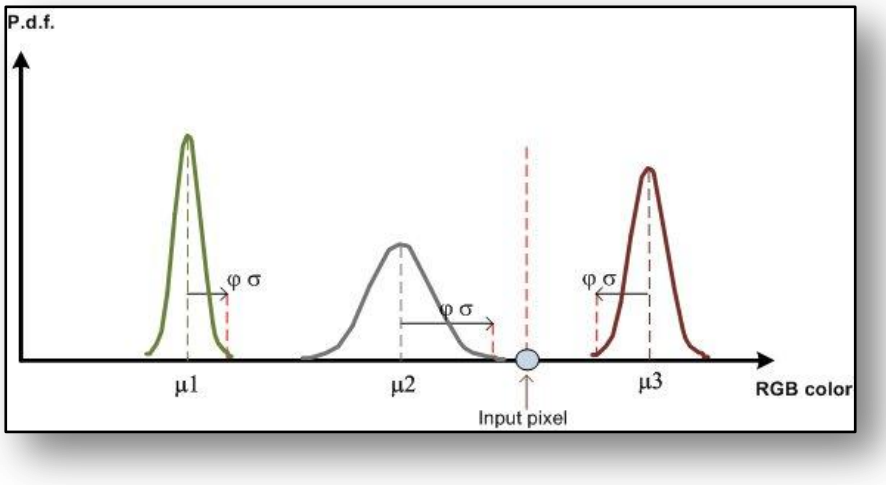


Figure 3 Gaussian mixture model foreground pixel detection

- If the inequality is not true for one or more than one Gaussian, it is decided that the input value matches the background probabilistic model. In the case that the pixel matching more than one Gaussian distribution, it will be decided that the Gaussian with higher weight and lower variance better represents the pixel. In Figure 4 we can observe a graphical example where the input pixel matches one of the three Gaussian distributions that model the pixel.

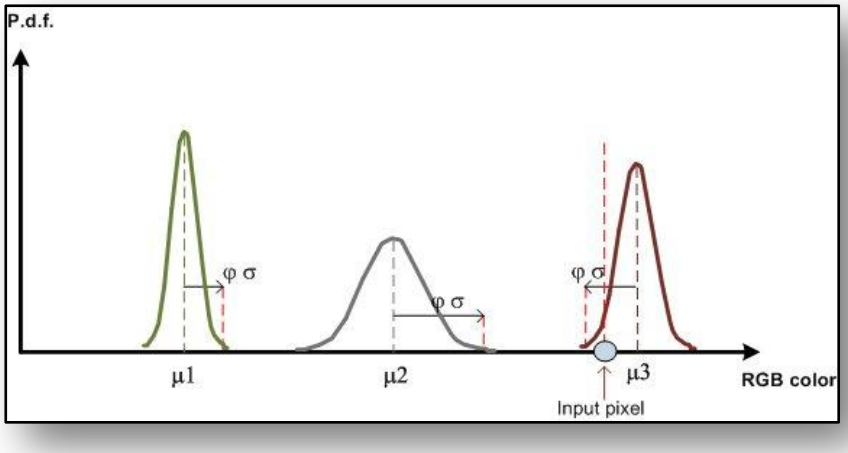


Figure 4 Gaussian mixture model background pixel detection

If the input pixel matches one of the B first Gaussian distributions, that conform the background model, it implies that this value has appeared enough to be considered as background pixel. Otherwise, it will be considered as foreground.

If the input pixel doesn't match any of the K Gaussian distributions of the probabilistic model, a new Gaussian distribution is created. This allows the update of the background model when there are changes in the background (for example, when a foreground object becomes part of the background). In this case, the Gaussian distribution with the lowest weight is removed and replaced by a new Gaussian modeling the new value. This is done in order to maintain the initial number of Gaussians.

When the input pixel has matched one of the K Gaussian distributions that form the pixel probabilistic model, a model updating is carried on in the following way:

The Gaussian distribution that has matched the pixel is updated:

Mean updating:

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho\, X_t$$

Equation 3-10

Variance updating:

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T (X_t - \mu_t)$$

Equation 3-11

For all the Gaussian distributions of the pixel probabilistic model:

Weight updating:

$$\omega_{k,t} = (1 - \alpha)\, \omega_{k,t-1} + \alpha(M_{k,t})$$

Equation 3-12

Where:

ρ: mean and variance updating rate. Commonly. $\rho = 0.01$
T: denotes transposed.
α: weight updating rate. Commonly $\alpha = 0.005$
$M_{k,t}$: is 1 for the Gaussian distribution that has matched the pixel value and 0 otherwise.

The main advantage of this algorithm is that the probabilistic model of the pixel represents several pixel values (r, g, b). This technique allows foreground segmentation in dynamic background scenarios.

3.1.1.4 Eigenbackgrounds

Oliver *et al* (9) propose this foreground segmentation technique based on the eigenbackground decomposition of the overall image. The work flow is as follows:

- Learning step
    - The average image μ_b is obtained by means of analyzing n training frames of the sequence. Then, the difference between each frame and μ_b is calculated
    - The covariance matrix is calculated and the M eigenvectors corresponding to the largest eigenvalues are kept in a eigenvector matrix ϕ_{Mb} with MxP dimension.

- Classification step
    - Each new frame *I* is projected to the eigenspace as: $I' = \phi_{Mb} (I - \mu_b)$
    - Then P is projected back to the image space as $I'' = \phi^T_{Mb} I' + \mu_b$. The eigenspace is a good model for static regions of the image, but not for moving objects. Then, *I''will not* contain any of these objects.
    - Those pixels where $|I - I''| > T$ is satisfied, will be considered as foreground.

This method has higher computational cost than the methods showed before, due to the matrices operations needed to obtain image eigenvectors and eigenvalues.

3.1.3 Tracking. Methods

Over the twenty last years, there has been a considerable activity in the area of foreground objects tracking. This interest arises from the necessity of different applications for video surveillance, video conferencing, video coding etc.

In this section we explain some tracking algorithms that are currently used to track foreground objects previously detected via foreground segmentation algorithms. P. F. Gabriel *et al.* (10) extends the state of the art of foreground objects tracking detailing different techniques.

3.1.3.1 Tracking Based on Connected Components

A Connected Components in an image is a region (group of pixels) where any two pixels belonging to that region are connected with a path of pixels inside the same region

L-Qu *et al.* (11), propose a connected components based tracking system where, for each frame, every tracked object is associated to any foreground connected component from the previous foreground segmentation step. In this method, a feature vector is saved for each object containing the following information:

- Width
- Height
- Area
- Centroid
- Histogram

For each input frame, the distance between the connected components features and the features of each object is analyzed. The objects' features whose connected component minimizes this distance, will be update.

The connected components not associated to any object for several consecutive frames will be detected as new object to track.

The objects not associated to any connected component for several consecutive frames, will be removed.

The main disadvantages of this technique appear in the occlusion situation between two or more objects, where the system can't guarantee correct tracking due to the merging of two objects in one connected component.

Another source of errors appears when there are problems with the foreground segmentation, like false object detections due to a background dynamic scene or the presence of shadows, or false negative detections due to the similarity between the foreground object and the background.

3.1.3.2 Mean Shift Based Tracking

Mean Shift (12) (13) (14) is a non-parametric technique for the analysis of a complex multimodal feature space. The basic computational module of the technique is an old pattern recognition procedure for feature space analysis, the mean shift. For discrete data the convergence of a recursive mean shift procedure to the nearest stationary point of the underlying density function is proved, thus, it is suited for detecting the modes of the density.

This technique is used in:

- Image filtering preserving discontinuities
- Segmentation
- Real time objects tracking

Mean-shift tracking algorithm is an iterative scheme based on comparing the histogram of the original object to track in the current image frame and the histogram of candidate regions in the next image frame. The aim is to maximize the correlation between two histograms.

The algorithm computes for each object the Equation 3-13:

$$\hat{y}_1 = \frac{\sum_{i=1}^{nh} x_i \; w_i \; g\left(\left\|\frac{\hat{y}_0 - x_i}{h}\right\|^2\right)}{\sum_{i=1}^{nh} \; w_i \; g\left(\left\|\frac{\hat{y}_0 - x_i}{h}\right\|^2\right)}$$

Equation 3-13

Where:

$\hat{y}_1$: object centroid in next frame t+1
x_i: object pixels
nh: number of object pixels for kernel h
g(): profile kernel K derivative
h: kernel radius
$\hat{y}_0$: object centroid in frame t
w_i: bin weight of the pixel

$$w_i = \sum_{u=1}^{m} \delta\,[b(x_i) - u] \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{y}_0)}}$$

Equation 3-14

Where:

b(x_i): histogram bin of the pixel
u: histogram bin
m: number of histogram bins
δ[x-y]: Kronecker delta (1 if x=y; 0 otherwise)
$\hat{p}_u(\hat{y}_0)$: color u value in the t+1 histogram calculated with $\hat{y}_0$ position and the object area in t
$\hat{q}_u$: color u value in the objet histogram in t

The algorithm work flow is the following:

1. Find object histogram $\hat{q}_u$
2. Find histogram of frame t+1 $\hat{p}_u(\hat{y}_0)$
3. Obtain w_i weights for each histogram bin according to Equation 3-14
4. Obtain new object centroid based on Mean Shift vector. Equation 3-13
5. Calculate $\hat{p}_u(\hat{y}_1)$ and to evaluate the distance between histograms: Bhattacharyya distance:

$$\rho[\hat{p}\,(\hat{y}_1), \hat{q}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\hat{y}_1)\;\hat{q}_u}$$

Equation 3-15

6. While $\rho[\hat{p}\,(\hat{y}_1), \hat{q}] < \rho[\hat{p}\,(\hat{y}_0), \hat{q}]$

   Do $\hat{y}_1 \leftarrow \frac{1}{2}\,(\hat{y}_0 + \hat{y}_1)$

   Evaluate $\rho[\hat{p}\,(\hat{y}_1), \hat{q}]$

7. If $\|\hat{y}_1 - \hat{y}_0\| < \epsilon$ → STOP

   Otherwise: → Establish $\hat{y}_0 \leftarrow \hat{y}_1$ go to step 2

This technique has been used for tracking connected components after foreground segmentation in (15). In the work we presented in (16), we developed a tracking system using foreground segmentation that avoided erroneous objects detection due to a wrong object segmentation in more than one Connected Component.

3.1.3.3 Particle Filter

Particle filters (17) (18) (19) (20) were proposed in 1993 by N. Gordon, D. Salmond and A. Smith to implement Bayesian recursive filters.

Particle filters are sequential Monte Carlo methods based upon point mass (or "particle") representations of probability densities, which can be applied to any state space model, and which generalize the traditional Kalman filtering methods.

The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples and weights. As the number of samples becomes very large, this Monte Carlo characterization becomes an equivalent representation to the usual functional description of the posterior pdf.

In order to develop the details of the algorithm for objects tracking, let $X = \{x^{(n)}, w^{(n)}) | n = 1 .. N\}$, be a *Random Mearure* where $x^{(n)}$,is a set of support points with associated weights $w^{(n)}$ where $\sum_{n=1}^{N} w^{(n)} = 1$. Then, the posterior density at *k* can be approximated as:

$$p(x_{0:k}|z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k}|x_{0:k}^i)$$

Equation 3-16

Therefore a discrete weighted approximation to the true posterior $p(x_{0:k}|z_{1:k})$ is obtained. The weights are chosen using the principle of *Importance Sampling* (21).

Hence, it's possible to convert difficult integrals into summations easily computable.

Algorithm Work Flow

The particles are possible states of the process, and can be represented as points in the space state. It has four main stages:

- **Initialization.**
- **Updating.**
- **Estimation.**
- **Prediction.**

To track a foreground object along a frame sequence, the particle filter "throws" randomly over the image a set of points (initialization stage, a particle set in a random state is created), after some calculations, a value to each point of the set is assigned(updating stage). Starting from these values a new set of points that will replace the previous one is created. The values given to each point make more probable to choose those points that remains over the region occupied by the object under analysis (estimation stage). Once a new set of points is created, a light modification stage (position) of each point of the set is performed, so as to estimate the object's state in next instant (prediction stage).

3.2 FOREGROUND SEGMENTATION AND TRACKING BASED ON FOREGROUND AND BACKGROUND MODELING

This kind of techniques propose to combine both foreground and background probabilistic models to segment foreground regions from background. In this way, foreground segmentations won't be performed as an exception to the background model, improving the foreground segmentation results. The main advantage of this approach is that using prior information of foreground objects, we are segmenting foreground regions that we know belong to each object. Hence, the tracking step is not needed because this process is implicitly included in the object segmentation. Moreover, the segmentation will be more robust in situations where the foreground colors are similar to the background

In this way, foreground regions will be detected when the foreground model provides a better representation of the object than the background model.

However, we need to take into account that an initialization of the foreground objects needs to be done. For this aim, a generic model for the foreground needs to be defined, or otherwise we can use an "exception to the background" model to find new appearing objects.

3.2.1 Pixel wise based foreground segmentation by means of Foreground uniform model and Background Gaussian model

This is a pixel-wise foreground segmentation approach for monocular static sequences that combines background and foreground color probabilistic modeling. As Landabaso and Pardas (22) propose, in order to obtain an accurate 2D-segmentations using a Bayesian framework, a single-class statistical model is adopted for modeling the background color of a pixel $z_i = (r, g, b)$ as in (6), and a uniform statistical model is used for modeling the foreground.

We include the description of this system here because it is a first step towards foreground and background modeling, and it can be used for the aforementioned initialization.

Hence, given observations of pixel color value z_i across time, a Gaussian probability density function is used to model the background color as can be read in section 3.1.1.2:

$$G\left(z_i; \mu_{\mathrm{bg,i}}, \Sigma_{\mathrm{bg,i}}\right) = \frac{1}{(2\pi)^{\mathrm{d}/2} \cdot \left|\Sigma_{\mathrm{bg,i}}\right|^{1/2}} \cdot \mathrm{e}^{-\frac{1}{2}\left[\left(\mathrm{z_i}-\mu_{\mathrm{bg,i}}\right)^{\mathrm{T}} \Sigma_{\mathrm{bg,i}}^{-1} \left(\mathrm{z_i}-\mu_{\mathrm{bg,i}}\right)\right]}$$

Equation 3-17

Where $\mathrm{d} = 3$, and i denotes the pixel spatial index. Often it is assumed that $\Sigma_{\mathrm{bg,i}}$ is diagonal with (r, g, b) sharing the same variances: $\Sigma_{\mathrm{bg,i}} = \sigma_{z_i}^2 \cdot I$.

The adaptation of the background model is the same proposed in (6) and also can be read in section 3.1.1.2.

The foreground model proposed is a uniform p.d.f. to model the foreground process in each pixel, which is in fact the probabilistic extension of classifying a foreground pixel as an exception to the model. Since a pixel admits 256^3 colors in the RGB color space, its p.d.f. is modelled as:

$$U_{z_i}(z_i) = \frac{1}{256^3}$$

Equation 3-18

Once the foreground and background likelihoods of a pixel are introduced, and assuming that we have some knowledge of foreground and background prior probabilities, $P(fg)$ and $P(bg)$ respectively (approximate values can be obtained by manually segmenting the foreground in some images, and averaging the number of segmented points over the total), the classification of a pixel as foreground can be done when the following inequality is verified:

$$p(fg|z_i) > p(bg|z_i)$$

Equation 3-19

Where $P(fg|z_i)$, $P(bg|z_i)$ are the posterior probability obtained by a Bayes development in

$$P(l|z_i) = \frac{p(z_i|l) \cdot P(l)}{p(z_i)}$$

Equation 3-20

Where $l \in \{fg, bg\}$.

Then, in the case of the models described above, Equation 3-19 can be expressed as:

$$p(z_i|fg) \cdot P(fg) > p(z_i|bg) \cdot P(bg)$$

$$\frac{1}{256^3} \cdot P(fg) > G\left(z_i; \mu_{\mathrm{bg,i}}, \Sigma_{\mathrm{bg,i}}\right) \cdot P(bg)$$

Equation 3-21

In practice this is very similar to the approach defined in (6) consisting in determining background when a pixel value falls within 2.5 standard deviations of the mean of the Gaussian.

3.2.2 Regiond based foreground segmentation based on spatial-color Gaussians Mixture Models (SCGMM)

In this kind of modeling both the foreground and background are modeled using spatial-color Gaussian mixture models (SCGMM). Each pixel of the image is defined with five dimensional feature vector, i.e., $\mathrm{z} = (x, y, r, g, b)$, representing the pixel's spatial information, (x, y) coordinates, and color information, (r, g, b) color values. Then, the likelihood of a pixel belonging to the foreground or background can be written as:

$$p(z|l) = \sum_{k=1}^{K_l} w_{l,k} G(z; \mu_{l,k}, \Sigma_{l,k})$$

Equation 3-22

Where $l \in \{fg, bg\}$ represents foreground or background; $w_{l,k}$ is the prior weight of the k_{th} Gaussian component in the mixture model, and $G(z; \mu_{l,k}, \Sigma_{l,k})$ is the k_{th} Gaussian component:

Figure 5 Spatial representation of the SCGMM models. Foreground SCGMM in red, background SCGMM in green.

$$G(z; \mu_{l,k}, \Sigma_{l,k}) = \frac{1}{(2\pi)^{d/2} \cdot |\Sigma_{l,k}|^{1/2}} \cdot e^{-\frac{1}{2}\left[(z-\mu_{l,k})^T \Sigma_{l,k}^{-1} (z-\mu_{l,k})\right]}$$

Equation 3-23

Where $\mathrm{d} = 5$ is the dimension of the SCGMM models.

It is commonly assumed that the spatial and color components of the SCGMM models are decoupled, i.e., the covariance matrix of each Gaussian component takes the block diagonal form,

$\Sigma_{l,k} = \begin{pmatrix} \Sigma_{l,k,s} & 0 \\ 0 & \Sigma_{l,k,c} \end{pmatrix}$, where s and c stand for the spatial and color features respectively. With such decomposition, each GMM Gaussian component has the following factorized form:

$$\mathrm{G}(\mathrm{z}; \mu_{\mathrm{l,k}}, \Sigma_{\mathrm{l,k}}) = \mathrm{G}(\mathrm{z}; \mu_{\mathrm{l,k,s}}, \Sigma_{\mathrm{l,k,s}}) \cdot \mathrm{G}(\mathrm{z}; \mu_{\mathrm{l,k,c}}, \Sigma_{\mathrm{l,k,c}})$$

Equation 3-24

The parameter estimation can be reached in the initialization period via Bayes' development, with the EM algorithm (23).

As in the previous section, the posterior distribution of a class given the pixel under analysis z can be written as:

$$p(l|z) = \frac{p(z|l) \cdot p(l)}{p(z)}$$

Equation 3-25

Where $p(z|l)$ is the likelihood defined in Equation 3-22, $p(z)$ is the prior probability of the pixel and $p(l)$ is the prior probability of the class obtained according to:

$$p(l) = \gamma_{l,t-1}$$

Equation 3-26

Where $\gamma_{l,t-1}$ is the area covered by each class divided by the total area, and can be obtained from the previous frame. They satisfy $\gamma_{fg,t-1} + \gamma_{bg,t-1} = 1$.

