Machine Vision and Applications manuscript No.
(will be inserted by the editor)

Pearson-based Mixture Model for Color Object Tracking

W. Ketchantang[1,2], S. Derrode[1], L. Martin[2] and S. Bourennane[1]

[1] Univ. Paul Cézanne, Institut Fresnel (CNRS UMR 6133),
Dom. Univ. de Saint Jérôme, 13013 Marseille cedex 20, France
e-mail: william.ketchantang@fresnel.fr
[2] ST Microelectronics, ZI Rousset BP 2, 13106 Rousset, France
e-mail: lionel.martin@st.com

Received: date / Revised version: date

Abstract To track objects in video sequences, many studies have been done to characterize the target with respect to its color distribution. Most often, the Gaussian Mixture Model (GMM) is used to represent the object color density. In this paper, we propose to extend the normality assumption to more general families of distributions issued from the Pearson's system. Precisely, we propose a method called Pearson Mixture Model (PMM), used in conjunction with Gaussian copula, which is dynamically updated to adapt itself to the appearance change of the object during the sequence. This model is combined with Kalman filtering to predict the position of the object in the next frame. Experimental results on gray-level and color video sequences show tracking improvements compared to classical GMM. Especially, the PMM seems robust to illumination variations, pose and scale changes, and also to partial occlusions, but its computing time is higher than the computing time of GMM.

1 Introduction

Video object tracking is a primary task required by various applications such as visual surveillance, automated video analysis, robotic, biometry, ... Most of applications require the tracker to be robust to partial occlusions, scale and appearance variations of the object of interest. The problem is difficult when no prior information can be used from the scene background. Moreover the algorithm should run in "real time" and have some computer resources left for higher level tasks such as target recognition and semantic interpretation. Hence, efforts are still needed to improve the compromise between the model sophistication and computing time. Strategies for tracking depend es-

sentially on the way the object is described and characterized. There is two principal cues: (i) the object contour in which case the tracking is realized by following the shape [1,2], or (ii) the intensity/color distribution of pixels within the object and the tracking is then done by dynamically updating a distribution model [3]. Each of these cues may fail under certain conditions such as rapid changes in shape or fast variations in illumination, respectively. Recent works, such as the ones by Leymarie *et al.* [1] for snakes and Xiong *et al.* [4], McKenna *et al.* [3] and Stern *et al.* [5] for statistical color modeling, try to compensate for and get more robust tracking systems. This work lies within the second category and we propose some improvements for characterizing object color distribution.

Several approaches have been proposed to characterize color distributions. Among them Peng *et al.* [6] used a non parametric model based on the mean shift algorithm to find modes, whereas McKena *et al.* [3], Störring *et al.* [7], Xiong *et al.* [4] proposed a parametric Gaussian Mixture Model (GMM). Most of the time, Gaussian distributions are used to characterize the colors distribution. This choice is motivated by some common, and generally well-accepted, assumptions on natural scene, sensor noise and colorimetric space characteristics. Another important fact promoting the normality assumption comes from its ease of use and its low computation time. However, this assumption may sometimes appear too strong. For example, it is not well-suited to certain kind of sensors, like infra-red video cameras or those from medical modalities. Also it is usual for tracking to work on other color spaces than RGB to make the system more robust to illumination changes. But certain nonlinear transformations between color spaces may result in an inadequacy of the statistical model. Hence, the mixture does not suit the object histogram well and, as a consequence, the target may be lost. In this paper, we propose to model the color distribution by a statistical mixture coming from the Pearson's system of distributions. This system holds a large set of shapes, among them Gaussian, Gamma and Beta families, and has been used with success in satellite images [8]. The "best distribution" can be automatically selected and their parameters estimated by adapting the SEM (Stochastic Estimation Maximization) principle to the context. When the number of bands in the image is greater than one (*i.e.* non intensity video sequences), we face the classical problem of modeling multidimensional non Gaussian densities. We then use the theory of copulas [9], first introduced in image processing in [10], and especially a Gaussian copula with margins from Pearson's system, to model the correlation between bands. Kalman filtering is introduced to predict target position in the next frame.

This paper is organized as follows. The PMM model for both intensity and color sequences with Pearson's system and Gaussian copula is presented in section 2. The tracking principle is then detailed in section 3, by adapting previous works to the new context. Section 4 presents tracking experiments on gray level video sequences and on color video sequences. Section 5 draws conclusion and recommendations.

2 PMM-based object tracking

This section presents the non-Gaussian model used to describe the distribution of the object to track. We start by presenting the case of intensity sequences by means of the Pearson's system of distributions and followed by the more difficult case of color sequences by means of Gaussian copulas.

