Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 25 (2007) 285–296
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Integration of deformable contours and a multiple hypotheses Fisher color model for robust tracking in varying illuminant environments

Francesc Moreno-Noguer [a,*], Alberto Sanfeliu [a], Dimitris Samaras [b]

[a] Institut de Robòtica i Informàtica Industrial UPC-CSIC, Llorens i Artigas 4-6, 08028 Barcelona, Spain
[b] Computer Science Department, State University of New York at Stony Brook, Stony Brook, NY 11794-400, USA

Received 15 October 2004; received in revised form 23 May 2005; accepted 11 October 2005

Abstract

In this paper, we propose a new technique to perform figure-ground segmentation in image sequences of moving objects under varying illumination conditions. Unlike most of the algorithms that adapt color, there is not the assumption of smooth change of the viewing conditions. To cope with this, we propose the use of a new colorspace that maximizes the foreground/background class separability based on the 'Linear Discriminant Analysis' method. Moreover, we introduce a technique that formulates multiple hypotheses about the next state of the color distribution (some of these hypotheses take into account small and gradual changes in the color model and others consider more abrupt and unexpected variations) and the hypothesis that generates the best object segmentation is used to remove noisy edges from the image. This simplifies considerably the final step of fitting a deformable contour to the object boundary, thus allowing a standard snake formulation to successfully track non-rigid contours. In the same manner, the contour estimate is used to correct the color model. The integration of color and shape is done in a stage called 'sample concentration', introduced as a final step to the well-known CONDENSATION algorithm

Keywords: Tracking; Deformable contours; Color adaption; Particle filters

1. Introduction

Color and deformable contours have been extensively used in computer vision applications, such as object detection and tracking tasks [1,2]. Usually, these methods are based on a first step where the object is roughly (but robustly) located by the color module. This simplifies the subsequent step of accurately fitting the contour model to the rigidly or non-rigidly deformed object boundary. In environments with controlled lighting conditions and uncluttered background, color can be considered a reliable and invariant cue, which can be robustly used for tracking. However, when dealing with real scenes with changing illumination and confusing backgrounds, the apparent color of the objects might considerably vary over time, and in these circumstances, an important challenge for any figure-ground segmentation system, is the ability to accommodate color and appearance changes (Fig. 1).

In the literature, the techniques that cope with change in color appearance can be divided in two groups. On the one side, there is a group of approaches that search for color constancy (e.g. [3]); but in practice, these methods work mostly in artificial and highly constrained environments. On the other hand, there are the techniques that generate a stochastic model of the color distribution, and adapt this model over time. In this sense, in [4], color is represented by a histogram that is adapted online, as the weighted function of previous histograms and a predicted histogram. Yang and Lu [5], parameterize object color by a unique Gaussian, the mean and covariance of which are estimated using a linear combination of the parameters in previous Gaussians. Raja and McKenna [6] approximate color with a mixture of Gaussians, and dynamically update it using a weighted sum of previous estimates with estimates based on new data.

The drawback in all these approaches is that they assume that color varies slowly and that it can be predicted by a dynamic model based in only one hypothesis. However, this assumption does not suffice to cope with general scenes, where the dynamics of the color distribution might follow an unknown or unpredictable path.

The main contributions of this paper are summarized below. They are the building blocks of a system, which does not impose constraints on the illuminant color of the scene:

* Corresponding author. Tel.: +34 93 4015791.
E-mail addresses: fmoreno@iri.upc.edu (F. Moreno-Noguer), asanfeliu@iri.upc.es (A. Sanfeliu), samaras@cs.sunysb.edu (D. Samaras).