The foreground segmentation using this model is obtained finding the evolution of the foreground-background five dimensional SCGMM models for each video frame, and deciding for each pixel, the one that maximizes the class posterior Equation 3-25

Greenspan *et al.* (24) propose an statistical video representation and modeling squeme where unsupervised clustering via Gaussian mixture modeling extracts coherent space-time regions in feature space, and corresponding coherent segments (video-regions) in the video content, while the system proposed by Yu et al. (1) in "*Monocular Video Foreground/Background Segmentation by Tracking Spatial-Color Gaussian Mixture Models"*, is a good example of the SCGMM application in foreground segmentation task. Next it will be explained in detail because it is the basis of our proposals.

3.2.2.1 Tracking Spatial Color Gaussian Mixture Models (SCGMM)

This technique proposed by Yu, *et al*. (1) presents an approach to segment monocular videos captured by static or hand-held cameras filming large moving non-rigid foreground objects. The foreground and background are modeled using spatial-color Gaussian Mixture Models (SCGMM), and segmented using the graph cut algorithm, which minimizes an Energy function based on: a first order Markov Random Field. With this technique, the authors propose to combine the two SCGMMs into a generative model of the whole image, and maximize the joint data likelihood using a constrained Expectation-Maximization (EM) algorithm.

Using spatial and color information to model the scene, SCGMM has better discriminative power than color-only GMM widely used in pixel wise analysis.

The segmentation problem is solved by means of iterating the tracking-segmentation-updating process showed in Figure 6.

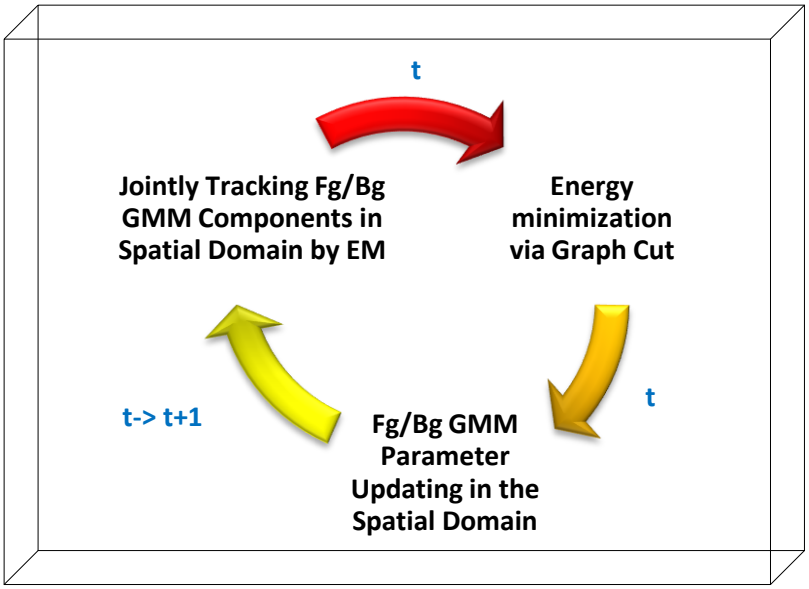


Figure 6 The Iterative Circle of Foreground/Background Segmentation for One Frame

The first frame of the sequence is used to initialize the foreground and background models by means of the EM algorithm (23) in both models. Hence, an initial classification into foreground and background pixels is needed.

For each frame after the first one, first the SCGMM of the foreground and the background are combined and updated with the EM, thus performing a joint tracking of the foreground regions and the background. Afterwards the image SCGMM model is split back into two models, one describing the foreground, the other describing the background. Components belonging to the foreground before tracking are placed in the foreground SCGMM, and components belonging to the background before tracking are placed in the background SCGMM. The two SCGMM models are then used to perform graph cut segmentation, as it is described in Annex I.

The segmentation results can be used for a post-updating of the SCGMM models, where the foreground and background SCGMMs are trained separately with the segmented pixels, which often provides better discriminative power for segmenting future frames.

Considering that the foreground and background colors stay the same across the sequence, a constrained update on the two models is performed. That is, apply Expectation Maximization algorithm on the foreground or background region to update the SCGMM models, forcing the color means and variances to be constant. In this way, propagation errors due to color updates are avoided.

The *joint tracking*, e*nergy minimization* and u*pdating* steps will be explained next.

SCGMM Joint Tracking

Suppose two SCGMM models defined by a set of parameters θ are learned during the system initialization period, in the first frame, using the popular EM algorithm which maximizes the data likelihood (ML) of each segment:

$$\theta^*_{l,0\,ML} \stackrel{\text{def}}{=} \left\{w^*_{l,k,0},\ \mu^*_{l,k,0},\ \Sigma^*_{\mathrm{l,k,0}}\right\} = arg \max_{w^*_{l,k,0},\ \mu^*_{l,k,0},\ \Sigma^*_{\mathrm{l,k,0}}} \mathcal{L}(\theta^*_{l,0}|I_0)$$

$$= arg \max_{w^*_{l,k,0},\ \mu^*_{l,k,0},\ \Sigma^*_{\mathrm{l,k,0}}} \prod_{z_l \in I_0} \left[\sum_{k=1}^{K_I} w_{l,k}\, \mathrm{G}\left(\mathrm{z_l}; \mu_{\mathrm{l,k}}, \Sigma_{\mathrm{l,k}}\right)\right]$$

Equation 3-27

Where $l \in \{fg, bg\}$; z_l are the features of the pixels having label l; I_0 denotes the initialization frame, $w_{l,k}$ is the weight of the Gaussian k with label l, $\mu_{\mathrm{l,k}}$ the mean and $\Sigma_{\mathrm{l,k}}$ the covariance matrix.

An Expectation Maximization algorithm can be formulated to find the maximizer of the likelihood function.

The aim of this part of the process is to propagate these SCGMM models over the rest of the sequence, since both the foreground and background objects can be constantly moving. For this purpose, the algorithm looks for ways to obtain an approximate SCGMM model for the current frame before the graph cut segmentation.

It is assumed that from time *t − 1* to *t*, the colors of the foreground and background objects do not change. Hence, the color parts of the SCGMM models remain identical:

$$\mathrm{G}\left(\mathrm{z_c}; \mu_{\mathrm{l,k,c,t}}, \Sigma_{\mathrm{l,k,c,t}}\right) = \mathrm{G}\left(\mathrm{z_c}; \mu_{\mathrm{l,k,c,t-1}}, \Sigma_{\mathrm{l,k,c,t-1}}\right)$$

Equation 3-28

where *c* denotes the color dimension, $k = 1, \ldots, K_I$ the Gaussian distribution number and $\mathrm{z_c}$ the color pixel information.

Next we explain how to formulate an updating scheme for the spatial parts $\mathrm{G}(\mathrm{z_s};\mu_{\mathrm{l,k,s,t}},\Sigma_{\mathrm{l,k,s,t}})$ given the new input image I_t, where *s* denotes the spatial features.

Since we do not have a foreground/background segmentation on I_t, first a global SCGMM model of the whole image is formed by combining the foreground and background SCGMM models of the previous frame: $\theta_{I,t}^{0}$, where superscript *0* indicates that the parameter set is serving as the initialization value for the later update.

The probability of a pixel of the image $z_i = (r, g, b, x, y)$ given the global model $\theta_{I,t}^{0}$ can be expressed as the combination of both foreground and background models:

$$p(z_t|\theta_{I,t}^{0}) = p(fg)\, p(z_t|\theta_{fg,t-1}) + p(bg)p(z_t|\theta_{bg,t-1}) =$$

$$= \gamma_{fg,t-1}\, p(z_t|\theta_{fg,t-1}) + \gamma_{bg,t-1}p(z_t|\theta_{bg,t-1}) =$$

$$= \sum_{k=1}^{K_I} w'^{0}_{k,t}\ \mathrm{G}(\mathrm{z_s};\ \mu^{0}_{k,s,t},\ \Sigma^{0}_{\mathrm{k,s,t}}) \cdot \mathrm{G}(\mathrm{z_c};\mu_{\mathrm{k,c,t}},\Sigma_{\mathrm{k,c,t}})$$

Equation 3-29

Denote $K_I = K_{fg} + K_{bg}$ as the number of Gaussian components in the combined image level SCGMM model, where we assume the first K_{fg} Gaussian components are from the foreground SCGMM, and the last K_{bg} Gaussian components are from the background SCGMM.

The Gaussian term over the color dimension is defined in Equation 3-28 and remains fixed at this moment. The Gaussian component weights $w'^{0}_{k,t}$, $k = 1, \dots, K_I$ are different from their original values in their individual foreground or background SCGMMs due to $p(fg)$ and $p(bg)$:

$$w'^{0}_{k,t} = \begin{cases} w^{0}_{fg,k,t} \cdot p(fg) & if \quad k \le K_{fg} \\ w^{0}_{bg,(k-K_{fg}),t} \cdot p(bg) & if \quad K_{fg} < k \le K_I \end{cases}$$

Equation 3-30

Given the pixels in the current frame I_t, the objective is to obtain an updated parameter set $\{w'^{*}_{k,t},\ \mu^{*}_{k,s,t},\ \Sigma^{*}_{\mathrm{k,s,t}}\}$ over the spatial domain, which maximizes the joint data likelihood of the whole image, for all $k = 1, \dots, K_I$, i.e.,

$$\{w'^{*}_{k,t},\ \mu^{*}_{k,s,t},\ \Sigma^{*}_{\mathrm{k,s,t}}\} = arg \max_{w^{*}_{k,t},\ \mu^{*}_{k,s,t},\ \Sigma^{*}_{\mathrm{k,s,t}}} \prod_{z_t \in I_t} p(z_t|\theta_{I,t})$$

Equation 3-31

The EM algorithm is adopted here to iteratively update the model parameters from their initial values $\theta_{I,t}^{0}$. However, as it can be seen in Equation 3-31, unlike the traditional EM algorithm,

where all model parameters are simultaneously updated, only the spatial parameters of the SCGMM models are updated in this phase, and the color parameters are kept unchanged. This can be implemented by constraining the color mean and variance to be fixed to their corresponding values in the previous frame (see Equation 3-28).

Such a restricted EM algorithm is shown below in Figure 7. In the E-step, the posteriori of the pixels belonging to each Gaussian component is calculated, and in the M-step, the mean and variance of each Gaussian component in spatial domain are refined based on the updated posteriori probability of pixel assignment from E-step. In the literature this EM algorithm is called Expectation Conditional Maximization (25).

After the EM process, the global SCGMM model of the image $\theta_{I,t}^{0}$ is split again into the foreground and background model $\theta_{fg,t}$, $\theta_{bg,t}$ as shown in Equation 3-32, maintaining the weights resulted from the EM step:

$$\theta_{fg,t} = \text{First } k_{fg} \text{Gaussians of } \theta_{I,t}$$
$$\theta_{bg,t} = \text{Last } k_{bg} \text{Gaussians of } \theta_{I,t}$$

Equation 3-32

Then, the weights of each model θ_{fg}, $\theta_{bg,}$ are normalized for model updating according to Equation 3-33:

$$w_{l,k} = \frac{w'_{l,k}}{\sum_{k=1}^{K_l} w'_{l,k}}$$

Equation 3-33

Where $l \in \{fg, bg\}$.

These resultant models will be used in the Energy minimization step.

Expectation Conditional Maximization

1.st E-step, calculate the Gaussian component assignment probability for each pixel ***z***:

$$p^{(i)}(k|z) = \frac{w'^{(i)}_k \cdot G\left(z_s;\ \mu^{(i)}_{k,s},\ \Sigma^{(i)}_{k,s}\right) \cdot G\left(z_c;\ \mu_{k,c},\ \Sigma_{k,c}\right)}{\sum_{k=1}^{K_I} w'^{(i)}_k\ G\left(z_s;\ \mu^{(i)}_{k,s},\ \Sigma^{(i)}_{k,s}\right) \cdot G\left(z_c;\ \mu_{k,c},\ \Sigma_{k,c}\right)}$$

2.nd M-step, update the spatial mean and variance, and the weight of each Gaussian component as:

$$\mu^{(i+1)}_{k,s} = \frac{\sum_{z\in I_t} p^{(i)}(k|z) \cdot z_s}{\sum_{z\in I_t} p^{(i)}(k|z)}$$

$$\Sigma^{(i+1)}_{k,s} = \frac{\sum_{z\in I_t} p^{(i)}(k|z) \cdot (z_s - \mu^{(i+1)}_{k,s}) \cdot (z_s - \mu^{(i+1)}_{k,s})^T}{\sum_{z\in I_t} p^{(i)}(k|z)}$$

$$w'^{(i+1)}_k = \frac{\sum_{z\in I_t} p^{(i)}(k|z)}{\sum_{k=1}^{K_I}\sum_{z\in I_t} p^{(i)}(k|z)}$$

Figure 7 Expectation Conditional Maximization algorithm for foreground/background joint tracking

Energy minimization

After the joint foreground/background model have been combined into a generative model of the image, the model has been updated using EM, and split back into foreground and background models, the segmentation problem is solved using energy minimization. At any time instant *t*, let the feature vectors extracted from the video pixels be $z_{i,t}, i = 1, \dots, P$, where P is the number of pixels in each frame. Denote the unknown label of each pixel as $f_{i,t}, i = 1, \dots, N$, where $f_{i,t}$ is a binary variable, with $f_{i,t} = 1$ representing pixel i labeled as foreground, and $f_{i,t} = 0$ as background. In the following discussions, we may ignore subscript t when it causes no confusion.

The energy–based function is formulated over the unknown labeling variables of every pixel, $f_{i,t}, i = 1, \dots, N$, in the form of a first-order Markov Random Field (MRF) energy function:

$$E(f) = E_{data}(f) + \lambda\, E_{smooth}(f) = \sum_{p\in P} D_p(f_p) + \lambda \sum_{\{p,q\}\in N} V_{p,q}(f_p, f_q)$$

Equation 3-34

Where N denotes the set of 8-connected pair-wise neighboring pixels, P is the set of pixels in each image. The role of λ is to balance the data $D_p(f_p)$ and smooth cost $V_{p,q}(f_p, f_q)$. The above energy function can be efficiently minimized by a two-way graph cut algorithm (Annex I.), where the two terminal nodes represent foreground and background labels.

The pair-wise smoothness energy term $E_{smooth}(f)$ is modeled as:

$$E_{smooth}(f) = \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) = \sum_{\{p,q\} \in N} \frac{1}{d(p,q)} \cdot e^{-\frac{(I_p - I_q)^2}{2\sigma^2}}$$

Equation 3-35

where I_p, I_q denote the intensity of pixel p and q respectively, σ is the average intensity difference between neighboring pixels in the image, and $d(p,q)$ is the distance between two pixels p and q. This smoothness constraint penalizes the labeling discontinuities of neighboring pixels if they have similar pixel intensities.

It favors the segmentation boundary along regions where strong edges are detected.

The data energy term $E_{data}(f)$ evaluates the posterior probability of each pixel belonging to the foreground or background. The posterior can be calculated according to Equation 3-25.

Given the SCGMM models, the data cost $E_{data}(f)$ is defined as:

$$E_{data}(f) = \sum_{p \in P} D_p(f_p) = \sum_{p \in P} -\log\, p(f_p | z_p)$$

Equation 3-36

Where $p(f_p|z_p)$ is computed using Equation 3-25.

Fg/Bg GMM Parameter Updating in the spatial domain

Given foreground and background pixels $I_{t,fg}$, $I_{t,bg}$, obtained from the Energy Minimization step, the objective is to obtain an updated parameter sets $\{w^*_{fg,k,t},\ \mu^*_{fg,k,s,t},\ \Sigma^*_{fg,k,s,t}\}$ and $\{w^*_{bg,k,t},\ \mu^*_{bg,k,s,t},\ \Sigma^*_{bg,k,s,t}\}$ over the spatial domain, which maximizes data likelihood of each $I_{t,fg}$, $I_{t,bg}$ image region:

$$\{w^*_{l,k,t},\ \mu^*_{l,k,s,t},\ \Sigma^*_{l,k,s,t}\} = arg \max_{w^*_{l,k,t},\ \mu^*_{l,k,s,t},\ \Sigma^*_{l,k,s,t}} \prod_{z_t \in I_{l,t}} p(z_t|\theta_{l,t})$$

Equation 3-37

Where $l \in \{fg, bg\}$.

The spatial domain mean and variances are updated applying Expectation Conditional Maximization algorithm (Figure 7) for each foreground and background models separately, forcing the color means and variances to be constant and using for each model $I_{t,fg}$, $I_{t,bg}$ respectively instead of all I_t pixels.

After the updating process, the workflow shown in Figure 6 is executed again for each frame, obtaining as a result the foreground segmentation of each frame of the sequence.

4 OUR APPROACHES

4.1 INTRODUCTION

In this Section we will detail the investigation and development work that it has been done in the foreground segmentation and tracking area to develop this project. We will detail three new different approaches to improve the foreground segmentation and tracking state of art.

- *Foreground segmentation and tracking via SCGMM for static and moving monocular video sequences.*
- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss combined models*.
- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking combined models.*

The first system is an improvement based in the SCGMM solution proposed by Yu *et al.* (1) explained in section 3.2.2.1.

The second and third systems propose to combine a region based probabilistic model (SCGMM) with a pixel-wise probabilistic model (1Gaussian) to achieve correct modeling of the foreground and background.

4.2 FOREGROUND SEGMENTATION AND TRACKING VIA SCGMM IN STATIC AND MOVING CAMERA SCENARIOS

We propose this method for *Foreground Segmentation and Tracking via SCGMM for Static and Moving Monocular Video Sequences* to segment and track foreground objects in all possible video sequence environments: static or moving camera, with all kind of object speed, orientation, scale, and rotation.

In the following lines, we expose the basis of this technique.

4.2.1 Characteristics

It makes objects foreground segmentation and tracking possible in moving and static camera sequences.
Robustness towards object scale, orientation and rotation changes.
Foreground and background modeled with SCGMM.
It doesn't allow real time analysis

Table 1 Foreground Segmentation and Tracking Via SCGMM Characteristics

4.2.2 Basis

After studying and testing the Joint tracking SCGMM method (1) (explained in section 3.2.2.1), we detected and analyzed its limitations with the objective of improving the system to adapt the algorithm for analyzing different kind of sequences other with moving camera. The main weaknesses are:

- Updating only spatial components
- High number of Gaussians to model the background
- High computational cost

Updating only spatial components

As can be read in the state of the art, Yu *et al.* (1) proposes to update both the foreground and background SCGMM models at each frame, but only with respect to the spatial domain of the model, assuming that the color components remain constant. In our case, with the aim of segmenting foreground objects in moving camera sequences, this assumption is not valid because these sequences present background changes every frame.

High number of gaussians to model the background

In a standard video sequence it is normal to observe a background with several regions according to the scene. The joint tracking SCGMM algorithm (1), analyzes all the background with a fixed number of Gaussian distributions, then, to achieve a correct probabilistic model, a high number of Gaussians are needed (theoretically one for each different region). This factor increases the computational cost to maximize the data likelihood via EM algorithm. Faster computations can be reached by reducing the number of gaussians, but this produces an incomplete background modeling and as consequence, an incorrect foreground segmentation with false positives. In Figure 8 this phenomenon can be observed. Two segmentations are shown: one defining ten gaussians to model the foreground and twenty gaussians to model the background, and another one using twenty and forty to model foregroundd/background respectively. As it can be appreciate, we need to use more Gaussian distributions for modeling better foregroundd and background regions avoiding false detections.

High computational cost

The SCGMM method proposes to analyze all pixels of the image, even in situations where the foreground objects take up only a small region. This increases, like the previous point, the computational cost used to maximize the data likelihood via EM algorithm.