2.1 PMM for gray level object based tracking

The object intensity is modeled by a mixture of K probability density functions (pdf) $\{f_k(.|\theta_k)\}$ parameterized by θ_k, according to:

$$p(x|\Theta) = \sum_{k=1}^{K} \alpha_k \, f_k(x|\theta_k), \tag{1}$$

with $\Theta = \{(\theta_1, \alpha_1), \cdots, (\theta_K, \alpha_K)\}$ and weights α_k are such that $\sum \alpha_k = 1$. Most of the time, all pdf in the mixture are supposed Gaussian [11] and $\theta_k = (\mu_{1,k}, \mu_{2,k})$, μ_1 denoting the mean and μ_2 the variance. When the Gaussian assumption is not considered reliable, it is always possible from eq. (1) to choose for another family of densities, such as Gamma, Beta, ... The selection of a specific family requires some prior knowledge on the physics of the sensor and/or the scene. When this information is not available, a simple and quite efficient solution consists in automatically selecting the best family within the Pearson's system of distributions [8]. These families are solutions of the following differential equation:

$$\frac{1}{f(x)} \frac{d\,f(x)}{d\,x} = -\frac{x+a}{c_0 + c_1\,x + c_2\,x^2}.$$

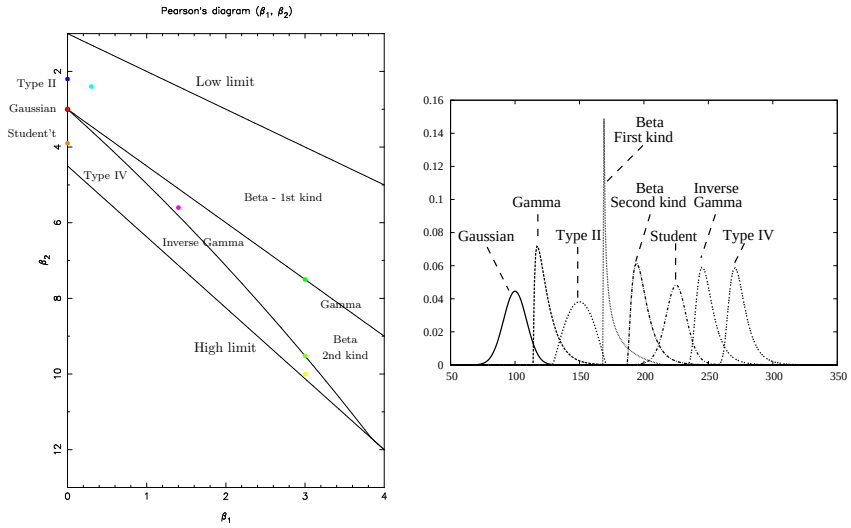
This system is made up of mainly eight families of densities (including Gaussian, Gamma and Beta) and offers a large variety of shapes (symmetrical or not, with finite or semi-finite support, ...) [12]. Each density is defined uniquely by its mean (μ_1) and its first three centered moments (μ_2, μ_3 and μ_4). All of them can be represented in the so-called Pearson diagram (as shown in Fig. 1) in which axes β_1 and β_2 are given by:

- Skewness $\sqrt{\beta_1}$ with $\beta_1 = \frac{\mu_3^2}{\mu_2^3}$,
- Kurtosis $\beta_2 = \frac{\mu_4}{\mu_2^2}$.

Gaussian densities are located at $(\beta_1 = 0, \beta_2 = 3)$, Gamma ones (III) on the straight line $\beta_2 = 1.5\,\beta_1 + 3$ and inverse Gamma ones on the curve with equation

$$\beta_2 = \frac{3}{\beta_1 - 32}\left(-13\beta_1 - 16 - 2\,(\beta_1 + 4)^{3/2}\right), \text{ for } 0 < \beta_1 < \frac{96}{25}.$$

First kind Beta densities are located between the lower limit and the Gamma line, second kind Beta densities are located between the Gamma and the



(a) (b)

Fig. 1 The Pearson's (β_1, β_2)-diagram (left) and examples of pdfs for each main family of distributions (right). Note that the β_2 axis is reversed in (a). Densities in (b) are drawn with $\beta_1 > 0$ and shifted for a better visualization.

inverse Gamma ones, and Type IV densities are located between the inverse Gamma densities and the upper limit.

Once parameters $\theta_k = (\mu_{1,k}, \mu_{2,k}, \mu_{3,k}, \mu_{4,k})$ have been estimated for each pdf k in the mixture, it becomes possible to assess (i) the family of distributions within the Pearson's system from coordinates $(\beta_{1,k}, \beta_{2,k})$ and (ii) the parameters that precisely characterize the pdf within its family, *e.g.* the Gamma parameters can be expressed in term of its four first moments. Note that the GMM is a particular case of PMM since Gaussian distributions belong to the Pearson's system, with parameters $\theta_k = \left(\mu_{1,k}, \mu_{2,k}, 0, 3\,\mu_{2,k}^2\right)$, which gives $\beta_{1,k} = 0$ and $\beta_{2,k} = 3$.
Similarly to the GMM case, the estimation of parameters in set Θ can be done by maximizing the log-likelihood $\mathcal{L}(\chi)$ of an identically and independently distributed sample $\chi = \{x_1, \cdots, x_N\}$:

$$\mathcal{L}(\chi) = \log p(\chi; \Theta) = \log \prod_{n=1}^{N} p(x_n | \Theta) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \alpha_k \, f_k(x_n | \theta_k). \quad (2)$$

No analytic solution exists to maximise the log-likelihood, and we use the SEM algorithm proposed by Celeux *et al.* [13] to estimate parameters. The SEM algorithm is an iterative process which updates estimations of $\{\alpha_k, \theta_k\}_{k=1,\ldots,K}$ until convergence. The principle of SEM consists in simulating a realization of the hidden process Y using draws according to

$p\left(y_n = k \left| x_n, \alpha_k^{[i-1]}, \theta_k^{[i-1]} \right.\right)$, followed by an estimation step from completed data using empirical estimators.