0262-8856/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2005.10.008

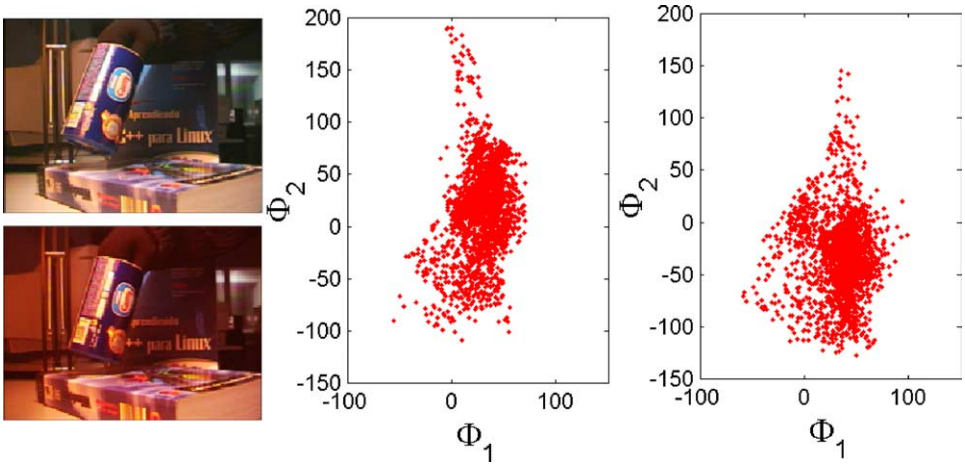

Fig. 1. Abrupt change of illumination. Left: two consecutive frames from a sequence. Light conditions have changed abruptly (from natural to red illuminant). Center and right: corresponding color distributions of the foreground (the can). $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2$ are the coordinates in a 2D colorspace.

- Fisher colorspace: instead of using the classical *RGB*, *rgb*, *XYZ* or *HSV* colorspaces, we propose the use of a colorspace efficient for the discrimination between foreground and background classes. This colorspace will be the 2D projection of the *R*, *G* and *B* components on the plane obtained from a non-parametric *linear discriminant analysis* (LDA) [7].
- Multihypotheses framework: the use of a particle filter formulation to predict the color distribution in subsequent iterations, offers robustness to abrupt and. unexpected changes in the color appearance of the object. In previous work [8], we have suggested a similar multihypotheses framework to track objects in which color could be approximated by a unimodal distribution, represented by a histogram. In the present work, we deal with multicolored objects, approximated by a mixture of Gaussians (*MoG*). Note the difference between our work and all previous tracking approaches using a particle filter formulation (e.g. [2,9,10]). While in these approaches the multihypotheses are formulated about the object position, in our method we formulate the multihypotheses about the color distribution of the object.
- Integration of color and deformable contours in a particle filter framework: the color estimation is used to generate a rough estimation about the object position and remove noisy edges from the image. This simplifies the stage of fitting a deformable contour to the object boundary, and even with a standard *snake* formulation [11], non-rigid objects can be accurately tracked in cluttered backgrounds with abrupt changes of illumination. The fusion of the multihypotheses color model and the deformable contour is done in a final stage that we have introduced to the well-known CONDENSATION algorithm [2].

The basic steps of the algorithm are depicted in the flow diagram of Fig. 2, and in the following sections, a detailed description of each one of the modules will be given. Fisher colorspace is described in Section 2. In Section 3, the object color model and initialization step are presented. The dynamic model for generating multiple hypotheses of the (object and background) color distributions is depicted in Section 4. Section 5 deals with the global and local deformable model fitting process. In Section 6, the complete tracking algorithm and model adaptation is explained in detail, and results and conclusions are presented in Sections 7 and 8, respectively.

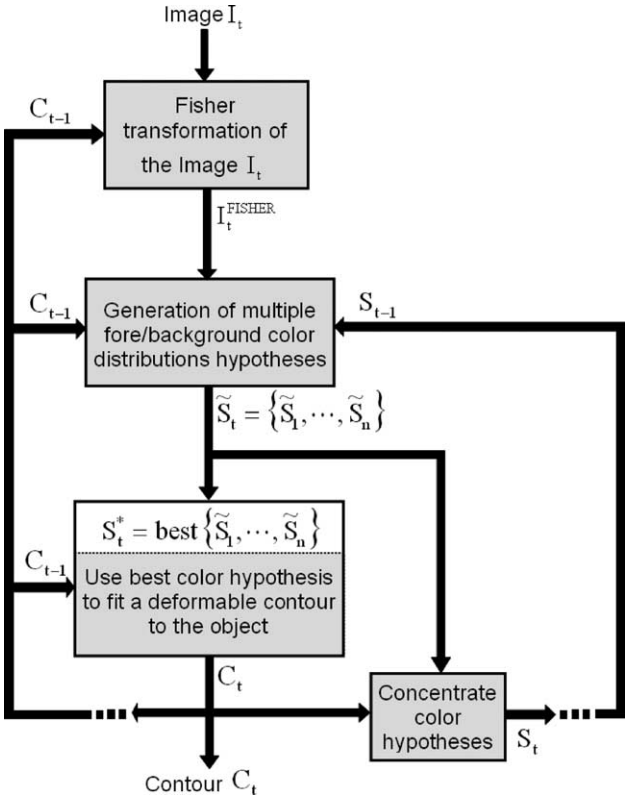

Fig. 2. Flow diagram of the proposed algorithm. I_t is the input *RGB* image at time t. I_t^{FISHER} represents the input image in the Fisher colorspace. $\tilde{S}_t$ and S_t are the set of color distributions of the foreground and background, before and after the 'concentration' stage, respectively. C_t is the resulting contour at time t.