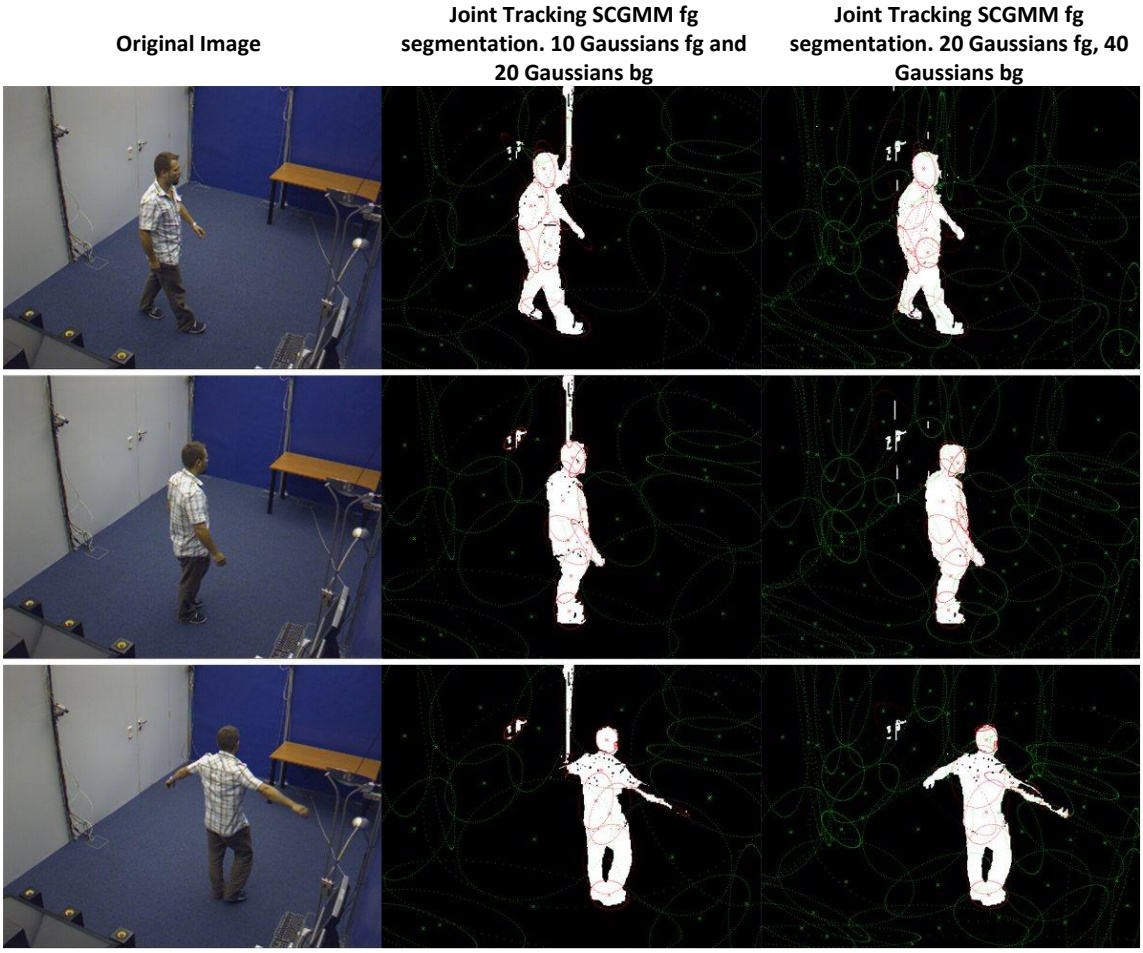


Figure 8 Joint Tracking Fg segmentation analysis according to the number of gaussian distributions.

In this way, the *Foreground Segmentation and Tracking via SCGMM for Static and Moving Monocular Video Sequences* that we propose, adds two main modifications to solve the problems in (1) and detailed above. These modifications allow us to segment and track objects in moving camera sequences by minimizing the computational cost:

- **Updating the background model with spatial and color information:**
  This modification allows us to adapt background SCGMM model with new background regions that appear in scene due to the camera motion.

- **Analyze only the nearest background region to the foreground object**
  With this strategy, the SCGMM background model works over small regions that appear near the object, without using a high number of gaussians and hence, minimizing the computational cost due to reduced amount of background pixels to analyze, and the low number of gaussians needed to model correctly the background.

4.2.2.1 Work Flow

In Figure 9 the workflow of our approach is shown, where it can be appreciated that the algorithm works only over the analysis region near the object to segment, it also maintains two different updates for foreground and background SCGMM.

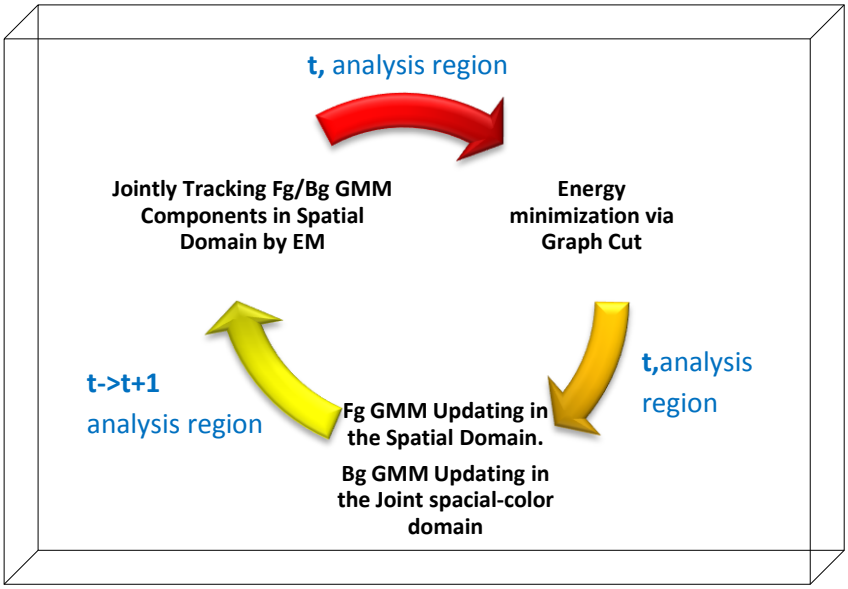


Figure 9 The Iterative Circle of Foreground/Background Segmentation for One Frame

Formally, *segmentation and tracking via SCGMM* can be described as follows:

SCGMM Joint Tracking

Given the pixel analysis region I' defined by the pixels of I inside the rectangular area with the following limits:

- Top foreground pixel + d'
- Bottom foreground pixel + d'
- Left foreground pixel + d'
- Right foreground pixel + d'

Where *d'* is a predefined size proportional to the object area that allows all possible movements of the object, so as to achieve a correct segmentation. A graphic example can be observed in Figure 10 where the foreground pixels are shown in white, I'_{fg}, and the background pixels in black (The background pixels inside the analysis rectangle area performs I'_{bg}).

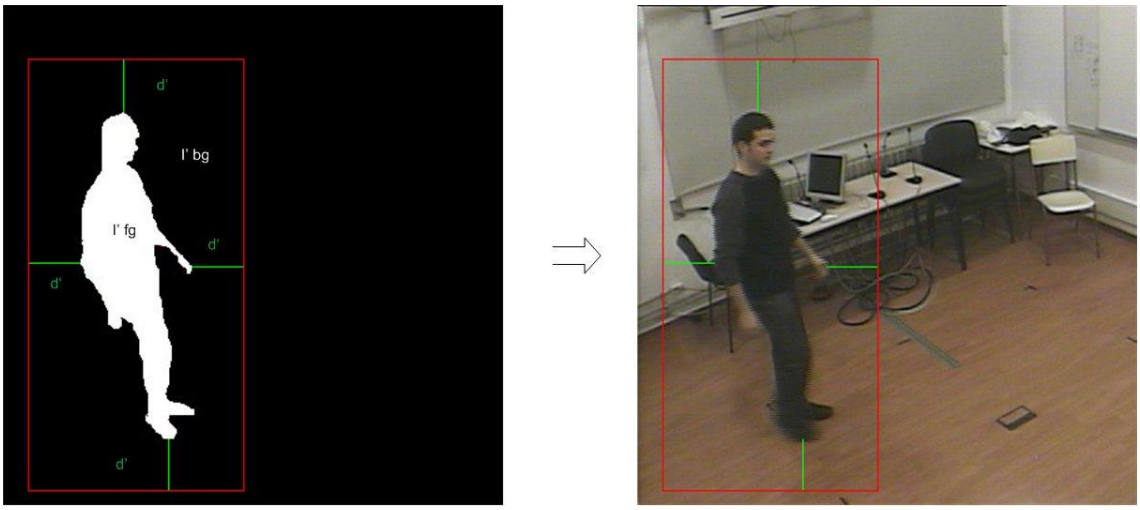


Figure 10 Example of I' region

Then, the algorithm runs as Section 3.2.2.1-SCGMM Joint Tracking shows, with some modifications:

Given a first classification into foreground and background pixels, two SCGMM models (θ) are learned during the system initialization period, in the first frame, by means of the EM algorithm according to Equation 3-27.

In this way, considering the same assumption as Equation 3-28, the formulation and updating scheme for the spatial parts $\mathrm{G}(\mathrm{z_s}; \mu_{\mathrm{l,k,s,t}}, \Sigma_{\mathrm{l,k,s,t}})$ (where s denotes the spatial features) given the new input region I_t' is the same as proposed in Equation 3-29, Equation 3-31 and Equation 3-32 but computed using the pixels of I_t' region instead of all the pixels of the image I_t. Also, the Expectation Conditional Maximization algorithm exposed in the state of the art (Figure 7, page 28) is adopted here to iteratively update the model parameters from their initial values.

Fg GMM Updating in the Spatial Domain. Bg GMM Updating in the Joint spacial-color domain

As it has been said before, we propose to update only the spatial components of the SCGMM foreground model, maintaining the same updating as is proposed in (1). However, for Background updating, we propose to update both spatial and color domain of the model as follows:

After the Graph cut minimization, the global SCGMM model of the image $\theta_{I',t}^{0}$ is split again into the foreground and background model θ_{fg}, $\theta_{bg,}$ according to Equation 3-32 The weights of each model are normalized according to Equation 3-33

Now we assume the following hypothesis for the background:

$$G(z_c; \mu_{bg,k,c,t}, \Sigma_{bg,k,c,t}) \neq G(z_c; \mu_{bg,k,c,t-1}, \Sigma_{bg,k,c,t-1})$$

Equation 4-1

The background updating process consists in executing the standard Expectation Maximization algorithm for background SCGMM model without forcing the color means and variances to be constant using the.background pixels, I'_{bg} as input data.

This algorithm can be observed in Figure 11:

Expectation Maximization for background update

1.st E-step, calculate the Gaussian component assignment probability for each pixel ***z***:

$$p^{(i)}(k|z) = \frac{w_k^{(i)} \cdot G\left(z_s;\ \mu_{k,s}^{(i)},\ \Sigma_{k,s}^{(i)}\right) \cdot G\left(z_c;\ \mu_{k,c}^{(i)},\ \Sigma_{k,c}^{(i)}\right)}{\sum_{k=1}^{K_{bg}} w_{k,t}^{(i)}\ G\left(z_s;\ \mu_{k,s}^{(i)},\ \Sigma_{k,s}^{(i)}\right) \cdot G\left(z_c;\ \mu_{k,c}^{(i)},\ \Sigma_{k,c}^{(i)}\right)}$$

2.nd M-step, update the spatial and color means and variances, and the weight of each Gaussian component as:

$$\mu_{k,s}^{(i+1)} = \frac{\sum_{z \in I'_{t,bg}} p^{(i)}(k|z) \cdot z}{\sum_{z \in I'_{t,bg}} p^{(i)}(k|z)}$$

$$\Sigma_{k,s}^{(i+1)} = \frac{\sum_{z \in I'_{t,bg}} p^{(i)}(k|z) \cdot (z - \mu_k^{(i+1)}) \cdot (z - \mu_k^{(i+1)})^T}{\sum_{z \in I'_{t,bg}} p^{(i)}(k|z)}$$

$$w_k^{(i+1)} = \frac{\sum_{z \in I'_{t,bg}} p^{(i)}(k|z)}{\sum_{k=1}^{K_{bg}} \sum_{z \in I'_{t,bg}} p^{(i)}(k|z)}$$

Where $I'_{t,bg}$ denoted all the background pixels inside the I′ region.

Figure 11 EM algorithm for background spatial and color domains update

Energy Minimization

The Energy minimization step for the I' region will be the same as proposed by Yu et al in (1), and explained in the State of the art section 3.2.2.1

4.2.3 Implementation overview

The algorithm needs one frame (the first one of the sequence) with the correct foreground segmentation mask of the object we want to segment, to initialize both foreground and background SCGMM models. The implementation steps are shown next:

1. **Define the number of Gaussian distributions for foreground and background models and specify the d' distance.**

   Common values are: ten or twenty gaussians for each model depending on the number of color regions that the object and the background have. And $d' = object\ width/2$.

2. **Initialize both foreground and background models with the first frame and its foreground object mask:**

   - Execute EM algorithm for each model (θ_{fg}, θ_{bg}) with its corresponding pixels ($I'_{0,fg}, I'_{0,bg}$) to initialize color (r, g, b) and spatial mean and variance of each Gaussian distribution.

3. **For the next frames t>1:**

   - Combine both foreground, background SCGMM models (θ_{fg}, θ_{bg}) into one I'_t model ($\theta_{I'}$):

   $$\theta_{I',t} \stackrel{\text{def}}{=} \{\theta_{fg,t-1}, \theta_{bg,t-1}, \gamma_{fg,t-1}, \gamma_{bg,t-1}\}$$

   Where the k_{fg} foreground gaussians are placed in first position and the k_{bg} background gaussians are placed in the last positions, and:

   $$\gamma_{l,t-1} = \frac{\#I'_{t,l}}{\#I'_t}$$

   Where $l \in \{fg, bg\}$, # denotes the cardinality operator and can be understood as the region area.

   Then, these coverage percentages are used to update the gaussians weights:

   - Execute **Expectation Conditional Maximization** [Figure 7] algorithm for $\theta_{I'_t}$ model with its corresponding pixels I'_t to update spatial mean and variance of each Gaussian distribution.

     The weights of the Gaussian distributions of each model are normalized as follows:

     $$w_{l,k} = \frac{w_{l,k}}{\sum_{k=1}^{K_l} w_{l,k}}$$

     Where $l \in \{fg, bg\}$.

   - Apply Energy Minimization Graph cut algorithm [section 3.2.2.1]

     A common value for λ parameter is 200.
     Thus, we obtain the foreground segmentation mask for frame t, that is $I'_{fg,t}$ and $I'_{bg,t}$

- Separate I'_t SCGMM model ($\theta_{I',t}$) into both SCGMM models $\theta_{fg,t}$, $\theta_{bg,t}$:

  $\theta_{fg,t}$ =First k_{fg}Gaussians of $\theta_{I',t}$.

  $\theta_{bg,t}$ =Last k_{bg}Gaussians of $\theta_{I',t}$.

- Foreground-Background update:
  - Update foreground model $\theta_{fg,t}$ applying **Expectation Conditional Maximization** algorithm (Figure 7) for $\theta_{fg,t}$ model with its corresponding pixels $I'_{fg,t}$ to update the spatial mean and variance of each foreground Gaussian distribution.
  - Update background model $\theta_{bg,t}$ applying **Expectation Maximization** algorithm (Figure 11) for $\theta_{bg,t}$ model with its corresponding pixels $I'_{bg,t}$ to update the spatial and color (r,g,b) mean and variance of each background Gaussian distribution.

- Analysis region I' update:

  Detect the largest foreground connected component (object to segment) from the foreground segmentation mask.

  Apply the *d'* distance to the top, bottom, left and right foreground pixel to create the I'_{t+1} analysis area.

4.2.4 Results

In this section we show some segmentation results obtained with the foreground segmentation and tracking proposed method via SCGMM in static and moving camera scenarios. For this purpose, several video sequences with different difficulty degree will be analyzed.

Figure 12 Soccer player foreground segmentation.

Figure 12 shows the segmentation of a soccer player. A correct segmentation is obtained thanks to the correct foreground/background modeling due to the color difference between foreground and background regions.

In Figure 13 the foreground segmentation of an skier can be observed. The segmentation results shows a correct definition of the foreground object under light background changes that are correctly modeled by the background model.

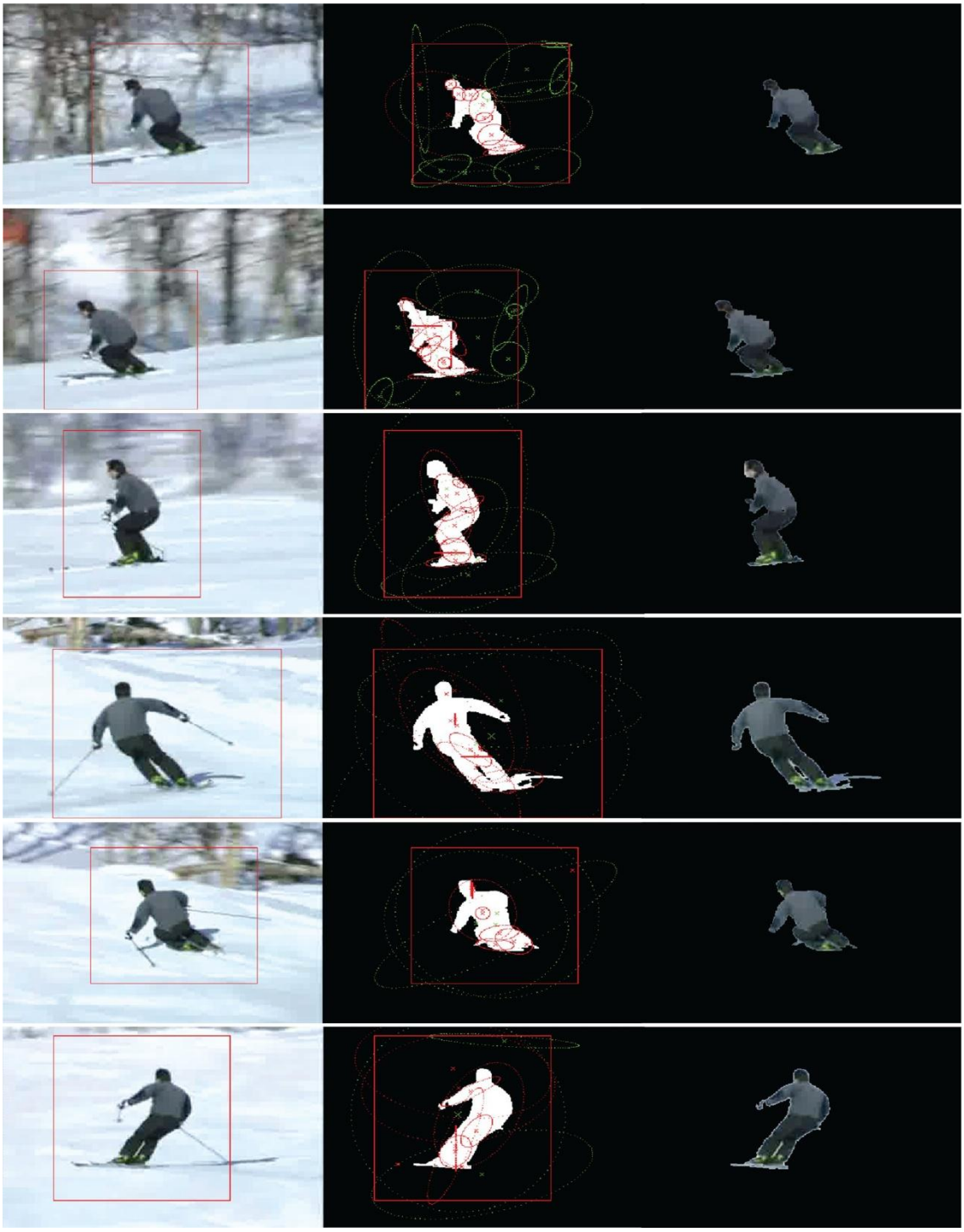


Figure 13 Foreground segmentation results. Skier sequence.