2.2 PMM with Gaussian copulas for object color based tracking

When dealing with video sequences, color information plays an important role and tracking algorithms should try to take benefit of it. In such a context, the mixture becomes M-dimensional, M being the number of components in the color space, *i.e.* the number of colorimetric bands of the sequence (*e.g.* 3 for RGB, 2 for $C_b\, C_r$ sequences, ...). In the GMM case, the bands are supposed to be correlated and it requires the estimation of K covariance matrices for the K multidimensional Gaussian distributions. However, in a non-Gaussian context, multidimensional pdfs are quite hard to obtain. One can deal with multivariate versions of classical distributions [14]. But they are very few and impose all margins to be from the same family (*e.g.* Gamma margins for a multidimensional Gamma distribution). Another solution consists of using tools from multivariate data analysis such as Principal Component Analysis (PCA) which considers correlation between bands. However, PCA estimates the principal component and not the marginales.

An interesting solution to our problem comes from the theory of copulas [9] which has been introduced in the signal and image processing field recently [10]. A bi-variate copula is a cumulative density function (cdf) on the unit square with uniform margins. Such functions have the capability of giving an exhaustive description of the dependence between two random variables. Indeed, Sklar [9] has shown that the link between any continuous joint distribution F_{X_1,X_2} and its marginal distributions F_{X_1}, F_{X_2} is achieved with a copula $\mathcal{C}$:

$$F_{X_1,X_2}(x_1, x_2) = \mathcal{C}\left(F_{X_1}(x_1), F_{X_2}(x_2)\right). \tag{3}$$

Copulas act as a parametric model of the dependence between observations, whatever the marginal distributions. By derivation, the density may be written as

$$f_{X_1,X_2}(x_1, x_2) = f_{X_1}(x_1)\ f_{X_2}(x_2)\ c\left(F_{X_1}(x_1), F_{X_2}(x_2)\right), \tag{4}$$

with $c(u, v) = \frac{\partial^2 \mathcal{C}(u,v)}{\partial u\, \partial v}$ is the density of the copula. These results are also available for a cdf F on $\mathbb{R}^M$ with margins F_1, ..., F_M.

Many parametric copulas exist, namely Normal, Student's t, Frank's and Clayton's for examples. We only consider here the Normal copula whose density is given by

$$c\left(u_1, \ldots, u_M; \boldsymbol{\rho}\right) = |\boldsymbol{\rho}|^{-\frac{1}{2}}\ e^{-\frac{1}{2}\left(\boldsymbol{\zeta}^t\left(\boldsymbol{\rho}^{-1} - \boldsymbol{I}\right)\boldsymbol{\zeta}\right)}, \tag{5}$$

where $\boldsymbol{\zeta}^t = \left(\phi^{-1}(u_1), \ldots, \phi^{-1}(u_M)\right)$ with ϕ the cdf of the normalized Gaussian density, $\boldsymbol{I}$ the identity matrix, $\boldsymbol{\rho}$ the correlation matrix and $|\boldsymbol{\rho}|$ its

determinant. The correlation matrix can be estimated using SEM by estimating variances and covariances. The Gaussian copula is very useful since computations are rather easy and the dependence structure is very intuitive, based on the usual correlation coefficients. The main advantage on multivariate Gaussian densities is that the margins are not necessarily Gaussian and can be chosen from the Pearson's system. Also, note that a Gaussian copula with Gaussian margins gives precisely a multivariate Gaussian density. Finally, the product copula, which is defined by $c(u_1, \ldots, u_M) = 1$, corresponds to the independence case since we can write from eq. (4)

$$f_{X_1,\ldots,X_M}(x_1, \ldots, x_M) = f_{X_1}(x_1) \cdots f_{X_M}(x_M).$$

We can notice that the product copula, contrary to the Gaussian copula, does not take into account the correlation between bands.

As a conclusion, the mixture in eq. (1) can be re-written as

$\forall\, \mathbf{x} = (x_1, \ldots, x_M)^t \in \mathbb{R}^M$,

$$\begin{aligned} p(\mathbf{x}|\Theta) &= \sum_{k=1}^{K} \alpha_k \, f_k(\mathbf{x}|\theta_{k,1}, \ldots, \theta_{k,M}, \boldsymbol{\rho}_k) \\ &= \sum_{k=1}^{K} \alpha_k \prod_{m=1}^{M} f_{k,m}(x_m|\theta_{k,m}) \; c(x_1, \ldots, x_M; \boldsymbol{\rho}_k), \end{aligned} \tag{6}$$

where $\theta_{k,m}$ is the set of the four moments defining the m^{th} 1-D margin $f_{k,m}$ of the k^{th} M-dimensional density f_k.