2. Fisher colorspace

The selection of the colorspace is an important initial issue for any color-based figure-ground segmentation system. The typical selection criterion is based on the invariance of the color representation to illumination changes, and according to this idea, color is usually represented by two components of the *rgb*, *HSV* or *xyz*[1] colorspaces. However, these representations are not robust enough to cope with abrupt illumination changes. In this paper, we propose a different criterion and select a 2D colorspace that maximizes the separability of the object and background classes.

Let $\mathbf{x}$ be a 3D vector with the color value of image pixels in *RGB* space, which must be classified as foreground ($\mathcal{O}$) or background ($\mathcal{B}$). When we are dealing with multicolored objects, the parameterization of color distributions in 3D colorspace becomes very complex. To simplify, we reduce the dimensionality to 2D by projecting the data on a plane $\mathbf{\Phi} = [\phi_1, \phi_2] \in \mathcal{M}_{3\times 2}$, that is, $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$, where $\mathbf{y}$ are the linearly transformed 2D coordinates used for classification. The most popular way to find the best linear features is the parametric version of the linear discriminant analysis method [12], where training data is used to construct the within-class S_w and between-class S_b scatter matrices, in the N_c-class problem defined as

$$S_w = \sum_{i=1}^{N_c} P(\mathcal{C}_i) E\left[(\mathbf{x}|_{\mathcal{C}_i} - \mu_i)(\mathbf{x}|_{\mathcal{C}_i} - \mu_i)^\mathrm{T}\right] = \sum_{i=1}^{N_c} P(\mathcal{C}_i) S_i \qquad (1)$$

$$S_b = \sum_{i=1}^{N_c} P(\mathcal{C}_i) E\left[(\mathbf{x} - \mu_o)(\mathbf{x} - \mu_o)^\mathrm{T}\right]$$

where $P(\mathcal{C}_i)$ is the prior of the ith class, μ_i and S_i are its expected value vector and covariance matrix, μ_o is the overall mean and $\mathbf{x}|_{\mathcal{C}_i}$ indicates that sample $\mathbf{x}$ belongs to $\mathcal{C}_i$ class.

A typical criterion for class separability is formulated by the maximization of $J = \mathrm{trace}\ ((\mathbf{\Phi}^\mathrm{T} S_\mathrm{w} \mathbf{\Phi})^{-1} (\mathbf{\Phi}^\mathrm{T} S_\mathrm{b} \mathbf{\Phi}))$, and searches for the separation of the class means in the transformed *Y*-space (high S_b), while at the same time the classes remain compact (small S_w). The classic LDA method maximizes J by constructing the columns of $\mathbf{\Phi}$ with the eigenvectors of $S_w^{-1} S_b$ having the highest eigenvalues.

One of the limitations associated with this approach is that it produces at most $N_\mathrm{c} - 1$ feature projections, i.e. since S_b is computed from only N_c class means, $S_w^{-1} S_b$ will have at most $N_\mathrm{c} - 1$ non-zero eigenvalues, and the maximum dimension of the projected *Y*-space will be $N_\mathrm{c} - 1$. This can be solved by the non-parametric *LDA* [7], that computes S_b using local information and the *k nearest neighbors* (KNN) rule. In the 2-class problem discussed here, this matrix (denoted Σ_b) is defined as

$$\Sigma_b = \frac{1}{N} \sum_{i=1}^{N_f} w_i(\mathbf{x}_i|_{\mathcal{O}} - M_b^k(\mathbf{x}_i|_{\mathcal{O}}))(\mathbf{x}_i|_{\mathcal{O}} - M_b^k(\mathbf{x}_i|_{\mathcal{O}}))^\mathrm{T} + \frac{1}{N} \times \sum_{i=1}^{N_b} w_i(\mathbf{x}_i|_{\mathcal{B}} - M_f^k(\mathbf{x}_i|_{\mathcal{B}}))(\mathbf{x}_i|_{\mathcal{B}} - M_f^k(\mathbf{x}_i|_{\mathcal{B}}))^\mathrm{T} \qquad (2)$$

where N_f and N_b are the number of samples of $\mathcal{O}$ and $\mathcal{B}$, $N = N_f + N_b$, $M_j^k(\mathbf{x}_i)$ is the mean of the k nearest neighbors in class $\mathcal{C}_j$ to a point $\mathbf{x}_i$, and w_i is a weighting function for deemphasizing samples far from the classification boundary (see [7]).