Figure 14 F1 car foreground segmentation 10 Gaussians model the foreground and 10 Gaussians model the background. This is a sequence with camera motion, object movement and scale change

Figure 14 shows some results in a sequence that presents special difficulty due to camera motion, object movement and 3D object rotation (resulting in a change in point of view), and camera zoom. It can be observed how thanks to background model color updating, new background regions that appear in each frame, are incorporated into the background model allowing a good enough object segmentation.

The object scale and point of view changes make difficult the adaptation of the foreground SCGMM model to the new situation (both in color and spatial domain) and may produce false negatives that will be propagated in the following frames. The same problem appears in the background model when region I_t' increases its size, generating false positives detections.

This kind of problems are common in several sequences because we are fixing the color foreground model to be constant. Propagation of these segmentation problems along the sequence will force us to improve the foreground color model updating as a future work. An example of this problem can be observed in Figure 15:

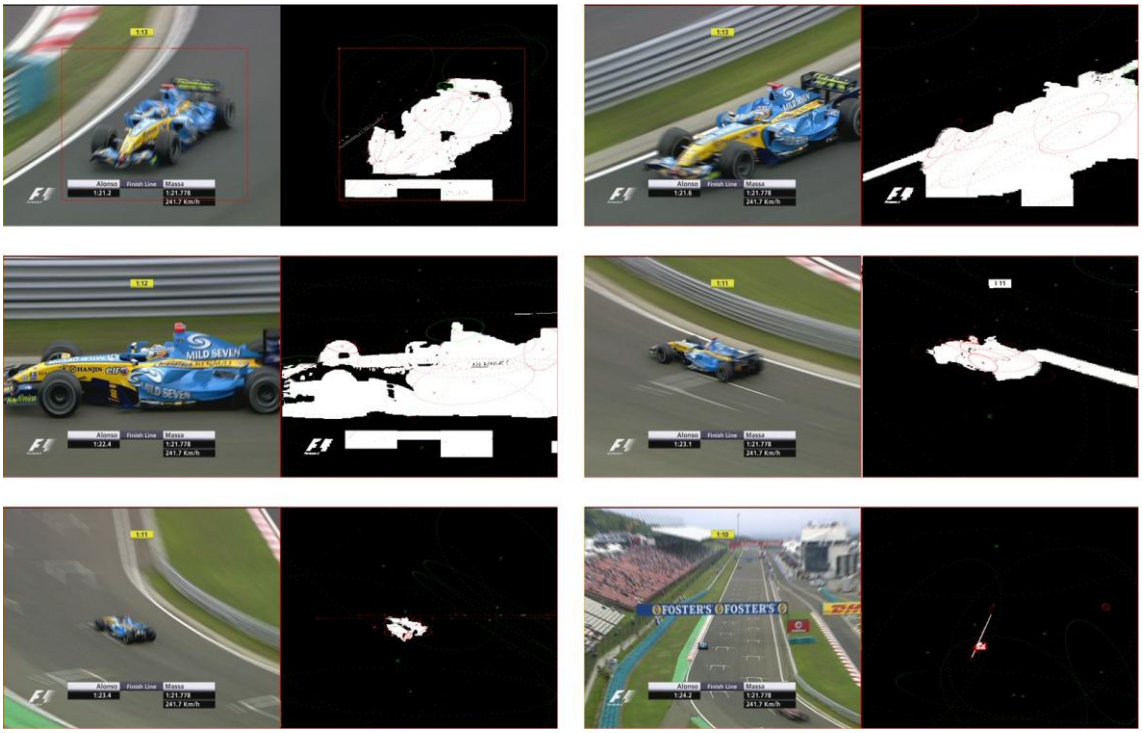


Figure 15 F1 car foreground segmentation. False positives and negatives due to bad model adaptation.

As it can be observed, when the object scale increases, 10 gaussians are not enough to model all foreground regions, neither, to model the background regions, forcing false positives and false negatives detections.

Another result with a F1 sequence is presented in Figure 16, where the segmented object occludes another object in some of the frames. In this sequence, the color similarity between the object to segment and the background regions is the main challenge

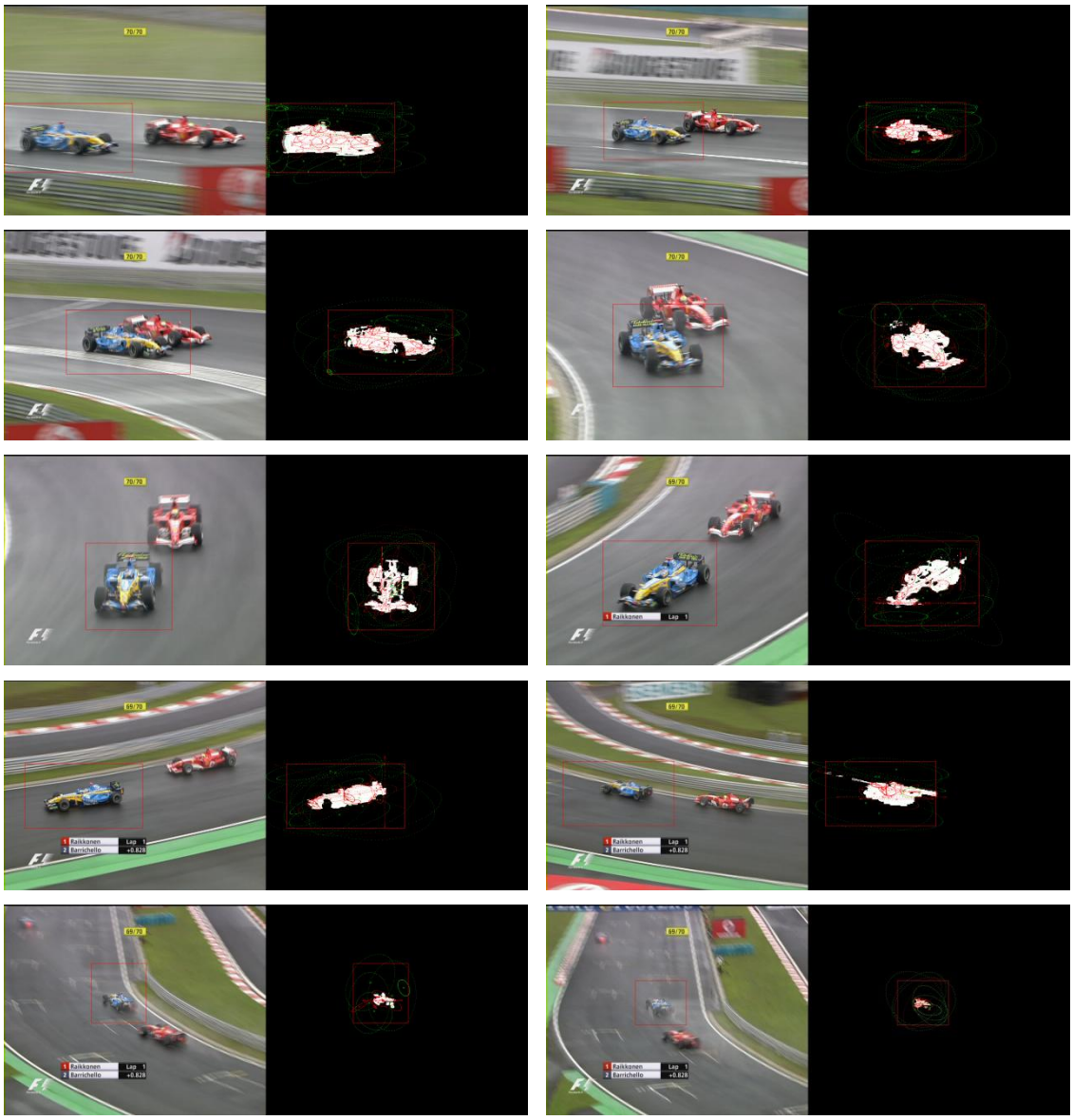


Figure 16 F1 car sequence 2. Camera motion, object rotation, occlusion and position, scale changes.

As it can be observed, we obtain a good segmentation, but some false negatives in the foreground region appear because of the similarity between some regions of the objects (in this case, the wheels). Foreground and background models rival each other to model these regions and eventually produce false positives and false negatives detections.

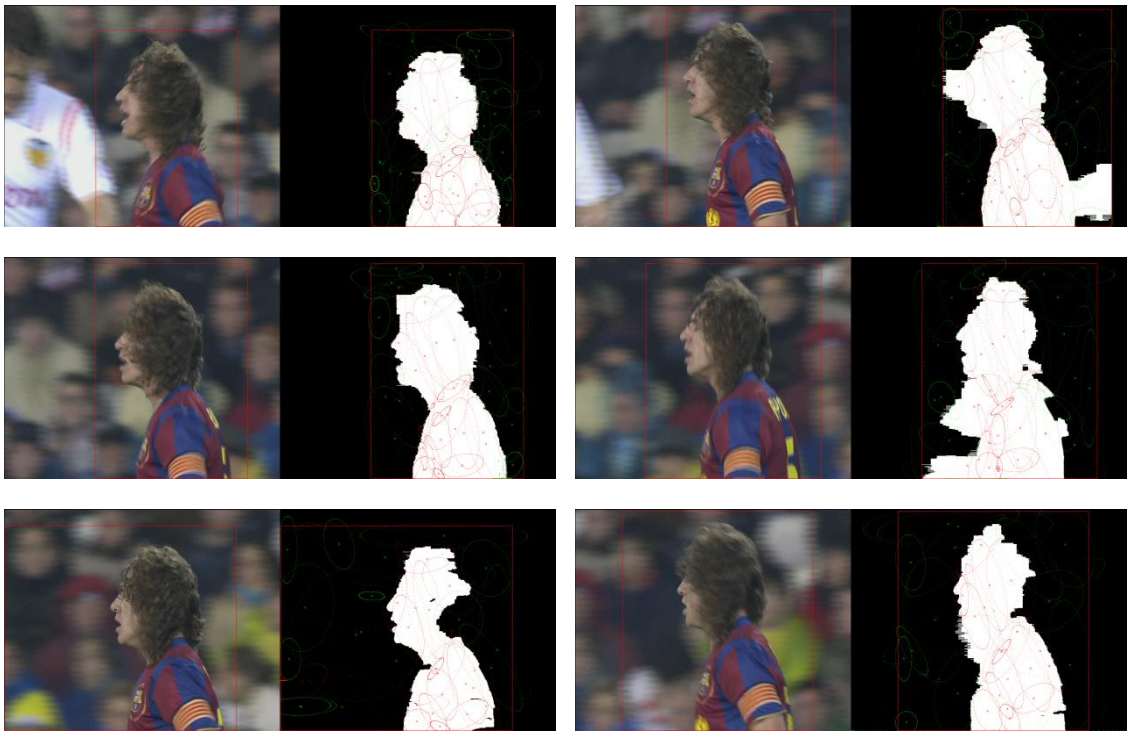


Figure 17 Soccer player foreground segmentation 1. Multi-region background and camera movement

Figure 17 shows a difficult scene due to the similarity between foreground and background regions which cause some false positives and false negatives in the boundaries of both regions. The swift background changes between frames, and the amount number of regions origins that the background model is not been able to model all regions correctly.

4.2.4.1 Conclusions

The *foreground segmentation and tracking via SCGMM* algorithm presents a correct object segmentation in static and moving camera sequences when there are clear differences between the object to segment and background regions. When there are object rotations or scale changes, our assumption of foreground object color invariance is not true, and same false positives and false negatives segmentation problems appear that could propagate it over the next frames. Also, in these cases, the models initialized in the first frame may not be enough to model all color-space regions, generating more false positives and false negatives.

This approach that we propose, can be an interesting research line for the future because of the possibilities for segmenting objects that it offers.

4.3 FOREGROUND SEGMENTATION IN MONOCULAR STATIC SEQUENCES VIA SCGMM-1GAUSS COMBINED MODELS.

In this approach we propose to improve the State of the Art in foreground segmentation for monocular static sequences by combining a color–spatial probabilistic model (SCGMM) to model the foreground object to segment, with a pixel-wise color model for background probabilistic modeling (1 Gauss/pixel modeling).

The so popular pixel-wise segmentation methods like (6) (8) try to create a probabilistic background model by means of modeling each pixel individually with a color Gaussian distribution. In these methods, no foreground prior modeling is used. Then, a pixel will be considered as foreground if the input value doesn't match the background model. The main characteristics of these methods are:

- Good definition in foreground objects detection (due to pixel model and pixel decision)
- False negative detections in foreground- background color similarity situations.

In this way, it is important to highlight that pixel-wise methods provide a good foreground segmentation when the objects to segment have different color than the background pixels they are occupying. Otherwise, false negatives will appear in the foreground segmentation of these regions, which is common in a non controlled environment.

The *Joint Tracking SCGMM* algorithm (1), proposes to model both the foreground and background with a Spatial-Color Gaussian Mixture Model. Both SCGMM models rival each other to model the pixels of the image. The main characteristics of this system are:

- Correct foreground segmentation when foreground object and background regions that the foreground occludes have different colors.
- In regions with foreground-background color similarity:
  - It improves the foreground segmentation obtained with common pixel-wise methods, reducing false negatives due to the object prior probability.
  - Less definition of foreground segmentation because both foreground and background SCGMM models try to model regions of the image (instead of individual pixels). This modeling may produce detection errors at small regions when there are few gaussians in the background or foreground model. For example, a black door handle in a white door may be detected as foreground if a person dressed in black is close to the door and the amount of background gaussians is not sufficient to place a Gaussian on the door handle.

A comparison between both methods can be observed in Figure 18, this is a difficult scene for foreground segmentation due to color similarity between foreground and background regions. In this scene, there is one foreground object to segment (the person).

In the SCGMM method some false negatives appear but also false detections because we are using both foreground and background model, and some parts of the background are not accurately modeled by the background model. Notice also how sensible it is to error propagation when the models are updated in each frame.

In 1Gauss pixel-wise foreground segmentation some false negatives appear because we are not using a probabilistic model for the object, and the color similarity between foreground and background makes that these pixels were associated to the background model. Also some false positives appear around the legs due to the shadows that the person projects in the ground regions. These false positives, originated by the shadows, can be removed with some techniques as Landabaso *et al.* propose in (26). We decide to examine the results without shadow removal to evaluate also our proposals in front of this common foreground segmentation problem.

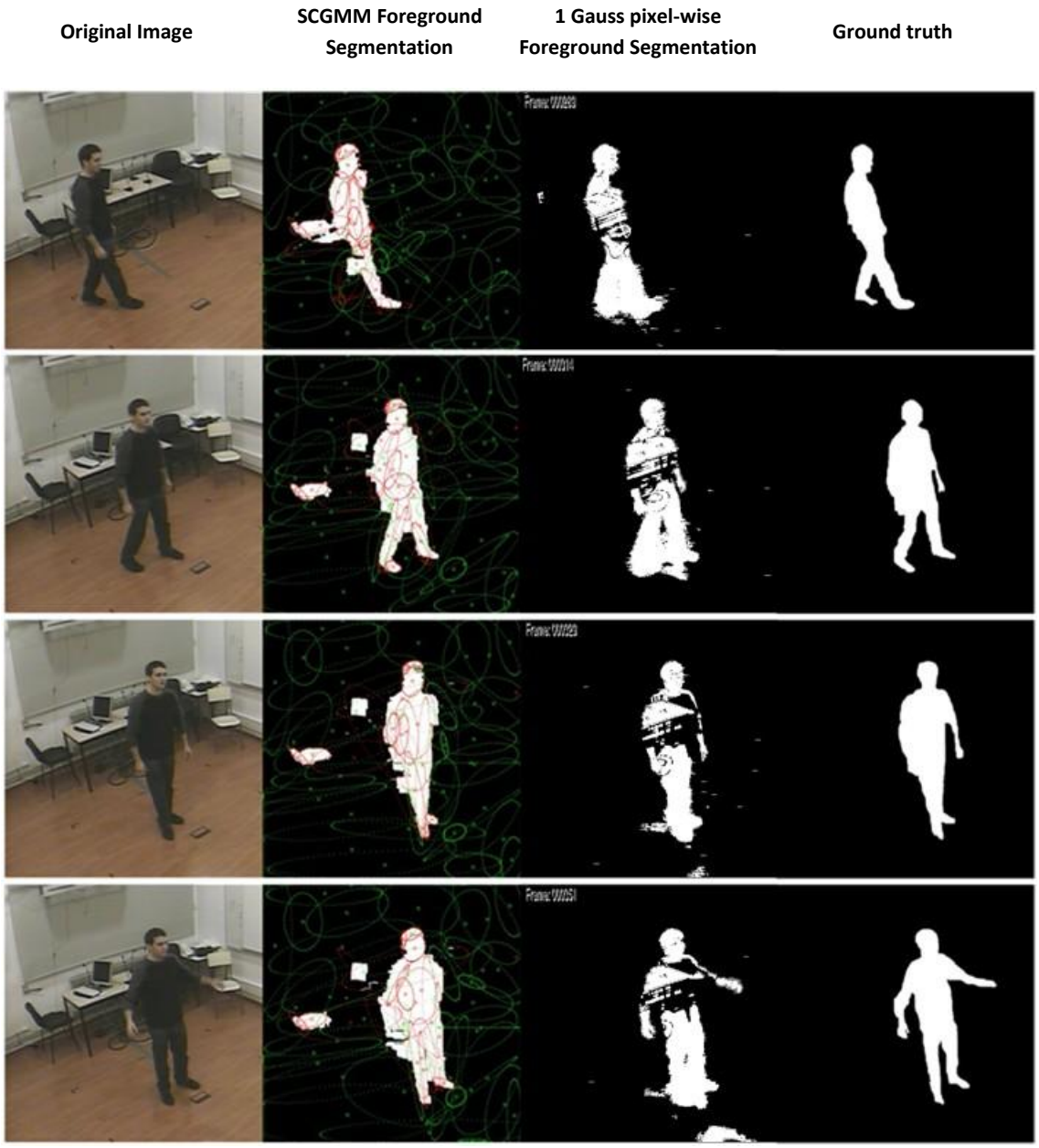


Figure 18 Foreground Segmentation comparison beween SCGMM region based system (in green background SCGMM, in red foreground SCGMM) and 1Gauss pixel-wise based system.

Thus, to achieve a correct foreground segmentation, (also in difficult scenes as we have seen in Figure 18), we propose two foreground segmentation methods that combine a pixel-wise and SCGMM models, taking the main advantages of each method. These methods are:

- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss combined models*.
- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking combined models.*

4.3.1 Foreground segmentation in monocular static sequences via SCGMM-1Gaussian combined models

In this section we explain the bases of this approach to improve the foreground segmentation in static sequences. The characteristics, work flow, implementation and results will be explained in the following lines.