3 Adaptive target tracking with PMM

The overall tracking process follows the same steps than *e.g.* [3]. It has been adapted to suit the new description of objects by a mixture of densities described by Pearson distributions for margins and Gaussian copulas for the dependence structure (*i.e.* correlation between bands). An overview of the different steps of the algorithm is presented in Fig. 2.

3.1 Adaptive target detection and segmentation

The prediction of the object position in the next frame is done by using a standard Kalman filter [15]. Using the predicted position of the object at image t, the Research Box (RB) is defined by a rectangle centered on the predicted position, with a size equal to twice the size of the object at previous image: $2h^{[t-1]} \times 2w^{[t-1]}$. Indeed, a factor two is a good compromise between a good object localization and a low computing time.

The Kalman filter is not used to manage total occlusions. The reason is the following: if a second moving target (B) hides the object of interest

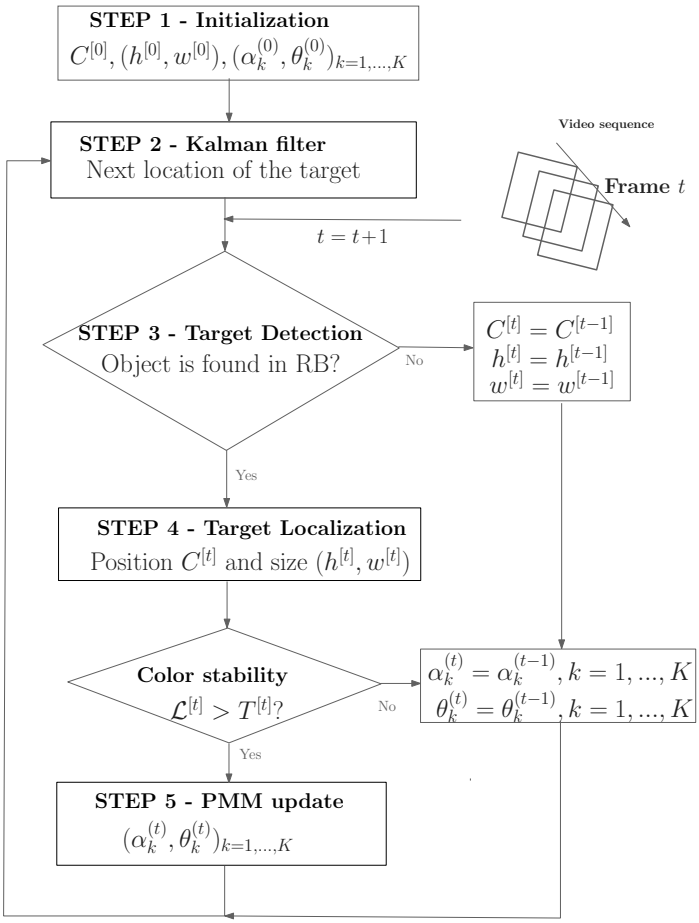


Fig. 2 Synopsis of the overall tracking algorithm.

(A), its color distribution will be learned automatically, and consequently, Kalman filter will focus on B instead of A. To avoid such situation, the presence of the object of interest in the prediction window is first verified. The prediction is stopped when A is considered lost. To do that, all pixels $\mathbf{x}$ such that $p\left(\mathbf{x}\left|\Theta^{[t-1]}\right.\right) \geq P$ are selected as target pixels. If a given percentage of pixels in RB do not verify this condition, then the object is considered lost. Consequently, Kalman prediction is stopped, until object reappears near to its previous localization. In order to cope with illumination changes, the threshold P is adaptively obtained using $P = \mu_p + k\,\sigma_p$ (in our experiment, we set $k = 0.5$), where

$$\mu_p = \frac{1}{N}\sum_{\mathbf{x}\in \mathrm{RB}} p\left(\mathbf{x}\left|\Theta^{[t-1]}\right.\right), \quad \sigma_p^2 = \frac{1}{N}\sum_{\mathbf{x}\in \mathrm{RB}} \left(p\left(\mathbf{x}\left|\Theta^{[t-1]}\right.\right) - \mu_p\right)^2, \quad (7)$$

(a) Target in area search

(b)

(c)

(d)

Fig. 3 Face segmentation results in image (a) with thresholds set to $C_b \in [77, 127]$ and $C_r \in [133, 173]$ (b), with GMM (c) and with PMM (d).

We denote by $S(x)$ the binary image obtained from probability thresholding. This technique reduces sensitivity to illumination variations and when the background color is similar to the object one.

Our segmentation algorithm using PMM visually provides good results even if color background distribution is almost similar to color object distribution (see Fig. 3(d)). The segmentation technique using GMM also visually gives good results but some object pixels are considered as belonging to background pixels (see Fig. 3(c)). Face segmentation method using heuristic thresholds on C_b, C_r [16] introduces some errors on face pixels classification, see Fig. 3(b). Indeed, some background pixels are considered as belonging to the face and conversely. The next step consists in locating precisely the object within the research box.