Given two sets $\{\mathbf{x}_{1|\mathcal{O}}, \ldots, \mathbf{x}_{N_f|\mathcal{O}}\}$, $\{\mathbf{x}_{1|\mathcal{B}}, \ldots, \mathbf{x}_{N_b|\mathcal{B}}\}$ of *RGB* pixel values used as training data, the optimum linear mapping is obtained with the following steps:

- Calculate S_w with Eq. (1) and whiten the data with respect to it. That is, transform $\mathbf{x}$ to $\mathbf{z} = \Lambda^{-1/2} \Omega^\mathrm{T} \mathbf{x}$, where Λ and Ω are the eigenvalue and eigenvector matrices of S_w.
- Select k and (in the Z-space) compute Σ_b using Eq. (2).
- Select the two eigenvectors Ψ_1, Ψ_2 of Σ_b with the two largest eigenvalues.
- The optimum linear mapping from the original *RGB* space to the discriminant subspace (we call it *Fisher colorspace*) is given by $\mathbf{y} = \Psi^\mathrm{T} \Lambda^{-1/2} \Omega^\mathrm{T} \mathbf{x}$.

In Fig. 3, we how the concept of Fisher colorspace. In the Section 7 it will be shown that we obtain better rates of class classification using the Fisher colorspace than using other 2D colorspaces.

For the rest of the paper we will represent the pixel values in the Fisher colorspace with the 2D vector $\mathbf{y}$.

3. Color model

After having selected the colorspace, the next step is to choose a model for representing the color distribution of the object and background. For a monochrome object, color histograms have been demonstrated to be an effective technique (e.g. [8]). However, when the object to be modeled contains regions with different colors, the number of pixels representing each color can be relatively low and a histogram representation may not suffice. In this case, a better approach is to use the *MoG* model, that expresses the conditional probability for a pixel $\mathbf{y}$ belonging to a multicolored object $\mathcal{O}$ as a sum of M_o Gaussian components: $p(\mathbf{y}|\mathcal{O}) = \sum_{j=1}^{M_o} p(\mathbf{y}|j) P(j)$. Similarly, the background color will be represented by a mixture of M_b Gaussians.

Given the foreground ($\mathcal{O}$) and background ($\mathcal{B}$) classes, the a posteriori probability that a pixel $\mathbf{y}$ belongs to object $\mathcal{O}$ is computed using the Bayes rule

$$p(\mathcal{O}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathcal{O}) P(\mathcal{O})}{p(\mathbf{y}|\mathcal{O}) P(\mathcal{O}) + p(\mathbf{y}|\mathcal{B}) P(\mathcal{B})} \qquad (3)$$

where $P(\mathcal{O})$, $P(\mathcal{B})$ represent the a priori probabilities of $\mathcal{O}$ and $\mathcal{B}$, respectively. These prior values are approximated to the

[1] When the colorspace is represented by lowercase letters, the sum of the three color components has been normalized to one.