4.3.1.1 Characteristics

Foreground objects segmentation for static monocular camera configuration.
Use SCGMM to model the foreground.
Use 1 Gaussian per pixel to model the background color.
It doesn't allow real time analysis

Table 2 Foreground Segmentation in Monocular Static Sequences VIa SCGMM-1Gauss Combined Models

4.3.1.2 Bases

We propose this system to improve the segmentation results for static sequences, obtained by combining pixel-wise methods (6) (8) and region based methods (1) (24). In this way we propose the following system that combines both models (Region Based model for foreground and pixel-wise for background). The reason for this choice is that the background is more stationary and it usually can be learned from frames with no foreground. Thus, it is possible to build an accurate model (pixel-wise) for it. However, the foreground is constantly changing and then a region model is more appropriate and robust to its changes.

4.3.1.3 Work Flow

The work Flow is showed in Figure 19:

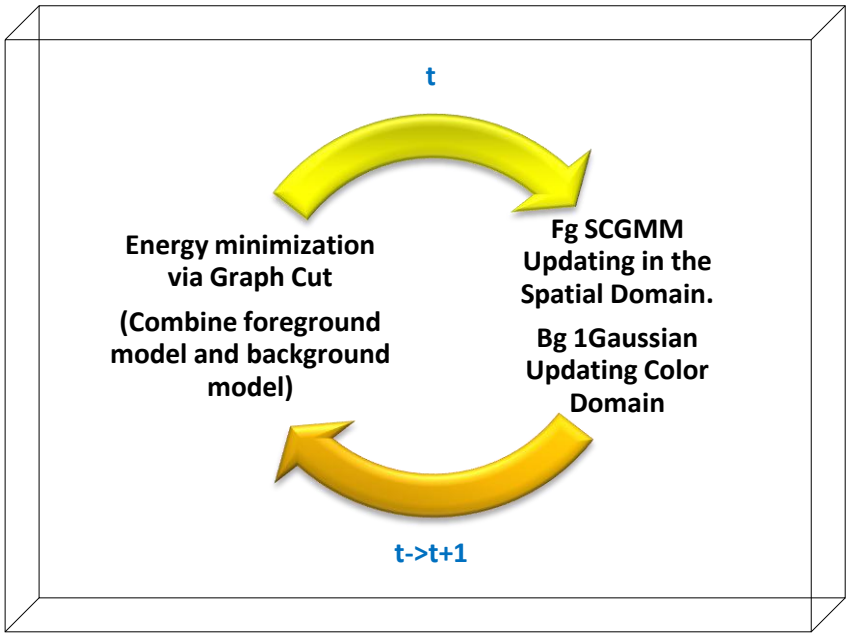


Figure 19 Foreground segmentation in monocular static sequences via SCGMM-1Gauss combined models Work Flow

In this system, we are using two different probabilistic models, hence the initialization for each one will be different:

- **Foreground SCGMM:** The First frame of the sequence is used to initialize the foreground model (color-space) by means of the EM algorithm (23). The input observations will be all the foreground object pixels $I_{0,fg}$. Hence, a initial foreground segmentation is needed.
- **Background 1Gauss/pixel:** To initialize this probabilistic model, a training sequence is needed (a few number of frames without foreground objects) that allows the Gaussian's color means and variances initialization for all pixels of the image I_0.

After the initialization is done, the algorithm operates as can be observed in Figure 19:

In the first step Energy Minimization with Graph Cuts using foreground and background models is used to decide if each pixel of the image is foreground or background. Here it is assumed that from time $t-1$ to t, the object maintain its spatial position since the object movement is small between frames, which is common in a standard frame rate sequence. Thus, we will use the model from the previous frame $t-1$ to detect foreground regions in the current one (t).

The second step consists in updating both foreground and background models, the first one in spatial domain and the second one in color domain, taking advantage of the previous step segmentation using only the pixels of each class to update each model.

Formally, the algorithm can be described as follows:

Given one SCGMM foreground model defined by a set of parameter (θ_{fg}) learned during the system initialization period, in the first frame, we use the EM algorithm to maximize the data *Likelihood* of the foreground:

$$\theta_{fg} \stackrel{\text{def}}{=} \{w_{fg,k},\ \mu_{fg,k},\ \Sigma_{fg,\mathrm{k}}\} = \arg \max_{w_{fg,k},\ \mu_{fg,k},\ \Sigma_{fg,\mathrm{k}}} \prod_{z_{fg} \in I_{fg,0}} \left[\sum_{k=1}^{K_{fg}} w_{fg,k}\ \mathrm{G}(\mathrm{z_{fg}}; \mu_{\mathrm{fg,k}}, \Sigma_{\mathrm{fg,k}}) \right]$$

Equation 4-2

z_{fg} are the five dimensional features of foreground pixels (r, g, b, x, y); I_0 denotes the initialization frame, $w_{fg,k}$ is the weight of the foreground Gaussian k, $\mu_{fg,k}$ its mean and $\Sigma_{fg,k}$ its covariance matrix.

Thus, we define:

$$p(z|fg) = \sum_{k=1}^{K_{fg}} w_{fg,k} G(z; \mu_{fg,k}, \Sigma_{fg,k})$$

Equation 4-3

where $w_{fg,k}$ is the weight of the k_{th} Gaussian component in the mixture model, z is the input pixel (r, g, b, x, y) that can be split into $z_c = (r, g, b)$ and $z_s = (x, y)$ and $G(z; \mu_{fg,k}, \Sigma_{fg,k})$ is the k_{th} Gaussian component:

$$G(z; \mu_{fg,k}, \Sigma_{fg,k}) = \frac{1}{(2\pi)^{\mathrm{d}/2} \cdot \left|\Sigma_{\mathrm{fg},k}\right|^{1/2}} \cdot \mathrm{e}^{-\frac{1}{2}\left[(\mathrm{z}-\mu_{fg,k})^{\mathrm{T}}\, \Sigma_{fg,k}^{-1}\, (\mathrm{z}-\mu_{fg,k})\right]}$$

Equation 4-4

Where $\mathrm{d} = 5$ is the dimension of the SCGMM models.

Assuming that the spatial and color components of the SCGMM models are decoupled, i.e., the covariance matrix of each Gaussian component takes the block diagonal form,

$\Sigma_{l,k} = \begin{pmatrix} \Sigma_{l,k,s} & 0 \\ 0 & \Sigma_{l,k,c} \end{pmatrix}$, with such decomposition, each GMM Gaussian component has the following factorized form:

$$\mathrm{G}\left(\mathrm{z};\mu_{\mathrm{l,k}},\Sigma_{\mathrm{l,k}}\right) = \mathrm{G}\left(\mathrm{z_s};\mu_{\mathrm{l,k,s}},\Sigma_{\mathrm{l,k,s}}\right)\cdot\mathrm{G}\left(\mathrm{z_c};\mu_{\mathrm{l,k,c}},\Sigma_{\mathrm{l,k,c}}\right)$$

Equation 4-5

Regarding the background 1Gauss/pixel background model (θ_{bg}) where the model defines one Gaussian per pixel ($K_{bg} = \#I_0$ gaussians), it is learned during the system initialization using the training sequence, to maximize the data *Likelihood* of the background as it is explained in section 3.1.1.2. That is, the mean and variance of each pixel computed indepently.

We formulate the 3-dimensional (r, g, b) pixel-wise model of the background proposed in (6) as a 5-dimensional (r, g, b, x, y) background model for the overall image, to make the formulation consistent for comparing both foreground and background models in a 5-dimensional Bayesian probabilistic framework. Thus, we define

$$p(z_i|bg) = \sum_{k=1}^{K_{bg}} w_{bg}\cdot G\left(z_{c,i};\mu_{bg,c,k},\Sigma_{bg,c,k}\right)\cdot\delta(i-k) =$$

$$= \sum_{k=1}^{K_{bg}} \frac{1}{\#I_0}\cdot G\left(z_{c,i};\mu_{bg,c,k},\Sigma_{bg,c,k}\right)\cdot\delta(i-k)$$

Equation 4-6

Where c denotes the three dimensional color features (r, g, b). $z_{c,i} \in I_0$ are the color features of pixel i. w_{bg} is the weight of the K_{th} Gaussian of the background model, # denotes cardinality and $\#I_0$ is understood as the image area (total amount of pixels in the image). $G(z_{c,i};\mu_{bg,k},\Sigma_{bg,k})$ is the Gaussian distribution that models the i_{th} pixel as:

$$G\left(z_i;\mu_{\mathrm{bg,i}},\Sigma_{\mathrm{bg,i}}\right) = \frac{1}{(2\pi)^{\mathrm{d}/2}\cdot\left|\Sigma_{\mathrm{bg,i}}\right|^{1/2}}\cdot \mathrm{e}^{-\frac{1}{2}\left[\left(\mathrm{z_{c,i}}-\mu_{\mathrm{bg,i}}\right)^{\mathrm{T}}\Sigma_{\mathrm{bg,i}}^{-1}\left(\mathrm{z_{c,i}}-\mu_{\mathrm{bg,i}}\right)\right]}$$

Equation 4-7

Where $\mathrm{d} = 3$ is the dimension of the SCGMM models.

Notice that the delta function in spatial domain in Equation 4-6 is like having a spatial Gaussian with zero variance.

The *energy minimization* and *updating* steps of the work flow will be formulated in next subsections.

Energy minimization

The foreground/background segmentation problem is solved, like the system proposed in (1) and explained in Section 3.2.2.1-Energy minimization, using energy minimization via Graph Cuts.

In this way, the energy–based function is formulated over the unknown labeling variables of every pixel, $f_{i,t}$, $i = 1, \dots, N$, in the form of a first-order Markov random field (MRF) energy function:

$$E(f) = E_{data}(f) + \lambda\, E_{smooth}(f) \;=\; \sum_{p \in P} D_p(f_p) + \;\lambda \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q)$$

Equation 4-8

The pair-wise smoothness energy term $E_{smooth}(f)$ is modeled as:

$$E_{smooth}(f) = \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) = \sum_{\{p,q\} \in N} \frac{1}{d(p,q)} \cdot e^{\frac{(I_p - I_q)^2}{2\sigma^2}}$$

Equation 4-9

The data energy term $E_{data}(f)$ evaluates the *posterior* of each pixel belonging to the foreground or background.

- The foreground posterior distribution is found via Bayes development. The posterior distribution of a class given the pixel in analysis z can be written as

$$p(fg|z) = \frac{p(z|fg) \cdot\, p(fg)}{p(z)}$$

Equation 4-10

Where the prior

$$p(fg) = \gamma_{fg,t-1}$$

Equation 4-11

is the coverage area of foreground from the previous frame. Notice that $\gamma_{fg,t-1} + \gamma_{bg,t-1} = 1$.

- The background posterior is:

$$p(bg|z_i) = \frac{p(z_i|bg) \cdot\, p(bg)}{p(z_i)}$$

Equation 4-12

where

$$p(bg) = \gamma_{bg,t-1}$$

Equation 4-13

is the prior probability (the background coverage area in the previous frame), z_i are the features of the i_{th} pixel.

Given the SCGMM foreground model $\theta_{fg,t}$, and the 1Gaussian/pixel background model $\theta_{bg,t}$, our aim is to comparable both models for pixel classification via Energy minimization.

Finally, both fg/bg models can be used in the energy minimization data cost $E_{data}(f)$ defined as:

$$E_{data}(f) = \sum_{p \in P} D_p(f_p) = \sum_{p \in P} -\log p(f_p|z_p)$$

Equation 4-14

Where $p(f_p|z_p)$ is computing using Equation 4-2 and Equation 4-12 for foreground and background analysis respectively.

Fg SCGMM Updating in the Spatial Domain

We assume that from time $t-1$ to t, the colors of the foreground objects do not change. Hence, the color parts of the SCGMM models remain identical:

$$\mathrm{G}\left(\mathrm{z_c};\mu_{\mathrm{l,k,c,t}},\Sigma_{\mathrm{l,k,c,t}}\right)=\ \mathrm{G}\left(\mathrm{z_c};\mu_{\mathrm{l,k,c,t-1}},\Sigma_{\mathrm{l,k,c,t-1}}\right)$$

Equation 4-15

where c denotes the color dimension, $k=1,\ldots,K_I$ the Gaussian distribution number and $\mathrm{z_c}$ the color pixel information.

Then, the updating process will update only the gaussians spatial dimension of the SCGMM model. For this purpose, Expectation Conditional Maximization (25) is used taking as data input, all foreground pixels detected in previous Energy minimization via Graph cuts step:

Expectation Conditional Maximization

1.st E-step, calculate the Gaussian component assignment probability for each pixel ***z***:

$$p^{(i)}(k|z)=\frac{w_k^{(i)}\cdot G\left(z_{\mathrm{s}};\ \mu_{k,s}^{(i)},\ \Sigma_{\mathrm{k,s}}^{(\mathrm{i})}\right)\cdot G\left(z_{\mathrm{c}};\ \mu_{k,c},\ \Sigma_{k,c}\right)}{\sum_{k=1}^{K_{fg}}w_k^{(i)}\ G\left(z_{\mathrm{s}};\ \mu_{k,s}^{(i)},\ \Sigma_{\mathrm{k,s}}^{(\mathrm{i})}\right)\cdot G\left(z_{\mathrm{c}};\ \mu_{k,c},\ \Sigma_{k,c}\right)}$$

2.nd M-step, update the spatial mean and variance, and the weight of each Gaussian component as:

$$\mu_{k,s}^{(i+1)}=\frac{\sum_{z\in I_{t,fg}}p^{(i)}(k|z)\cdot z_s}{\sum_{z\in I_{t,fg}}p^{(i)}(k|z)}$$

$$\Sigma_{k,s}^{(i+1)}=\frac{\sum_{z\in I_{t,fg}}p^{(i)}(k|z)\cdot(z_s-\ \mu_{k,s}^{(i+1)})\cdot(z_s-\ \mu_{k,s}^{(i+1)})^T}{\sum_{z\in I_{t,fg}}p^{(i)}(k|z)}$$

$$w_k^{(i+1)}=\frac{\sum_{z\in I_{t,fg}}p^{(i)}(k|z)}{\sum_{k=1}^{K_{fg}}\sum_{z\in I_{t,fg}}p^{(i)}(k|z)}$$

Where $I_{t,fg}$ denotes all pixels detected as foreground

Figure 20 Expectation Conditional Maximization for foreground updating

Background 1Gaussian Pixel Model Updating Color Domain

As it can be observed in pixel-wise foreground segmentation methods (6) (8) (State of the Art Section 3), the Background model has a color update of these Gaussians where the input value matches the background model. This update is important in outdoor scenes where progressive illumination changes occur, but also, it could be a problem in scenes where the foreground segmentation presents false negatives, generating an evolution of the model towards the foreground input value. To make our system robust, in our approach we maintain this background update, but taking as reference the foreground mask obtained from the previous step (*Energy minimization via Graph cuts*). Hence, the background updating process will be consistent with the foreground segmentation we are obtaining with the previous step.

Then, our algorithm will update the background model, **updating only the Gaussians that model pixels detected as background**. Background updating equations for the Gaussian that models these pixels ($z_{c,k}$) are:

$$\mu_{k,t} = \rho z_{c,k} + (1-\rho)\mu_{k,t-1}$$

$$\sigma_{k,t}^2 = \rho \cdot d^2 + (1-\rho)\sigma_{k,t-1}^2$$

$$if \quad \sigma_{k,t}^2 < \sigma_{th}^2 \quad \rightarrow \quad \sigma_{k,t}^2 = \sigma_{th}^2$$

$$(d = (z_c - \mu_{k,t})^2)$$

Equation 4-16

Where t denotes the frame time, $z_{c,k}$ is the color value of the K_{th} pixel detected as background, $\mu_{k,t,}$ is the mean of the Gaussian that models the K_{th} pixel, $\sigma_{k,t}^2$ is the variance, ρ is a weight that defines the update speed (commonly ρ=0.01) and d is the Euclidean distance between the mean and the input value of the pixel.

Notice also, that the variance has a minimum threshold to avoid very small values that can produce some spurious foreground detections when the background color is very stable. Common value for σ_{th}^2 in (r, g, b) 8 bit/channel domain is 20.

4.3.1.4 Implementation Overview

The algorithm needs one frame (the first one of the sequence) with the correct foreground segmentation mask of the object we want to segment, to initialize foreground SCGMM model. Also, it needs a training sequence (about 50 frames) without foreground objects to initialize the background model. The implementation steps are shown next, continuing along:

1- Define the number of Gaussian distributions for foreground model

Common values are: ten or twenty gaussians for each model depending on the number of color regions that the object and the background have.

2- $t = 1$ Initialize both foreground and background models.

- Foreground: using the first frame $t = 1$ foreground pixels $(I_{0,fg})$ executing the EM algorithm for the SCGMM model (θ_{fg}) to initialize color (r, g, b) and position (x, y) of each Gaussian distribution.
- Background: using the training sequence:
  - For $t_{training} = 1$, center all Gaussians means to the input value $z_{c,i}$

$$\mu_{t=1} = z_c$$

  - For $1 < t_{traning} < t_{training\ end}$, update mean and variance with the updating equations: Equation 4-16

3- For next frames $t > 1$:

- Apply Energy Minimization Graph cut algorithm for each pixel (Section 4.3.1-Energy minimization page 53).

  Common value for λ is 200.
  Result: Foreground segmentation mask, obtaining $I_{fg,t}$ and $I_{bg,t}$

- Foreground SCGMM spatial domain updating: Execute the **Expectation Conditional Maximization** (Figure 20) algorithm for θ_{fg} model with its corresponding pixels $I_{fg,t}$ to update position (x, y) of each Gaussian distribution.

- Background 1 Gaussian/pixel color domain updating: Use the updating equations Equation 4-16 to update all Gaussians $\theta_{bg,i}$ that model the pixels grouped in $I_{bg,t}$.

4.3.1.5 Results

Now we show some segmentation results obtained with Foreground segmentation in monocular static sequences via SCGMM-1Gauss combined models in smart room static camera scenarios. For this purpose, several video sequences will be analyzed.