3.2 Update of target localization parameters and color stability criterion

We now search for the Localization Box (LB) inside the RB, its position $C^{[t]}(c, l)$ and its width $w^{[t]}$ and height $h^{[t]}$. Since the geometric center may be too sensitive to misclassified pixels in image S, we propose the following estimations for c and l

$$c = \frac{\sum_{\mathbf{x}\in \mathrm{RB}} a\, p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)}{\sum_{\mathbf{x}\in \mathrm{RB}} p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)}, \quad l = \frac{\sum_{\mathbf{x}\in \mathrm{RB}} b\, p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)}{\sum_{\mathbf{x}\in \mathrm{RB}} p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)},$$

where (a, b) denotes the coordinates of the pixel $\mathbf{x}$ and $p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)$ is given by eq. (6).

Using image S, we compute $w^{[t]}$ and $h^{[t]}$ according to [16]

$$w^{[t]} = \left(\frac{4}{\pi}\right)^{1/4} \left(\frac{I_2^3}{I_1}\right)^{1/8}, \quad h^{[t]} = \left(\frac{4}{\pi}\right)^{1/4} \left(\frac{I_1^3}{I_2}\right)^{1/8},$$

with

$$I_1 = \sum_{\mathbf{x}\in \mathrm{RB}} (a-i)^2, \quad I_2 = \sum_{\mathbf{x}\in \mathrm{RB}} (b-j)^2,$$

where (a, b) denotes the coordinates of pixel $\mathbf{x}$ and (i, j) the coordinates of the gravity center of S.

Once the LB is defined, we check if the object color distribution is stable, *i.e.* it does not vary quickly between frames t and $t+1$. An unstable object color distribution can be expected in case of abrupt illumination changes, partial or total occlusions. To quantify the criterion, we check if the log-likelihood $\mathcal{L}^{[t]}$ of the data in the LB is higher than a threshold $T^{[t]}$. This threshold is computed with respect to the L previous values of the log-likelihood, according to

$$T^{[t]} = \mu - k_1\sigma - k_2, \quad \text{with } \mu = \frac{1}{L+1} \sum_{l=t-L}^{t} \mathcal{L}^{[l]}, \quad \sigma^2 = \frac{1}{L+1} \sum_{l=t-L}^{t} (\mathcal{L}^{[l]} - \mu)^2,$$

where $k_1 = e^{-10\,\sigma}$, $k_2 = 2$ and $L = 3$ are fixed experimentally. The PMM is updated if the stability criterion is achieved.

3.3 PMM parameters update

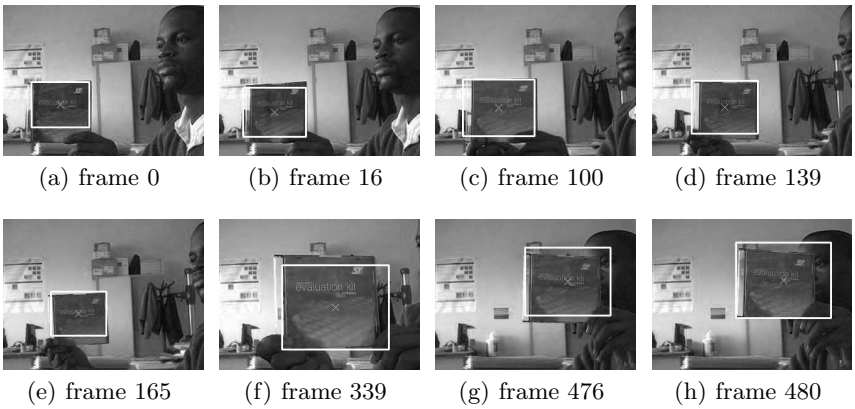
The color distribution of the target changes along the video sequence because of natural variations in the scene (illumination changes, shadows, partial occlusion of the object, ...). It is then important to update it in order to be robust to these changes. So, for a parameter $\nu^{[t]}$ estimated at image t by the SEM algorithm, we add a correcting term depending on previous estimations, with decreasing weights, and get the final estimation of parameter $\nu^{(t)}$ according to

$$\nu^{(t)} = \nu^{(t-1)} + \sum_{l=1}^{L} \left(\nu^{[t]} - \nu^{(t-l)}\right)\ e^{-l\ (L+3)}.$$

This formula is applied to all parameters in Θ, *i.e.* all moments and all correlation matrices. This small modification leads to a more flexible system. This heuristic method inspired from McKena *et al.* [3] is intuitive because the new PMM parameters depend recursively on previous ones, and it requires few computer ressources and a low computing time.

4 Experiments

This section is devoted to test both intensity-and color-based PMM tracking methods proposed in Section 2, within the framework detailed in Section 3.



(a) frame 0 (b) frame 16 (c) frame 100 (d) frame 139

(e) frame 165 (f) frame 339 (g) frame 476 (h) frame 480

Fig. 4 CD box tracking results using PMM. The tracker adapts the localization box according to the object size.