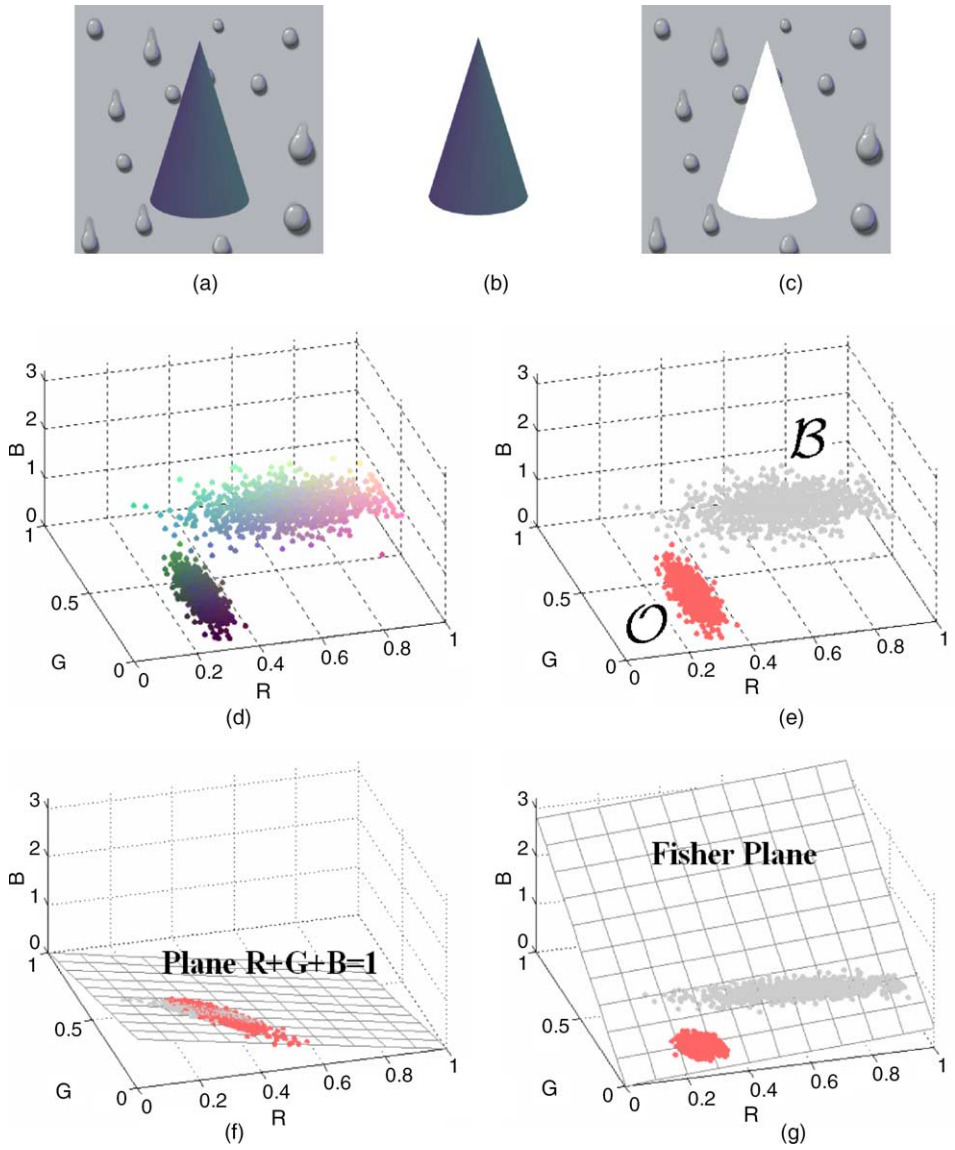

Fig. 3. Fisher Colorspace. (a) Training image. (b) Foreground. (c) Background. (d) Representation of image points in the *RGB* colorspace. (e) Hand-made classification of image points in foreground ($\mathcal{O}$) and background ($\mathcal{B}$) classes. (f) Normalization of colorpoints, equivalent to a projection on the plane $R+G+B=1$. The projected classes are not properly separated. (g) The projection of colorpoints on the Fisher plane gives a better discrimination between the $\mathcal{O}$ and $\mathcal{B}$ classes.

expected area ratios of the foreground and background classes in the image (see Fig. 4).

As in the problem of selecting the number of bins in histogram models, using *MoG* conceals the challenge of choosing the number of Gaussian components that better adjust the data. We initialize this, with the modified *EM* algorithm proposed in [13], that is based on a *minimum message length* criterion and iteratively fits and annihilates an initially large number of components (introduced by the user).

The initial configurations of the *MoG* for $\mathcal{O}$ and $\mathcal{B}$, after learning, are parameterized by

$$\mathcal{G}_\varepsilon = [\mathbf{p}_\varepsilon, \mu_\varepsilon, \lambda_\varepsilon, \theta_\varepsilon] \tag{4}$$

where $\varepsilon = \{\mathcal{O}, \mathcal{B}\}$, $\mathbf{p}_\varepsilon$ contain the priors for each Gaussian component, μ_ε the centroids, λ_ε the eigenvalues of the principal directions and θ_ε the angles between the principal directions with the horizontal. $\mathcal{G} = \{\mathcal{G}_\mathcal{O}, \mathcal{G}_\mathcal{B}\}$ will be the state vector representing the color model.