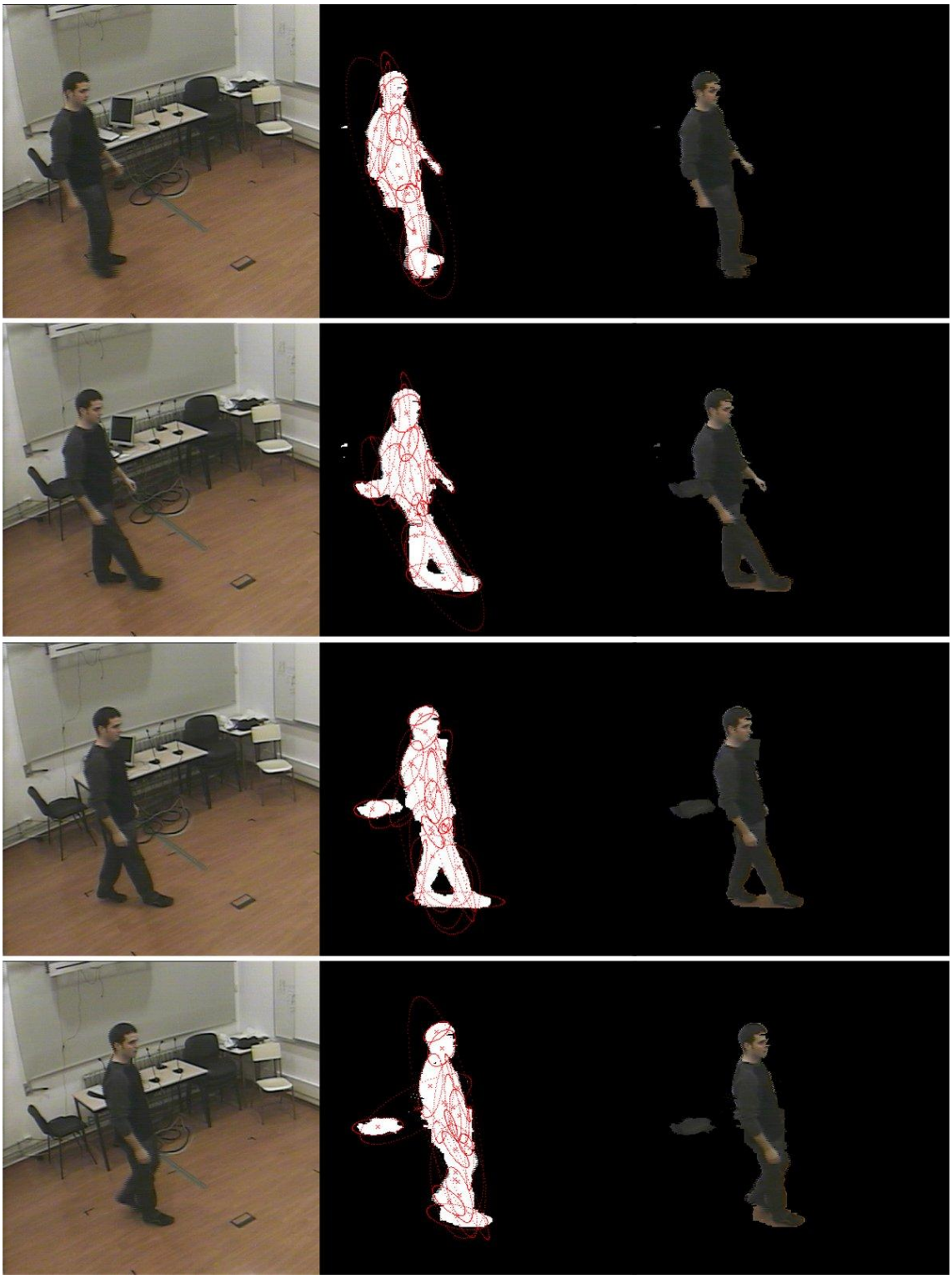


Figure 21 SCGMM-1Gauss combined models in smart room static camera scenario. In red the foreground Gaussian distributions in the spatial domain

Figure 21 presents a smart room sequence with special difficulty because of the color similarity between the foreground subject and the background regions it is occupying. It can be observed how our approach maintain the object detection with some False detections. In Figure 22 we show a comparison between 1Gaussian pixel-wise analysis (6), SCGMM analysis (1) and our approach.

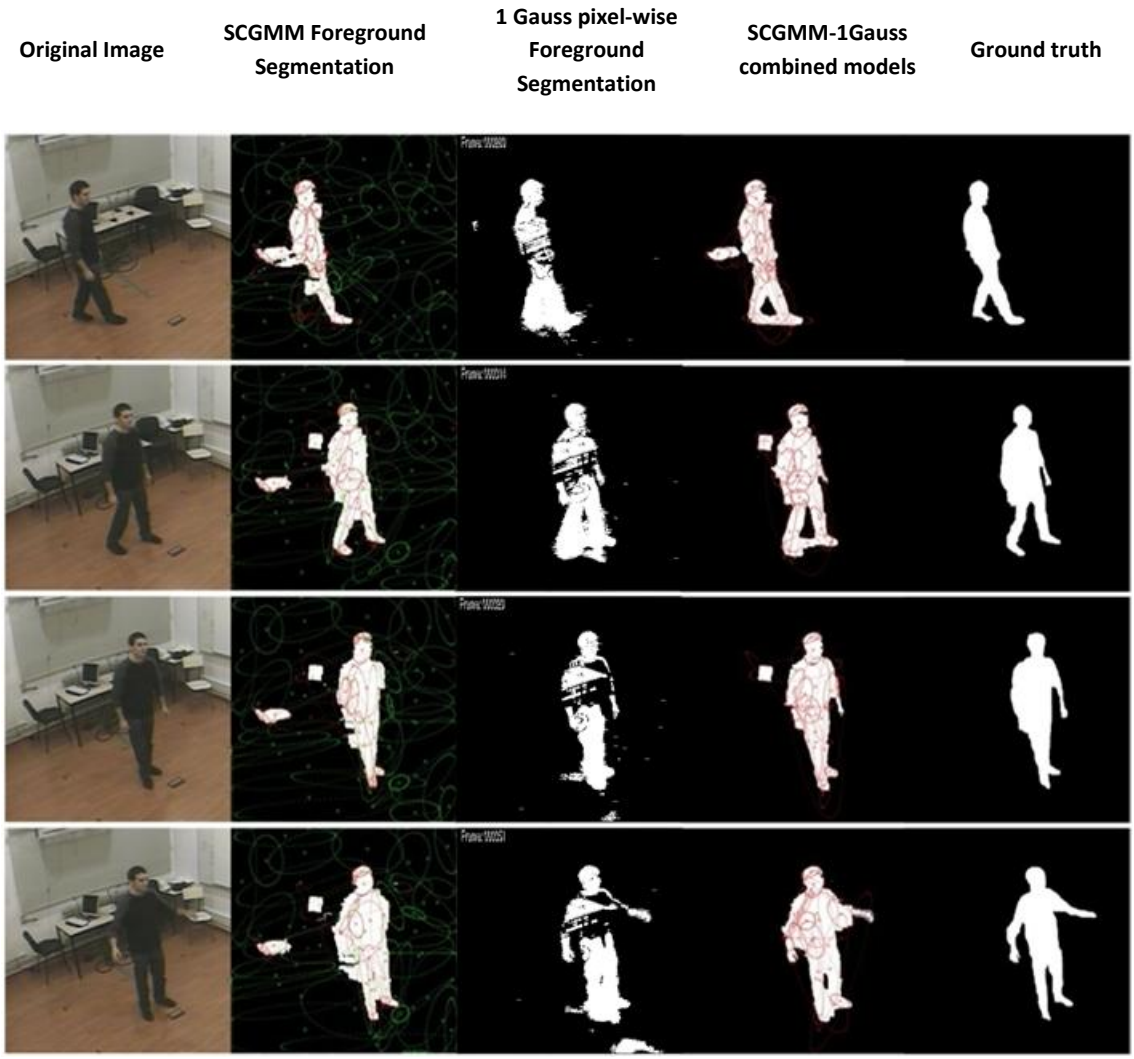


Figure 22 Foreground segmentation comparison between SCGMM joint tracking method, 1Gauss pixel-wise method and the SCGMM-1Gauss method

As we can observe in Figure 22 and Figure 23, the SCGMM-1Gauss combined models that we propose, offers better foreground segmentation than separated SCGMM and 1 Gauss methods. It can be observed how the object's silhouette is maintained and the false negatives are minimized, thanks to the correct modeling of the object with SCGMM. False positives are also reduced due to the accuracy in the background modeling that pixel-wise techniques present. Regarding to the shadows, our method avoids a high percentage of falses positives originated by this effect. Our approach combines the optimal features of pixel-wise background segmentation and region based SCGMM foreground segmentation achieving a more robust segmentation in foreground-background similar regions.

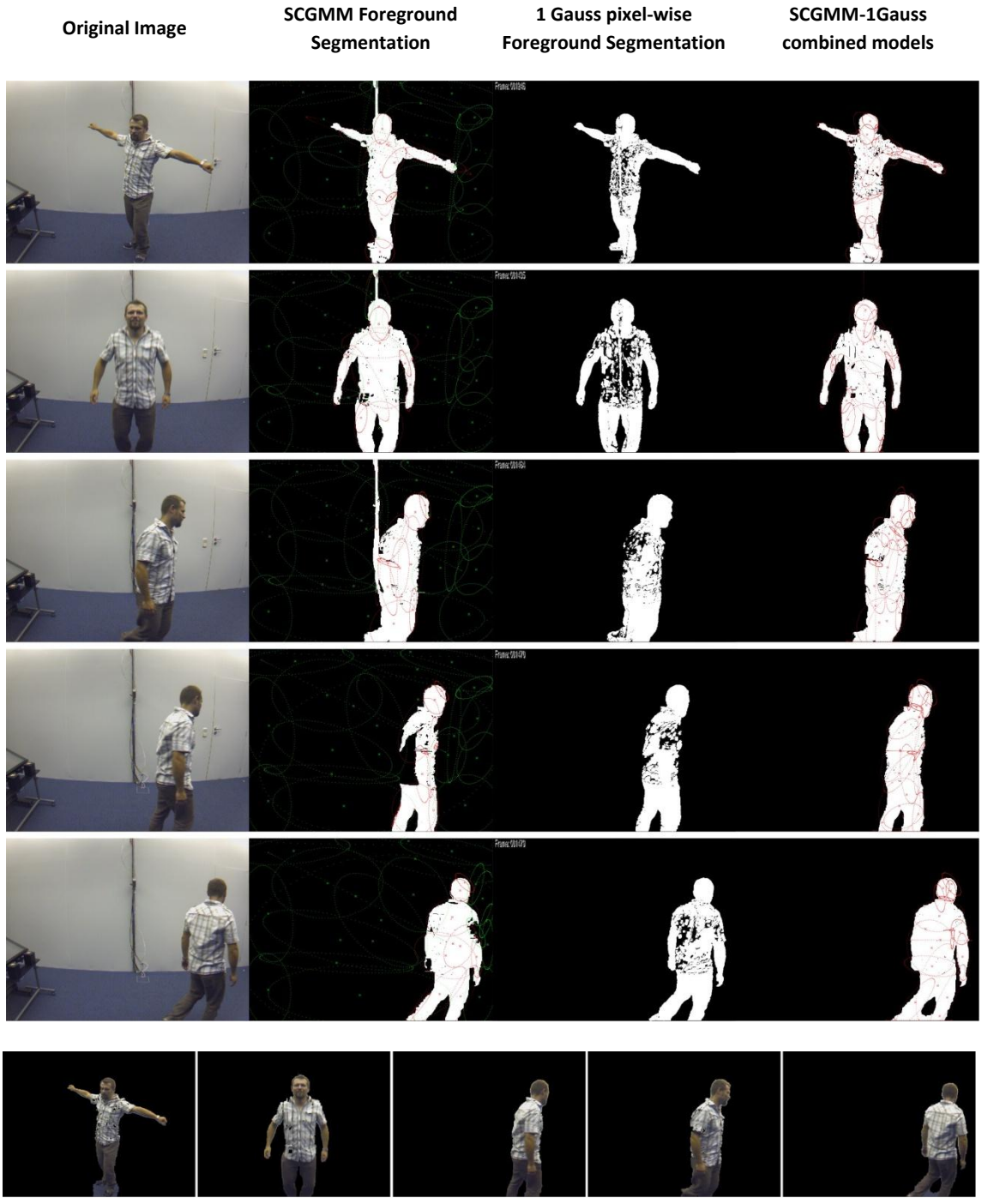


Figure 23 Up: Foreground segmentation comparison in smart room between SCGMM joint tracking method, 1Gauss pixel-wise method and the SCGMM-1Gauss method. Down: Results of SCGMM-1Gauss combined models

4.3.1.6 Conclusions

In spite of the appearance of some false positive detections, the proposed system improves the foreground segmentation obtained by other systems commonly used today. Using region based foreground probabilistic modeling combined with a pixel-wise probabilistic modeling, we are taking advantage of the strength of each method:

- Precision thanks to pixel wise background modeling
- Prior information of the object thanks to SCGMM foreground modeling.

The technique proposed has lead us to the next proposal (point 4.3.2), to improve the results.

4.3.2 Foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking

In this section we explain the bases of this approach to improve the foreground segmentation in static sequences. The characteristics, work flow, implementation and results will be explained in the following lines.

4.3.2.1 Characteristics

Foreground objects segmentation for static monocular camera configuration.
Use SCGMM to model the foreground.
Use 1 color Gaussian per pixel to model the background.
Foreground model updating before decision
It doesn't allow real time analysis

Table 4-1 Foreground Segmentation in Monocular Static Sequences Via SCGMM-1Gauss Joint tracking Characteristics

4.3.2.2 Bases

After studying the results of the previous approach (Section 4.3.1), we developed this new approach: *foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking* to improve the segmentation results for static sequences. With this method we propose the a system that combines both models (Region Based model for foreground and pixel-wise for background) for Energy minimization and also for the Expectation Maximization joint tracking step. In this way, the GMM of the foreground updates its spatial components taking into account the background model information, before the classification is performed by the graph cuts algorithm.

4.3.2.3 Work Flow

This foreground segmentation system is based, like in the previous approach (Section 4.3.1), in a combination between a pixel-wise model, to model the background pixels, with a region based model (SCGMM) to model the foreground. In contrast to our previous approach, we propose to include a first step to track the foreground model only in the spatial domain, and using the background pixel-wise model information for this aim.

The work Flow is showed in Figure 24:

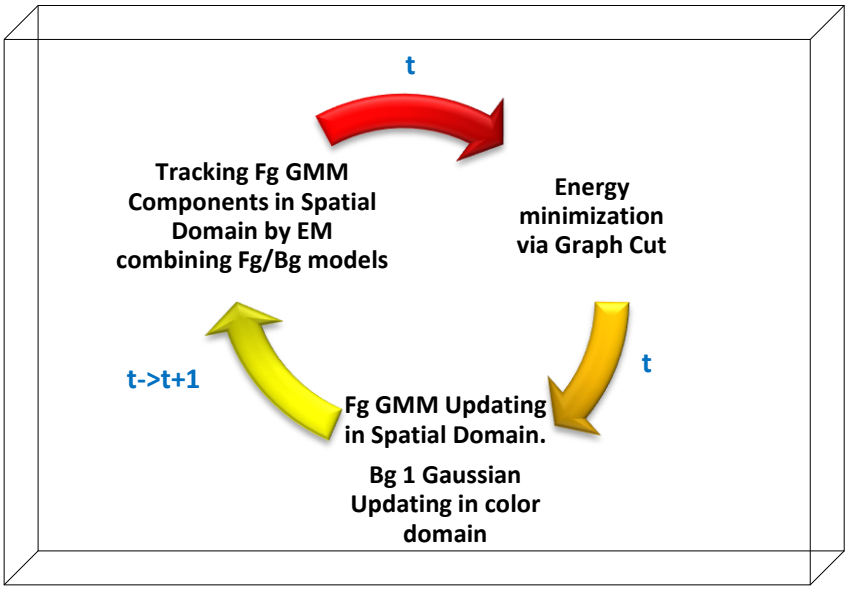


Figure 24 Foreground Segmentation in Monocular Static Sequences via SCGMM-1Gauss Joint Tracking. Work Flow.

As in Section 4.3.1, since we are using two different models for foreground/background probabilistic modeling, we need different kind of initializations for each one:

- **Foreground SCGMM:** First frame of the sequence is used to initialize the foreground model (color-space) by means of the EM algorithm (23). The input observations will be all the foreground object pixels $I_{0,fg}$. Hence, initial foreground segmentation is needed.
- **Background 1Gauss/pixel:** To initialize this probabilistic model, a training sequence is needed (a few number of frames without foreground object) that allows the Gaussian's color means and variances initialization for all pixels of the image I_0.

As can be observed in Figure 24, after the initialization, our algorithm proposes:

1. ***Tracking Foreground GMM Components in Spatial Domain by EM combining Foreground/Background models*** adapts the foreground model from the previous image t-1 to the current frame t. This tracking is obtained via Expectation Conditional Maximization algorithm for spatial domain, and using the background model information performs an accurate modeling over all the pixels of the image.

2. **Energy minimization via Graph Cuts** decides if each pixel of the image is foreground or background via Energy Minimization with Graph Cuts using foreground and background models.

3. **Foreground/Background updating** consists in updating both foreground and background models, the first one in spatial domain and the second one in color domain, taking advantage of the previous step using only the pixels of each class to update each model.

Formally, *foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking* can be described as follows:

We define two probabilistic models for foreground and background modeling, as it is explained in Section 4.3.1-Work Flow, where:

The foreground is modeled via one SCGMM model defined by a set of parameters (θ_{fg}), learned during the system initialization period in the first frame. We use the EM algorithm to maximize the data *Likelihood* of the foreground:

$$\theta_{fg} \stackrel{\text{def}}{\equiv} \{w_{fg,k},\ \mu_{fg,k},\ \Sigma_{fg,\mathrm{k}}\} = \ arg \max_{w_{fg,k},\ \mu_{fg,k},\ \Sigma_{fg,\mathrm{k}}} \prod_{z_{fg} \in I_{fg,0}} \left[\sum_{k=1}^{K_{fg}} w_{fg,k}\ \mathrm{G}\left(\mathrm{z_{fg}}; \mu_{\mathrm{fg,k}}, \Sigma_{\mathrm{fg,k}}\right) \right]$$

Equation 4-17

z_{fg} are the five dimensional features of foreground pixels (r, g, b, x, y); I_0 denotes the initialization frame, $w_{fg,k}$ is the weight of the foreground Gaussian k, $\mu_{fg,k}$ its mean and $\Sigma_{fg,k}$ its covariance matrix.

Thus, we define:

$$p(z|fg) = \sum_{k=1}^{K_{fg}} w_{fg,k} G(z; \mu_{fg,k}, \Sigma_{fg,k})$$

Equation 4-18

where $w_{fg,k}$ is the weight of the k_{th} Gaussian component in the mixture model, z is the input pixel (r, g, b, x, y) that can be split into $z_c = (r, g, b)$ and $z_s = (x, y)$ and $G(z; \mu_{fg,k}, \Sigma_{fg,k})$ is the k_{th} Gaussian component, assuming that the spatial and color components of the SCGMM models are decoupled.

The background is modeled via 1Gauss/pixel background model (θ_{bg}) where the model defines one Gaussian per pixel ($K_{bg} = \#I_0$ gaussians), it is learned during the system initialization using the training sequence, to maximize the data *Likelihood* of the background as it is explained in section 3.1.1.2. That is, the mean and variance of each pixel computed independently.

We formulate the 3-dimensional (r, g, b) pixel-wise model of the background proposed in (6) as a 5-dimensional (r, g, b, x, y) background model for the overall image. Thus, we define

$$p(z_i|bg) = \sum_{k=1}^{K_{bg}} w_{bg} \cdot G\left(z_{c,i}; \mu_{bg,c,k}, \Sigma_{bg,c,k}\right) \cdot \delta(i-k) =$$

$$= \sum_{k=1}^{K_{bg}} \frac{1}{\#I_0} \cdot G\left(z_{c,i}; \mu_{bg,c,k}, \Sigma_{bg,c,k}\right) \cdot \delta(i-k)$$

Equation 4-19

Where c denotes the three dimensional color features (r, g, b). $z_{c,i} \in I_0$ are the color features of pixel i. w_{bg} is the weight of the K_{th} Gaussian of the background model, # denotes cardinality and $\#I_0$ is understood as the image area (total amount of pixels in the image). $G(z_{c,i}; \mu_{bg,k}, \Sigma_{bg,k})$ is the Gaussian distribution that models the i_{th} pixel as:

The *joint tracking*, *energy minimization* and *updating* steps of the work flow will be formulated in next subsections.

SCGMM Joint Tracking

The aim of this step of the process is to propagate foreground SCGMM model over the rest of the sequence, since both the foreground and background objects can be constantly moving. For this purpose, our algorithm obtains an approximate foreground SCGMM model for the current frame before the graph cut segmentation.