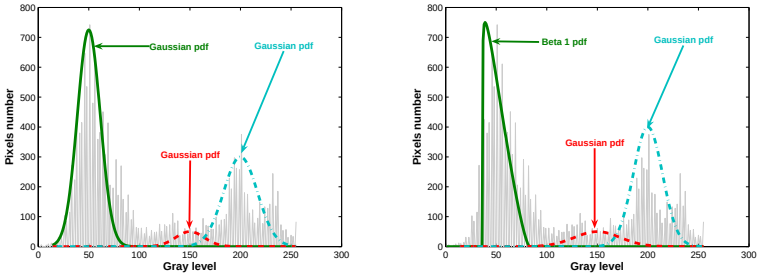
4.1 Object tracking results in a gray-level video sequence

In this first experiment, the object of interest is a Compact Disk (CD) box moving in a gray level video of a laboratory scene (see Fig. 4). The goal of this sequence is to test and compare the tracker when used in conjunction with PMM and GMM. The number of pdfs within the mixture was set to three in both cases.

The CD box is manually selected in frame 0. To get the first estimation $\Theta^{[0]}$ of parameters, the image is first segmented with a classical K-means classifier. Then the SEM algorithm is run until convergence. Figures 5(a) and 5(b) show respectively the GMM and PMM mixtures at frame 0. They constitute the initial GMM and PMM signatures of the CD box. For the PMM-based tracker, we get a first kind Beta distribution and two Gaussian distributions. Obviously, in the GMM case, we get three Gaussian distributions.

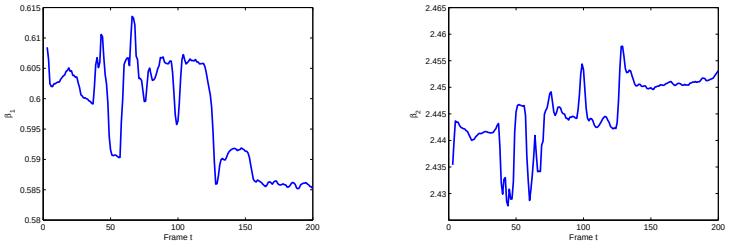
Then, in next frames, parameters are updated dynamically according to the method described in Section 3.3. The family of distributions for each pdf may vary within the Pearson's system. This is illustrated with Fig. 6 that shows the evolution of Pearson's β_1 and β_2 coefficients for one pdf along the sequence. In the GMM case, these parameters are constant ($\beta_1 = 0$ and $\beta_2 = 3$) whatever the pdf.

The good behavior is confirmed by experiments conducted with standard sequence from the CAVIAR dataset where the object of interest is a person walking in a room illuminated by lamps and natural light, as shown in Fig. 7. Object color distribution is modeled by a mixture of three distributions from Pearson's system (2 densities of type II and one Gaussian density) in luminance space. We notice that our model well localizes the object even when illumination changes, as illustrated by frames 27 and 50. But, when the luminance distribution of the object is similar to the background one



(a) (b)

Fig. 5 CD box histogram at frame 0 and estimated pdfs for the GMM (a) and the PMM (b).



(a) β_1 parameter (b) β_2 parameter

Fig. 6 Evolution of β_1 and β_2 parameters of one pdf in the PMM for the CD sequence.

(see frame 255), our model confuses some background and object pixels. As a consequence, target is not well localized.

4.2 Face tracking results in a color video sequence

The second experiment consists in tracking the face in figure 8(a), from a $C_b\,C_r$ sequence with (i) PMM and Gaussian copula and (ii) bi-dimensional GMM. The number of pdfs within the mixture was set to two in both cases, mostly for computational load reasons. The $C_b\,C_r$ colorimetric space was chosen from previous studies on skin color modeling [17], to make the algorithm less sensitive to illumination variations.

From results presented in Fig. 8 with PMM and copulas, we can notice that the tracker is able to adapt the size of the localization box along the sequence, see frames (a) to (d). This is because face segmentation is done according to the automatic threshold on probabilities $p\left(\mathbf{x}\left|\Theta^{[t]}\right.\right)$ (see eq. (7)), which is robust to illumination variations and non constant background (*e.g.* moving camera). In addition, the tracker is able to deal with

(a) frame 0 (b) frame 27 (c) frame 34 - luminosity increases

(d) frame 50 - luminosity decreases (e) frame 140 (f) frame 255 - bad localisation