4. Dynamic color model

Let $\mathcal{Y}_{\varepsilon,t-1} = [\mathbf{y}_{1,t-1}, \ldots, \mathbf{y}_{N_\varepsilon,t-1}]^T$, be the vectors containing the set of points (in Fisher colorspace coordinates) belonging to the classes $\mathcal{O}$ and $\mathcal{B}$, at time $t-1$. The third stage of the tracking algorithm (see Section 6), consists of propagating the components $\mathcal{G}_{\varepsilon,t-1}$ of the state vector to $\tilde{\mathcal{G}}_{\varepsilon,t}$, given a specific dynamical model and the image at time t, denoted as $\mathbf{Z}_t$. Instead of applying the dynamic model directly to $\mathcal{G}_{\varepsilon,t-1}$, we apply it to the distribution $\mathcal{Y}_{\varepsilon,t-1}$, to obtain the estimation $\tilde{\mathcal{Y}}_{\varepsilon,t}$, that will be used later to reestimate $\tilde{\mathcal{G}}_{\varepsilon,t}$. With this aim we define the following *affine random* dynamic model:

$$\mathcal{Y}_{\varepsilon,t} = \mathcal{A}_\varepsilon \mathcal{Y}_{\varepsilon,t-1} + \mathbf{v}_\varepsilon$$

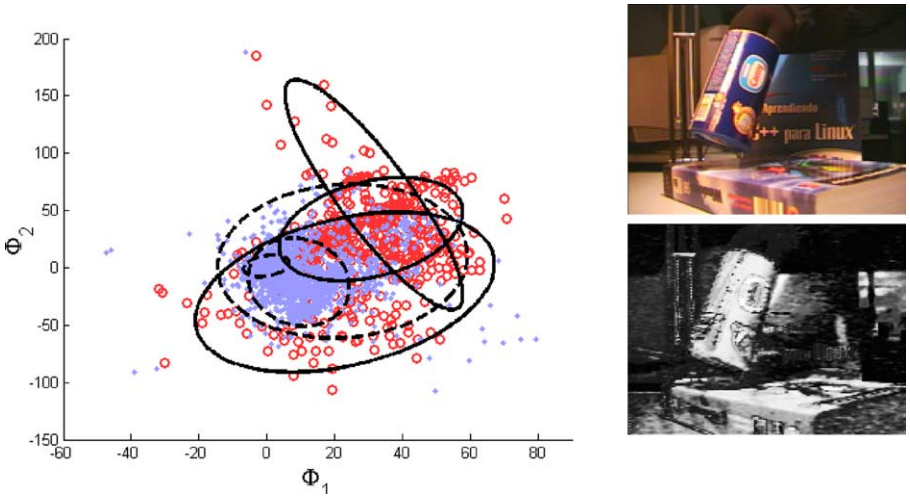

Fig. 4. Gaussian mixture components of $\mathcal{O}$ (the can) and $\mathcal{B}$. Left mage: solid dots and lines are $\mathcal{O}$ data points (in Fisher colorspace) and the variances of the Gaussian components, respectively. Hollow dots and dashed lines are $\mathcal{B}$ data and Gaussians. Lower right image: $p(\mathcal{O}|\mathbf{y})$ where brighter points correspond to more likely pixels.

In the case of representing color distributions in a 2D colorspace, matrix $\mathcal{A}_\varepsilon$ and translation vector $\mathbf{v}_\varepsilon$ are written as:

$$\mathcal{A}_\varepsilon = \begin{bmatrix} 1+a_{\varepsilon,11} & a_{\varepsilon,12} \\ a_{\varepsilon,21} & 1+a_{\varepsilon,22} \end{bmatrix} \quad \mathbf{v}_\varepsilon = \begin{bmatrix} v_{\varepsilon,1} \\ v_{\varepsilon,2} \end{bmatrix}$$