It is assumed that from time $t-1$ to t, the colors of the foreground objects do not change. Hence, the color parts of the SCGMM models remain identical:

$$\mathrm{G}\left(\mathrm{z_c};\mu_{\mathrm{fg,k,c,t}},\Sigma_{\mathrm{fg,k,c,t}}\right)=\ \mathrm{G}\left(\mathrm{z_c};\mu_{\mathrm{fg,k,c,t-1}},\Sigma_{\mathrm{fg,k,c,t-1}}\right)$$

Equation 4-20

where c denotes the color dimension, $k=1,\dots,K_I$ the Gaussian distribution number and $\mathrm{z_c}$ the color foreground pixel information.

The updating scheme for the foreground spatial parts $\mathrm{G}\left(\mathrm{z_s};\mu_{\mathrm{fg,k,s,t}},\Sigma_{\mathrm{fg,k,s,t}}\right)$ given the new input image I_t, and the pixel-wise background model where s denotes the spatial features, is as follows :

First it is formed a global SCGMM model of the whole image by combining the foreground SCGMM and background 1Gauss pixel-wise models of the previous frame: $\theta_{I,t}^{0}$, where superscript *0* indicates that the parameter set is serving as the initialization value for the later update.

The probability of a pixel of the image $z_i=(r,g,b,x,y)$ given the global model $\theta_{I,t}^{0}$ can be expressed as the combination of both foreground and background models:

$$p\left(z_t\middle|\theta_{I,t}^{0}\right)=\ p(fg)\,p\left(z_t\middle|\theta_{fg,t-1}\right)+p(bg)p\left(z_t\middle|\theta_{bg,t-1}\right)=$$

$$=\sum_{k=1}^{K_{fg}}w'^{0}_{k,t}\ \mathrm{G}\left(\mathrm{z_{s,i}};\ \mu_{k,s,t}^{0},\ \Sigma_{\mathrm{k,s,t}}^{0}\right)\cdot\mathrm{G}\left(\mathrm{z_{c,i}};\mu_{\mathrm{k,c,t}},\Sigma_{\mathrm{k,c,t}}\right)+$$

$$+\sum_{k=K_{fg}+1}^{K_I}w'^{0}_{k,t}\ \mathrm{G}\left(\mathrm{z_{c,i}};\mu_{\mathrm{k,c,t}},\Sigma_{\mathrm{k,c,t}}\right)\cdot\delta\left(i-k+K_{fg}\right)$$

Equation 4-21

Where i denotes the index of pixel z. Denote $K_I=K_{fg}+K_{bg}$ as the number of Gaussian components in the combined image level SCGMM model, where we assume the first K_{fg} Gaussian components are from the foreground SCGMM, and the last K_{bg} Gaussian components are from the background 1Gaussian/pixel model.

The Gaussian term over the color dimension is defined in Equation 4-20 for foreground, and remains fixed at this moment for background and foreground. The Gaussian component weights $w'^{0}_{k,t}$, $k=1,\dots,K_I$ are different from their original values in their individual foreground or background $w_{k,t}^{0}$ due to $p(fg)$ and $p(bg)$:

$$w'^{0}_{k,t}=\begin{cases}w^{0}_{fg,k,t}\cdot\ p(fg) & if\quad k\le K_{fg}\\ w^{0}_{bg,(k-K_{fg}),t}\cdot\ p(bg) & if\quad K_{fg}<k\le K_I\end{cases}$$

Equation 4-22

Thus, given the pixels in the current frame I_t, the objective is to obtain an updated parameter set $\{\mu^*_{k,s,t},\ \Sigma^*_{\mathrm{k,s,t}}\}$ for foreground model over the spatial domain, which maximizes the joint data likelihood of the whole image, for all $k = 1, \dots, K_I$, i.e.,

$$\{\mu^*_{k,s,t},\ \Sigma^*_{\mathrm{k,s,t}}\} = arg \max_{\mu^*_{k,s,t},\ \Sigma^*_{\mathrm{k,s,t}}} \prod_{z_t \in I_t} p(z_t|\theta_{I,t})$$

Equation 4-23

The Expectation Conditional Maximization algorithm shown in Figure 25 is adopted here to iteratively update the spatial foreground model parameters from their initial values $\theta^0_{I,t}$ keeping the color parameters unchanged from the previous frame.

$$w'_{fg,k} = w_{fg,k} \cdot \gamma_{fg,t-1}$$

$$w'_{bg,k} = w_{bg,k} \cdot \gamma_{bg,t-1}$$

Where $\gamma_{fg,t-1} = \frac{I_{fg,t-1}\_Area}{I_t\_Area}$; $\gamma_{bg,t-1} = \frac{I_{bg,t-1}\_Area}{I_t\_Area}$

Expectation Conditional Maximization

1.st E-step, calculate the foreground Gaussian component assignment probability for each pixel $\boldsymbol{z_i}$:

$$p^{(i)}(k|z_i) = \frac{w'^{(i)}_k \cdot G(z_{\mathrm{i,s}};\ \mu^{(i)}_{k,s},\ \Sigma^{(\mathrm{i})}_{\mathrm{k,s}}) \cdot G(z_{\mathrm{i,c}};\ \mu_{k,c},\ \Sigma_{k,c})}{\sum_{k=1}^{K_{fg}} w'^{(i)}_{k,t} \cdot G(z_{\mathrm{i,s}};\ \mu^{(i)}_{k,s},\ \Sigma^{(\mathrm{i})}_{\mathrm{k,s}}) \cdot G(z_{\mathrm{i,c}};\ \mu_{k,c},\ \Sigma_{k,c}) + \underbrace{w'_{bg,i} \cdot G_{bg}(z_{\mathrm{i,c}};\ \mu_{\mathrm{i},c},\ \Sigma_{\mathrm{i},c})}_{Background\ Gaussian\ pixel\ i}}$$

2.nd M-step, update the spatial mean and variance of the foreground gaussians, considering a constant weight for each Gaussian of the model:

$$\mu^{(i+1)}_{k,s} = \frac{\sum_{z \in I_t} p^{(i)}(k|z) \cdot z_{i,s}}{\sum_{z \in I_t} p^{(i)}(k|z)}$$

$$\Sigma^{(i+1)}_{k,s} = \frac{\sum_{z \in I_t} p^{(i)}(k|z) \cdot (z_{i,s} - \mu^{(i+1)}_{k,s}) \cdot (z_{i,s} - \mu^{(i+1)}_{k,s})^T}{\sum_{z \in I_t} p^{(i)}(k|z)}$$

Figure 25 Expectation Conditional Maximization algorithm for foreground/background joint tracking

Notice that we consider the weights constant, assuming that the background model is present in all the image, as an independent model and should not lose weight, and the foreground will change only the spatial features, maintaining the weights constant to be consequent with background weights assumption.

Energy minimization

The foreground/background segmentation problem is solved, like our previous system shows in section 4.3.1.3-*Energy minimization*, using energy minimization, evaluating the posterior probability of each class (Equation 4-10 and Equation 4-12) as Data function $E_{data}(f)$ in the energy minimization function shown in Equation 4-8.

Foreground-Background updating

After we have obtain the foreground segmentation result from the previous step *Energy Minimization*, we propose to update the foreground model only in color domain, and background model in spatial domain, both in the same way as section 4.3.1.3 shows in the subsections *Fg SCGMM Updating in the Spatial Domain* and *Background 1Gaussian Pixel Model Updating Color Domain* respectively.

4.3.2.4 Implementation Overview

As previous system, the algorithm needs one frame (the first one of the sequence) with the correct foreground segmentation mask of the object we want to segment, to initialize foreground SCGMM model. Also, it needs a training sequence without foreground objects to initialize the background model. The implementation steps are shown next, continuing along:

1- Define the number of Gaussian distributions for foreground model

Common values are: ten or twenty gaussians for each model depending on the number of color regions that the object and the background have.

2- $t = 1$ Initialize both foreground and background models.

- Foreground: using the first frame $t = 1$ foreground pixels $(I_{0,fg})$ executing the EM algorithm for the SCGMM model (θ_{fg}) to initialize color (r, g, b) and position (x, y) of each Gaussian distribution.
- Background: using the training sequence:
  - For $t_{training} = 1$, center all Gaussians means to the input value $z_{c,i}$

$$\mu_{t=1} = z_c$$

  - For $1 < t_{traning} < t_{training\ end}$, update mean and variance with the updating equations: Equation 4-16

3- For next frames $t > 1$:

- Combine foreground SCGMM and background 1 Gaussian pixel wised models (θ_{fg}, θ_{bg}) into one I'_t model ($\theta_{I'}$):

$$\theta_{I',t} \stackrel{\text{def}}{=} \{\theta_{fg,t-1}, \theta_{bg,t-1}, \gamma_{fg,t-1}, \gamma_{bg,t-1}\}$$

Where the K_{fg} foreground gaussians are placed in first position and the $K_{bg} = \#I_0$ background gaussians are placed in the last positions, and:

$$\gamma_{l,t-1} = \frac{\#I_{t,l}}{\#I_t}$$

Where $l \in \{fg, bg\}$, # denotes the cardinality operator and $\#I_t$ can be understood as the region area.

Then, these coverage percentages are used to update the gaussians weights:

- Apply Energy Minimization Graph cut algorithm for each pixel (Section 4.3.1-Energy minimization page 53).

Common value for λ is 200.

Result: Foreground segmentation mask, obtaining $I_{fg,t}$ and $I_{bg,t}$

- Foreground SCGMM spatial domain updating: Execute the **Expectation Conditional Maximization** (Figure 20) algorithm for θ_{fg} model with its corresponding pixels $I_{fg,t}$ to update position (x, y) of each Gaussian distribution.

- Background 1 Gaussian/pixel color domain updating: Use the updating equations Equation 4-16 to update all Gaussians $\theta_{bg,i}$ that model the pixels grouped in $I_{bg,t}$.

4.3.2.5 Results

In this section, we show some segmentation results obtained with *foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking* technique in smart room static camera scenarios. For this purpose, several video sequences will be analyzed.

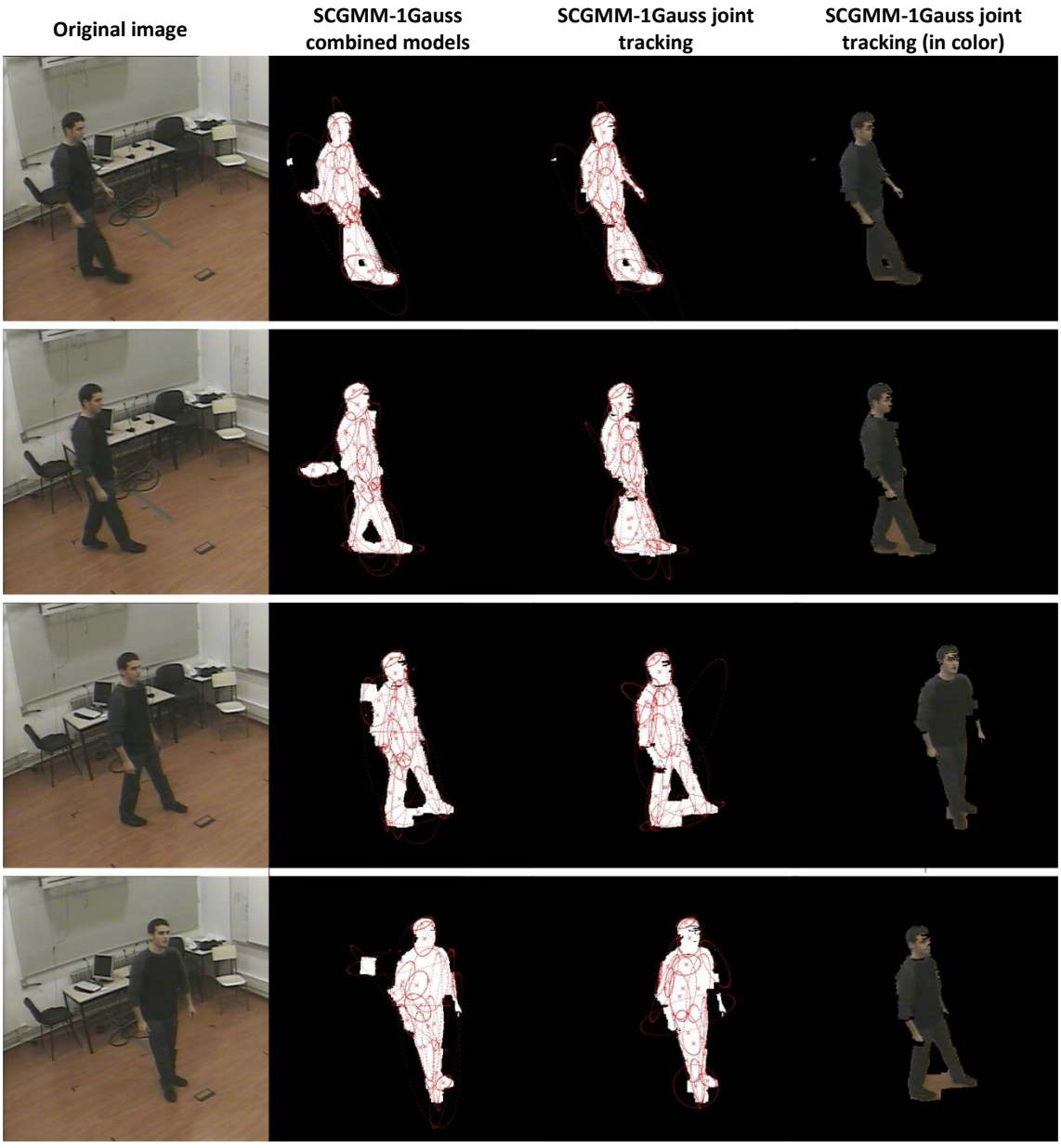


Figure 26 Foreground segmentation comparison between the SCGMM-1Gaussian combined models and SCGMM-1Gaussian joint tracking methods.

Figure 26 shows a comparison between SCGMM-1Gaussian combined models and SCGMM-1Gaussian combined models with joint tracking. As it can be observed, the previous approach detects less false negatives detections than the joint tracking approximation, but in opposite, this previous approach presents more false positives detections that, in some cases, may not be removed with an area filter because they belong to the same connected component as the correct one. Also, it can be observed how the previous approach presents more robustness to

the shadow effect. In spite of this, Figure 27 shows how the current foreground segmentation offers better results than 1 Gaussian pixel wise method in terms of false negatives and shadow errors and SCGMM joint tracking method in terms of false positives.

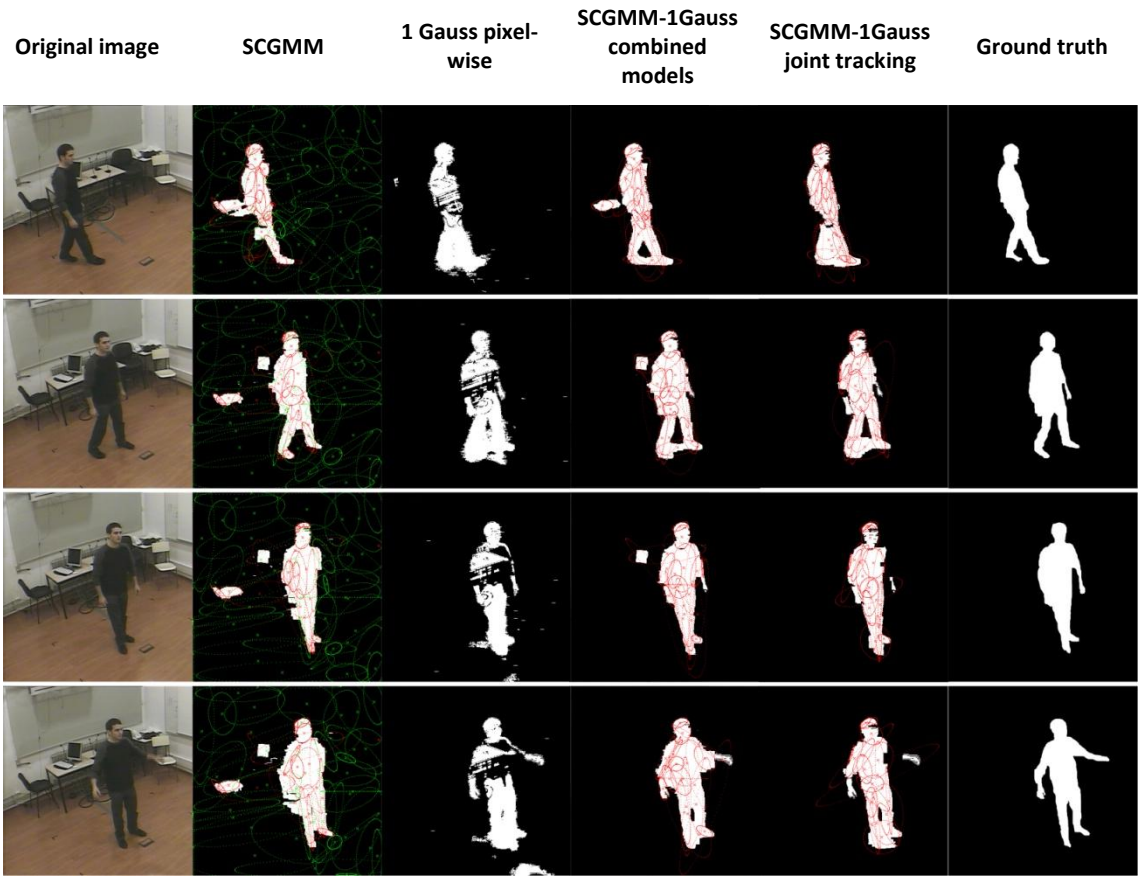


Figure 27 Smart room 1. Foreground segmentation comparison between the SCGMM joint tracking method, 1 Gaussian pixel-wise method, SCGMM-1Gaussian combined models method and SCGMM-1Gaussian joint tracking method. Smart Room1.

In Figure 28 we can observe the results in another sequence of a smart room, where the foreground segmentation of this system contains more false negatives than the previous one without the tracking step, but it maintains a better foreground segmentation than the other methods avoiding false positives compared with SCGMM method, and improving the false negatives compared to 1 Gaussian pixel wise foreground detection.