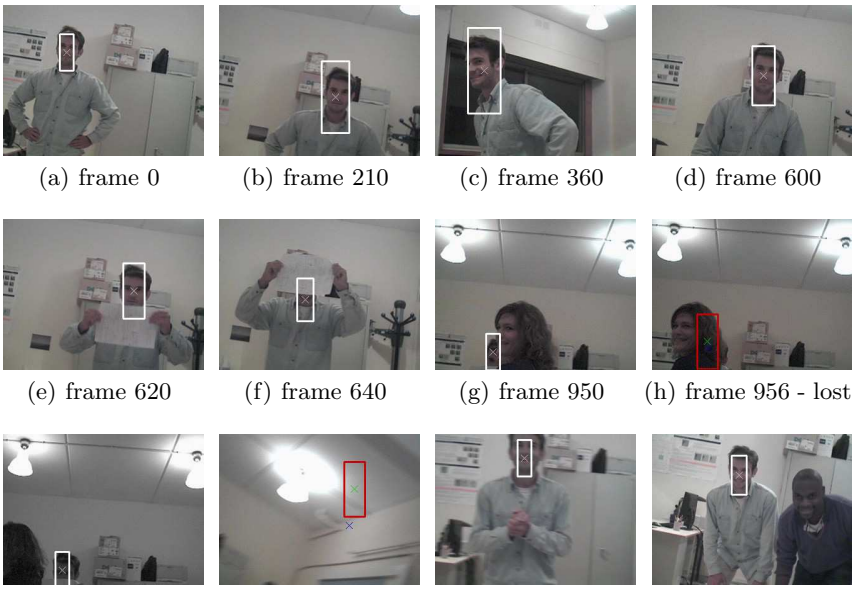
Fig. 7 People tracking in CAVIAR dataset using PMM. The tracker is robust to illumination variations in the scene.

	Class 1		**Class 2**	
α	0.36		0.64	
$\boldsymbol{\rho}$	$\begin{pmatrix} 1 & 0.286 \\ 0.286 & 1 \end{pmatrix}$		$\begin{pmatrix} 1 & 0.329 \\ 0.329 & 1 \end{pmatrix}$	
	Band 0	**Band 1**	**Band 0**	**Band 1**
Pdf family	Gamma	Gaussian	Gamma	Gaussian
β_1	0.32	0.01	0.37	0.02
β_2	3.48	3.01	3.56	3.03

Table 1 Some initial numerical values of PMM with Gaussian copulas.

partial occlusions, see frames (e) to (g). However the target may be lost when a total occlusion occurs, see frames (h) and (j). In frame (h), the hair color distribution is very different from the object distribution and the face is considered lost. The same case arises from frame (j) where the camera looks at the ceil. Our tracking model stops parameters update when the object is lost or when $\mathcal{L}^{[t]}$ is lower than $T^{[t]}$. The tracker can recapture the object of interest if it reappears not too far from the last detected position, see frames 961 and 1331.

To compare performances between PMM with Gaussian copula, PMM with product copula and GMM, we have computed $\mathcal{L}^{[t]}/T^{[t]}$ for the three cases, see Fig. 9. When this ratio is greater than one the model is updated, otherwise the model is not. We can observe that the PMM with Gaussian or product copulas often update the object color distribution. But, contrary



(a) frame 0 (b) frame 210 (c) frame 360 (d) frame 600

(e) frame 620 (f) frame 640 (g) frame 950 (h) frame 956 - lost

(i) frame 961 (j) frame 1306 - lost (k) frame 1331 (l) frame 1830

Fig. 8 Face tracking results using PMM with Gaussian copula with a moving color camera. PMM initial parameters are presented in Tab. 1

to the product copula case, PMM with Gaussian copula takes into account correlation between colorimetric bands, which allows to improve the fit between the multivariate model and the object color distribution. PMM with Gaussian copulas presents interesting performances for color target tracking as shown in Table 2, even if the rate of object localization failure is still high (22%). This is because of the high complexity of the scene since

- camera sometimes moves very quickly (frame 1306 from Fig. 8);
- important natural or artificial illumination variations can be observed (frames 210 and 360);
- sometimes, object and background color distributions are almost similar in the $C_b\,C_r$ space (frame 0);
- finally, partial and total occlusions are observed (frames 640 and 956).

All algorithms have been coded in C++, on a 2.6 GHz Pentium IV personal computer running Linux. The computation time of PMM with Gaussian copulas is higher than the classical GMM model and PMM with product copulas. The small computing time difference between GMM and product copula comes from the estimation of higher order moments for the two Pearson's distributions. The difference between PMM with Gaussian copulas and PMM with product copulas can be explained by the estimation of the two correlation matrices involved in the model, *cf.* eq. (5). It is important to note that the model parameters estimation is more reliable

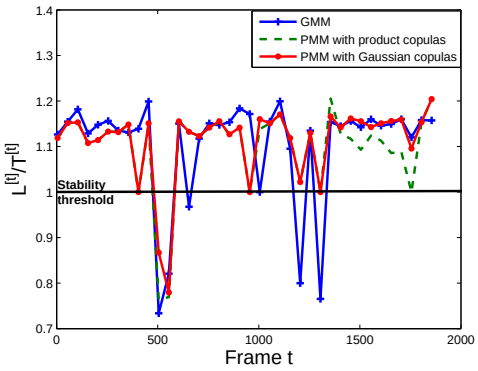


Fig. 9 Comparison between GMM and PMM updates. Color distribution model is updated when $\mathcal{L}^{[t]}/T^{[t]} > 1$.

	GMM	**PMM with product copula**	**PMM with Gaussian copula**
Computing time	19 fps	15 fps	7 fps
Object localization rate	41%	51%	78%

Table 2 Frames per second (fps) rates for GMM, PMM with product copulas and PMM with Gaussian copulas. The object localization rate is computed by dividing the number of frames where the object has been visually well localized divided and the total number of frames where the object is present. Results are given for CIF video images with dimension 352×288, acquired with a low cost camera.

when the object size is large, but the computing time linearly increases with the object dimensions. At the moment, the model does no allow real-time computation on a medium PC but we can expect to reach video frame rates in a near future with a coding optimized for a multi-core processor.