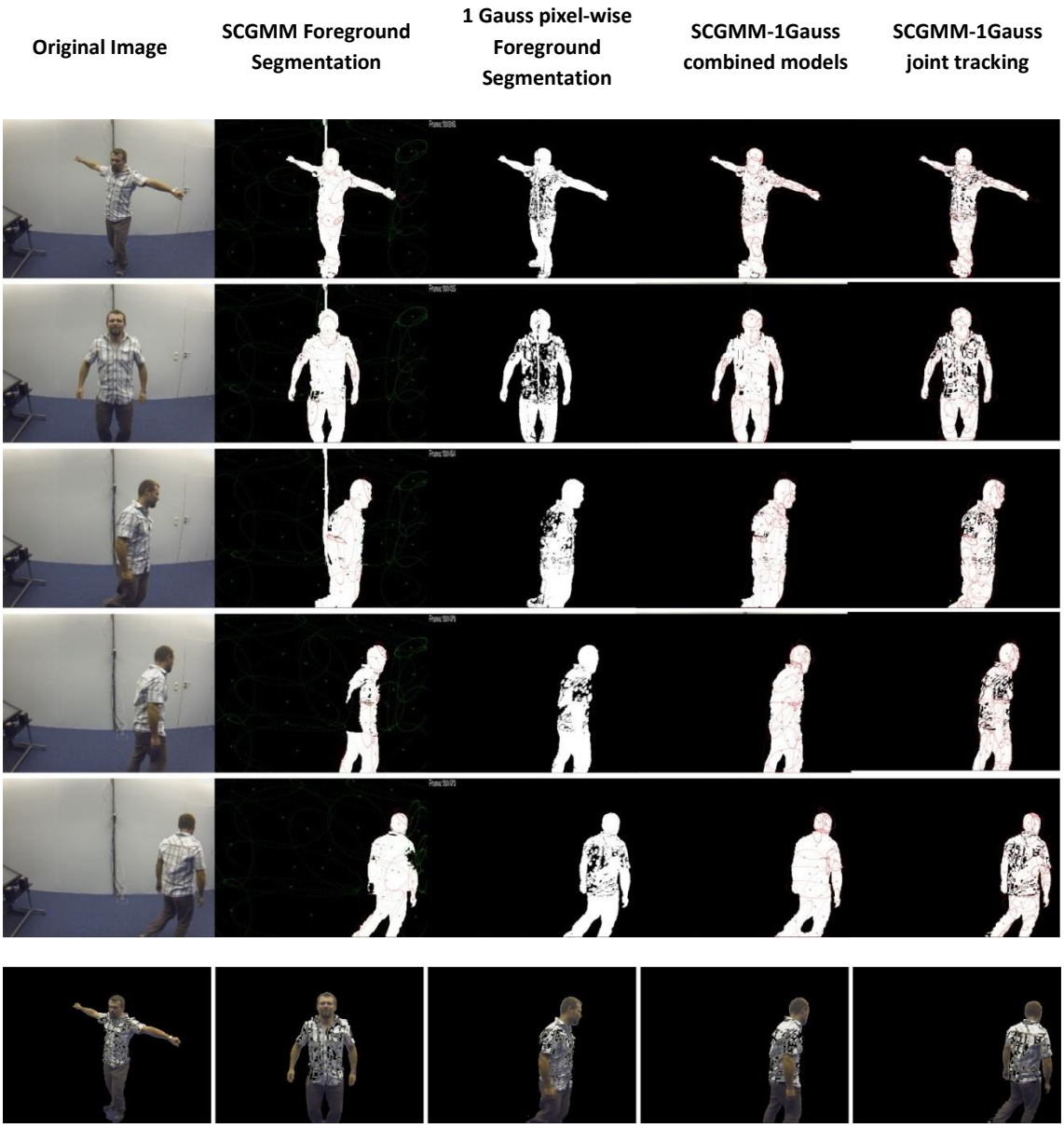


Figure 28 Smart room 2. Up: Foreground segmentation comparison between the SCGMM joint tracking method, 1 Gaussian pixel-wise method, SCGMM-1Gaussian combined models method and SCGMM-1Gaussian joint tracking method. Down: Results of SCGMM-1Gauss joint tracking.

4.3.2.6 Conclusions

In the results we have observed how the proposed system improves the foreground segmentation obtained with other systems of the state of the art. Also, this system reduces the false positives detections that the previous method (explained in 4.3.1) presents, but increasing false negatives detections. This can occur because we are maintaining the weights of the gaussian distributions fixed for the joint tracking step, and this can cause that some foreground gaussians increase their spatial variance without the corresponding weight modification. As a consequence, the probability of this foreground Gaussian distributions decreases. We tried to modify the weights of foreground and background distributions in the Expectation Conditional Maximization algorithm in the joint tracking step, but this cause that background distributions with similar color to the foreground or with some color modifications due to the presence of shadows reduce their weight when the foreground model has a close spatial position. As a consequence, in each EM iteration the background gaussians of these regions reduce their weight and finally the foreground model occludes these regions. This results in false positive detections in the foreground segmentation of the image. We are working in this issue to improve the foreground segmentation.

This system allows precise foreground segmentation with consistent foreground region detection thanks to combine the pixel-wise background model with the region-based foreground model. How to improve the foreground segmentation reducing the false negatives detections will be a future research line.

5 CONCLUSIONS

In the development of this project entitled *Monocular Video Foreground Segmentation via regularized Spatial Color Gaussian Mixture Models*, region-based and pixel-wise foreground segmentation techniques and tracking systems have been studied. After the study of some systems of the state of the art exposed in Section 3(*Running Gaussian average, Stauffer and Grimson GMM, SCGMM joint tracking, Connected Components based tracking, Mean Shift based tracking*), the main problems of these methods have been detected, and three main contributions to the state of the art have been developed on this project:

- *Adaptation of the foreground segmentation and tracking technique via SCGMM for static and moving monocular video sequences in order to speed up computations and allow the foreground segmentation and tracking in highly dynamic backgrounds scenarios.*
- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss combined models*.
- *Foreground segmentation in monocular static sequences via SCGMM-1Gauss joint tracking*

In the first system we propose to segment and track foreground objects in monocular moving camera sequences via SCGMM (Spatial Color Gaussian Mixture Model) foreground/background modeling techniques. We modify the technique proposed in (1) in such a way that the background and foreground are modeled in a small window containing the foreground object and which moves along with it while we track the object. This allows to speed up computations and also, with the proposed color updating of the background, to adapt the algorithm for moving camera sequences analysis. The results obtained with this method, denote that this system is a good solution to segment and track objects in monocular moving sequences.

In the second and third systems, we have proposed a foreground segmentation for monocular static sequences by means of a novel technique combining SCGMM region based modeling to model the foreground and 1Gauss Running average pixel-wised to model the background. These systems are explained in sections 4.3.1 and 4.3.2 respectively. We experimentally show how they improve the foreground segmentation obtained with other systems in difficult environments where the foreground/background similarity makes difficult the foreground segmentation from background. In these cases, the proposed methods reduce the false positive and false negative detections.

6 FUTURE WORK

To continue our work in foreground segmentation and tracking areas, there are several research lines that can be carried on:

Future work in this area should continue improving the foreground segmentation systems developed. For moving camera objects segmentation and tracking it should be developed an appropriate updating for foreground and background models, to improve the adaptation in cases of scaling, rotation and color changes of the objects. Also, occlusion between objects has to be analyzed to develop techniques for solving these situations. For static camera sequences, also the models updating should be improved, to reduce false positive and false negatives detections when foreground and background regions present similar colors. Furthermore, to improve the system for several objects detections is needed to implement a complete foreground segmentation and tracking system, may be combining pixel-wise foreground segmentation methods to initialize the object, continuing the foreground segmentation for next frames, with the methods we propose in this thesis.

Annex

I ANNEX

I.I ENERGY MINIMIZATION VIA GRAPH CUTS

Many vision problems, especially in early vision, can naturally be formulated in terms of energy minimization. The classical use of energy minimization is to solve the pixel-labeling problem, which is a generalization of such problems as stereo, motion, and image restoration . The input is a set of pixels $I = \{I_1, I_2, \dots I_i \dots I_N\}$ and a set of labels l. The goal is to find a labeling f (i.e., a mapping from I to l) which minimizes some energy function (27).

In this way, for a video sequence taken by a fixed camera, the foreground segmentation can be formulated as follows (28) (29):

Each frame image contains *N* pixels. Let *S* be the set of indices referring to each of the *N* pixels. Given a set of pixels I, S of current frame at time-step t , the task of object detection is to assign a label $l_i \in \{background(=0), foreground(=1)\}$ to each pixel $i \in S$, and obtain $l = \{l_1, l_2, \dots l_i \dots l_N\}$.

In most of the work in the literature, object detection was attempted by first modeling the conditional distribution $p(I_i|l_i)$ of feature value I_i at each pixel i independently. The model used can be either parametric (6) (8) or non-parametric (30) (31) based on a past window of observed feature values at the given pixel. The background and foreground model will be detailed presently. Assume the observed feature value of image pixels are conditionally independent given *l*, thus:

$$p(I\,|l\,) = \prod_{i=1}^{N} p(I_i|l_i)$$

Equation 6-1

However, it is clear that neighboring labels are strongly dependent on each other. The neighborhood consistency can be modeled with a Markov Random Field prior on the labels:

$$p(l) \propto \prod_{i=1}^{N} \prod_{j \in \varepsilon_i} \varphi(i,j)$$

Equation 6-2

$$\varphi(i,j) = exp\left(\lambda\left(l_i l_j + \ (1-l_i)(1-l_j)\right)\right)$$

Equation 6-3

where λ determines the pair-wise interaction strength among neighbors and ε_i is the four-neighborhood of pixel *i* .

Given the Markov Random Fields prior and the likelihood model above, moving object detection in a given frame reduces to maximum a posterior $P(l\,|I)$ solution. According to the Bayes rule, the posterior is equivalent to

$$P(l\,|I) = \frac{p(I\,|l\,)\cdot p(l)}{P(I)} = \frac{\prod_{i=1}^{N} p(I_i|l_i)\cdot p(l_i)\cdot exp\left(\sum_{i=1}^{N}\sum_{j\in\varepsilon_i}\lambda\left(l_i l_j + (1-l_i)(1-l_j)\right)\right)}{P(I)}$$

Equation 6-4

$P(I)$ is the density of I which is a constant when I is given.

Finally, the MAP estimate is the binary image that maximizes Equation 6-4:

$$arg\max_{l}\ p(I\,|l\,)\cdot p(l) = \arg\min_{l}[-\ln\big(p(I\,|l\,)\cdot p(l)\big)] =$$

$$= \arg\min_{l}\big[-\ln\big(p(I\,|l\,)\big) - \ln(p(l))\big]$$

Equation 6-5

The discrete cost function (Equation 6-5) leads to an standard form of the energy function that can be solved for global optimum using standard graph-cut algorithms (32):

$$E(f) = E_{data}(f) + \lambda\, E_{smooth}(f)\ = \sum_{p\in P} D_p\big(f_p\big) + \ \lambda \sum_{p,q\in N} V_{p,q}\big(f_p, f_q\big)$$

Equation 6-6

where $N \subset P \times P$ is a neighborhood system on pixels. $D_p\big(f_p\big)$ is a function derived from the observed data that measures the cost of assigning the label f_p to the pixel *p* (How appropriate a label is for the pixel). $V_{p,q}\big(f_p, f_q\big)$ measures the cost of assigning the labels f_p, f_q to the adjacent pixels *p,q* and is used to impose spatial smoothness. The role of λ is to balance the data $D_p(f)$ and smooth cost $V_{p,q}\big(f_p, f_q\big)$.

At the borders of objects, adjacent pixels should often have very different labels and it is important that *E* not overpenalize such labelings. This requires that *V* be a nonconvex function of $|f_p - f_q|$. Such an energy function is called discontinuity-preserving.

Energy functions like *E* are extremely difficult to minimize, however, as they are nonconvex functions in a space with many thousands of dimensions. They have traditionally been minimized with general-purpose optimization techniques (such as simulated annealing) that can minimize an arbitrary energy function. As a consequence of their generality, however, such techniques require exponential time and are extremely slow in practice. In the last few years, however, efficient algorithms have been developed for these problems based on graph cuts.

I.I.I GRAPH CUTS

Suppose $\varsigma = (\mathrm{Y}, \varepsilon)$ is a directed graph with non negative edge weights that has two special vertices (terminals), namely, the source s and the sink t. An *s*-*t*-cut (which we will refer to informally as a cut) *C = S; T* is a partition of the vertices in Y into two disjoint sets *S* and *T* such that $s \in S$ and $t \in T$. The cost of the cut is the sum of costs of all edges that go from *S* to *T*:

$$c(S,T) = \sum_{u \in S, v \in T, (u,v) \in \varepsilon} c(u,v)$$

Equation 6-7

The minimum *s*-*t*-cut problem is to find a cut C with the smallest cost. Due to the theorem of Ford and Fulkerson [14], this is equivalent to computing the maximum flow from the

source to sink. There are many algorithms that solve this problem in polynomial time with small constants.

It is convenient to note a cut *C = S,T* by a labeling f mapping from the set of the vertices $\mathrm{Y} - \{S, T\}$ to {0, 1}, where *f(v)*=0 means that $v \in S$ and *f(v)*=1 means that $v \in T$.

Note that a cut is a binary partition of a graph viewed as a labeling; it is a binary-valued labeling. While there are generalizations of the minimum *s*-*t*-cut problem that involve more than two terminals (such as the multi-way cut problem), such generalizations are NP-hard.

I.I.II ENERGY MINIMIZATION VIA GRAPH CUTS

In order to minimize E using graph cuts, a specialized graph is created such that the minimum cut on the graph also minimizes E (either globally or locally). The form of the graph depends on the exact form of V and on the number of labels.

In certain restricted situations, it is possible to efficiently compute the global minimum. This is also possible for an arbitrary number of labels as long as the labels are consecutive integers and V is the L1 distance.

However, a convex V is not discontinuity preserving and optimizing an energy function with such a V leads to over-smoothing at the borders of objects. The ability to find the global minimum efficiently, while theoretically of great value, does not overcome this drawback.

Moreover, efficient global energy minimization algorithms for even the simplest class of discontinuity-preserving energy functions almost certainly do not exist. Consider $V_{p,q}(f_p, f_q) = T[f_p \neq f_q]$, where the indicator function $T[\cdot]$ is 1 if its argument is true and otherwise 0. This smoothness term, sometimes called the Potts model, is clearly discontinuity-preserving.

However, graph cut algorithms have been developed that compute a local minimum in a strong sense. These methods minimize an energy function with nonbinary variables by repeatedly minimizing an energy function with binary variables.

REFERENCES

II REFERENCES

1. **Yu, T. and Zhang, C. and Cohen, M. and Rui, Y. and Wu, Y.** Monocular Video Foreground/Background Segmentation by Tracking Spatial-Color Gaussian Mixture Models. *Motion and Video Computing, 2007. WMVC'07. IEEE Workshop on.* 2007.

2. **P. Dickinson, A. Hunter, K. Appiah.** A Spatially Distributed Model for Foreground Segmentation. *Image and Vision Computing.* 2008.

3. **PICCARDI, M.** Background subtraction techniques: a review. *IEEE SMC 2004 International Conference on Systems, Man and Cybernetics.* 2004.

4. **Butler, D.E. and Bove, V.M. and Sridharan, S.** Real-Time Adaptive Foreground/Background Segmentation. *EURASIP JOURNAL ON APPLIED SIGNAL PROCESSING.* s.l. : EUROPEAN ASSOCIATION FOR SPEECH SIGNAL AND IMAGE PROCESSING, 2005. Vol. 14.

5. **Lo, BPL and Velastin, SA.** Automatic congestion detection system for underground platforms. *Intelligent Multimedia, Video and Speech Processing, 2001. Proceedings of 2001 International Symposium on.* 2001.

6. **Wren, C.R. and Azarbayejani, A. and Darrell, T. and Pentland, A.P.** Pfinder: Real-Time Tracking of the Human Body. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 1997.

7. **Koller, D. and Weber, J. and Huang, T. and Malik, J. and Ogasawara, G. and Rao, B. and Russell, S.** Towards robust automatic traffic scene analysis in real-time. *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on.* 1994. Vol. 1.

8. **Stauffer, C. and Grimson, W.E.L.** Adaptive background mixture models for real-time tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* 1999. Vol. 2.

9. **Oliver, N.M. and Rosario, B. and Pentland, A.P.** A Bayesian Computer Vision System for Modeling Human Interactions. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 2000.

10. **Gabriel, P.F. and Verly, J.G. and Piater, J.H. and Genon, A.** The state of the art in multiple object tracking under occlusion in video sequences. *Advanced Concepts for Intelligent Vision Systems.* 2003.

11. **Xu, LQ and Landabaso, JL and Lei, B.** Segmentation and Tracking of Multiple Moving Objects for Intelligent Video Analysis. *BT Technology Journal.* s.l. : Springer, 2004. Vol. 22, 3.

12. **Comaniciu, D. and Ramesh, V. and Meer, P.** Real-time tracking of non-rigid objects using mean shift. *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on.* 2000.

13. **Collins, RT.** Mean-shift blob tracking through scale space. *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on.* 2003. Vol. 2.

14. **Yilmaz, A.** Object Tracking by Asymmetric Kernel Mean Shift with Automatic Scale and Orientation Selection. *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on.* 2007.

15. **Collins, R. T.** Mean-shift Blob Tracking through Scale Space. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* 2003.

16. **Gallego, J. and Pardas, M. and Landabaso, J.L.** Segmentation and tracking of static and moving objects in video surveillance scenarios. *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on.* 2008.

17. **Arulampalam, MS and Maskell, S. and Gordon, N. and Clapp, T. and Sci, D. and Organ, T. and Adelaide, SA.** A tutorial on particle filters for online nonlinear/non-GaussianBayesian tracking. *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on].* 2002. Vol. 50, 2.

18. **Isard, M. and Blake, A.** CONDENSATION—Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision.* s.l. : Springer, 1998. Vol. 29, 1.

19. **A. L. Mendez, C.Cantón,.** Seguimiento 3D para Múltiples Personas en Entornos Multicámara Basado en Filtros de Partículas. *PFC .* s.l. : ETSTEB, 2007.

20. **Nummiaro, K. and Koller-Meier, E. and Van Gool, L.** An adaptive color-based particle filter. *Image and Vision Computing.* s.l. : Elsevier, 2003. Vol. 21, 1.

21. **Bergman, N. and Dept. of Electrical Engineering.** *Recursive Bayesian Estimation: Navigation and Tracking Applications.* s.l. : Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1999.

22. **Landabaso, JL and Pardas, M.** COOPERATIVE BACKGROUND MODELLING USING MULTIPLE CAMERAS TOWARDS HUMAN DETECTION IN SMART-ROOMS. *Proceedings of European Signal Processing Conference.* 2006.

23. **Dempster, A.P. and Laird, N.M. and Rubin, D.B. and others.** Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society.* 1977. Vol. 39, 1.

24. **Greenspan, H. and Goldberger, J. and Mayer, A.** Probabilistic Space-Time Video Modeling via Piecewise GMM. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 2004.

25. **MENG, X.L.I. and RUBIN, D.B.** Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika.* s.l. : Biometrika Trust, 1993. Vol. 80, 2.

26. **Landabaso, JL and Pardas, M. and Xu, LQ.** Shadow Removal with Blob-based Morphological Reconstruction for Error Correction. *Proceedings of International Conference on Acoustics, Speech and Signal Processing, IEEE Computer Society, Philadelphia, PA, USA, March.* 2005.

27. **Boykov, Y. and Veksler, O. and Zabih, R.** Fast Approximate Energy Minimization via Graph Cuts. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 2001.

28. **Sheikh, Y. and Shah, M.** Bayesian object detection in dynamic scenes. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on.* 2005. Vol. 1.

29. **Greig, D. and Porteous, B. and Seheult, A.** Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society, Series B.* 1989. Vol. 51, 2 .

30. **Elgammal, A. and Duraiswami, R. and Harwood, D. and Davis, L.S.** Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE.* 2002. Vol. 90, 7.

31. **Sheikh, Y. and Shah, M.** Bayesian Modeling of Dynamic Scenes for Object Detection. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 2005.

32. **Kolmogorov, V. and Zabih, R.** What Energy Functions Can Be Minimized via Graph Cuts? *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.* s.l. : IEEE Computer Society, 2004.

33. *Computer Vision Workload Analisys: Case Study of Video Surveillance Systems.* **T.P. Chen, H. Haussecker, A. Bovyrin, R. Belenov, K. Rodyushkin, A. Kuranov and V. Eruhimov.** May 19, 2005. Intel Technology Journal, Volume 09, ISSN 1535-864X,.

34. **McKenna, S.J. and Jabri, S. and Duric, Z. and Wechsler, H.** Tracking interacting people. *4th International Conference on Face and Gesture Recognition, Grenoble, France.* 2000.