5 Conclusion

In this paper, we have proposed to model the object color distribution by a PMM in conjunction with copula. The PMM is a mixture of pdfs automatically estimated from the Pearson's system of distributions. Moreover, to take into account the statistical links between color channels, we have introduced Gaussian copulas. The PMM is embedded into a tracking system with object position prediction (by Kalman filtering) and parameter updating switch (by log-likelihood thresholding).

Experimental results obtained on various quality videos (different SNR) acquired with different cameras show that the PMM improves, on the one hand, the tracking system flexibility and, on the other hand, the discrimination between the background and the object, even if the background color is

somewhat similar to the target one. However PMM parameters estimation becomes less reliable when object is small, in addition the PMM is 2.5 slower than the classical GMM, mainly because of the estimation of the correlation matrices required by Gaussian copula, . To reduce computing time, we can think of updating the PMM only each k frames, but tracker flexibility will decrease.

According to the tracking objectives, PMM is suggested if the aim is to precisely localize the object in the scene, and GMM is suggested if the aim is to track an object at real time frame rates. Future works include the estimation of the best number of pdfs to be used for tracking a given object. This number may evolve during the tracking, to adapt itself to the change of appearance of the object. This can be done with the Bayesian Information Criterion [18], but at the expense of higher computation time.

Acknowledgments

The authors would express their thanks to PACA region and ST Microelectronics for financial support. Our tests have been partly conducted on surveillance videos coming from the EC Funded CAVIAR project/IST 2001 37540, found at URL: `http://homepages.inf.ed.ac.uk/rbf/CAVIAR/`.

References

1. F. Leymarie and M. D. Levine. Tracking deformable objects in the plane using an active contour model. *IEEE trans. on PAMI*, 15(6):617–634, June 1993.
2. M. Kass, A. P. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. of Computer Vision*, 1(4):321–331, January 1988.
3. S. J. McKenna, Y. Raja, and S. Gong. Tracking colour objects using adaptive mixture models. *Image and Vision Computing*, 17(3/4):225–231, March 1999.
4. G. Xiong, C. Feng, and L. Ji. Dynamical Gaussian mixture model for tracking elliptical living objects. *Pattern Recognition Letters*, 27(7):838–842, May 2006.
5. H. Stern and B. Efros. Adaptive color space switching for tracking under varying illumination. *Image and Vision Computing*, 23(3):353–364, 2005.
6. N. S. Peng, J. Yang, and Z. Liu. Mean shift blob tracking with kernel histogram filtering and hypothesis testing. *Pattern Recognition Letters*, 26:605–614, August 2005.
7. M. Störring, T. Kocka, H. J. Andersen, and E. Granum. Tracking regions of human skin through illumination changes. *Pattern Recognition Letters*, 24(11):1715–1723, July 2003.
8. Y. Delignon, A. Marzouki, and W. Pieczynski. Estimation of generalized mixtures and its application in image segmentation. *IEEE trans. on Image Processing*, 6(10):1364–1375, October 1997.
9. R. B. Nelsen. *An introduction to copulas*, volume 139 of *Lectures Notes in Statistics*. Springer-Verlag, 1998.
10. N. Brunel and W. Pieczynski. Unsupervised signal restoration using hidden Markov chains with copulas. *Signal Processing*, 85:2304–2315, 2005.

11. S. J. McKenna, S. Gong, and Y. Raja. Modelling facial colour and identity with Gaussian mixtures. *Pattern Recognition*, 31(12):1883–1892, 1998.
12. N. L. Johnson and S. Kotz. *Distribution in statistics: Continuous univariate distributions, Vol. 1 and 2*. John Wiley and Sons, New York, 1994.
13. G. Celeux and D. Diebolt. The SEM algorithm : a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly*, 2(1):73–82, 1985.
14. S. Kotz, N. Balakrishnan, and N. L. Johnson. *Continuous multivariate distributions*, volume 1, Models and Applications of *Wiley Series in Probability and Statistics, Applied probability and statistics section*. Wiley Interscience, John Wiley & Sons, Inc, second edition, June 2000.
15. G. Minkler and J. Minkler. *Theory and Application of Kalman filtering*. Palm Bay (Florida, USA), Magellan Book Company edition, 1993.
16. W. Fang and L. Pengfei. A novel face segmentation algorithm. In *Proc. of the Int. Conf. Info-tech and Info-net (ICII'01)*, volume 3, Beijing (China), 29 Oct. - 1 November 2001.
17. D. Chai and K. N. Ngan. Face segmentation using skin-color map in videophone applications. *IEEE trans. on Circuits and Systems for Video Technology*, 9(4):551–564, June 1999.
18. G. Celeux, J. Nascimento, and J. Marques. Learning switching dynamic models for objects tracking. *Pattern Recognition*, (37):1841–1853, January 2004.