Variables $a_{\varepsilon,ij}$ and $v_{\varepsilon,i}$ are approximated by normal random distributions, $a_{\varepsilon,ij} \sim \mathcal{N}(0,\sigma_{a_{\varepsilon,ij}}), v_{\varepsilon,i} \sim \mathcal{N}(\mu_{v_{\varepsilon,i}},\sigma_{v_{\varepsilon,i}})$. The parameters $\sigma_{a_{\varepsilon,ij}}$ and $\sigma_{v_{\varepsilon,i}}$ are learned a priori by a least-squares procedure, from a training hand segmented sequence of the object when still, under an illumination change. It is interesting to point out that even if the testing sequences were not available, the variances $\sigma_{a_{\varepsilon,ij}}$ and $\sigma_{v_{\varepsilon,i}}$ could be empirically set to values sufficiently high in order to cope with abrupt changes of illumination. In that case, however, the number of particles should be increased, since they should sample a wider area of the state space. With respect to the rest of parameters, $\mu_{v_{\varepsilon,i}}$ accounts for the expected displacement between the class distributions in $t-1$ and t, and is approximated by the translation vector between the centroids of the sets $\mathcal{Y}_{\varepsilon,t-1}$ and $\mathcal{Y}_t$. Note that the vector $\mathcal{Y}_t = [\mathcal{Y}_{\mathcal{O},t}, \mathcal{Y}_{\mathcal{B},t}]^T$ representing the color distribution of the whole image $\mathbf{Z}_t$ is known, but the subsets $\mathcal{Y}_{\mathcal{O},t}$ and $\mathcal{Y}_{\mathcal{B},t}$ are unknown.

Using the *EM* algorithm initialized on $\mathcal{G}_{\varepsilon,t-1}$, a new mixture of Gaussians $\tilde{\mathcal{G}}_{\varepsilon,t}$, is fitted to each predicted distribution $\tilde{\mathcal{Y}}_{\varepsilon,t}$, and used to compute the a posteriori probability maps for the foreground class, following Eq. (3). In Fig. 5 we show several hypotheses (with the corresponding $p(\mathcal{O}|\mathbf{y})$ maps) used to estimate the abrupt change of illumination that exists in the pair of images of Fig. 5. Observe that some of the hypotheses are able to provide a 'good' foreground/background discrimination.

5. Global and local deformable model fitting

As color segmentation usually only gives a rough estimation about the object location, we use a deformable model [10,14] to fit its boundary and obtain more precise information about its position. This process is highly simplified by using the data that is estimated by the color model (Section 4) in order to preprocess the contour image and to remove those noisy edges that might disturb the deformable model fitting process. This simplification allows us to obtain good tracking results in rigid and non-rigid objects, even when using the simple well-known snake algorithm [11]. During the boundary adjustment process, first a global fit of an affine contour is performed, which deals with object translation and orientation (rigid motion), followed by local deformations that apply to non-rigid motions. The following are some details of these processes:

Let the contour of the object be parameterized by a curve $\mathbf{r}(s) = [u(s), v(s)]$, $s \in [0,1]$, that moves through the image. In the traditional *snake* formulation [11], the problem of snake fitting can be viewed as a force balance equation

$$\mathbf{F}_{\text{int}}(\mathbf{r}(s)) + \mathbf{F}_{\text{ext}}(\mathbf{r}(s)) = 0 \tag{5}$$

where $\mathbf{F}_{\text{int}}(\mathbf{r}(s)) = \alpha(\partial^2\mathbf{r}(s)/\partial s^2) + \beta(\partial^4\mathbf{r}(s)/\partial s^4)$ are the internal forces that control the bending and stretching of the snake (α and β are the elasticity and rigidity parameters, respectively). $\mathbf{F}_{\text{ext}}(\mathbf{r}(s))$ are the external forces that pull the curve towards the edge image features. In the literature, there exist several definitions for this external function. In particular, we use the *gradient vector flow* (GVM) external force proposed in [15], because it has a larger capture range and better convergence performance in boundary concavities than other methods.

Eq. (5) is solved by making the snake a function of both space and time, i.e. $\mathbf{r}(s, t)$ (we will write $\mathbf{r}_t$) and iterating over the following expression:

$$\frac{\mathbf{r}_t - \mathbf{r}_{t-1}}{\Delta t} = \alpha\frac{\partial^2\mathbf{r}_{t-1}}{\partial s^2} + \beta\frac{\partial^4\mathbf{r}_{t-1}}{\partial s^4} + \mathbf{F}_{\text{ext}}(\mathbf{r}_{t-1})$$

When the solution stabilizes ($r_{t-1} = r_t$), Eq. (5) is satisfied.

For the numerical implementation we approximate the derivatives with finite differences, and discretize the curve $\mathbf{r}(s, t)$ with N_P points, so that the previous gradient descent

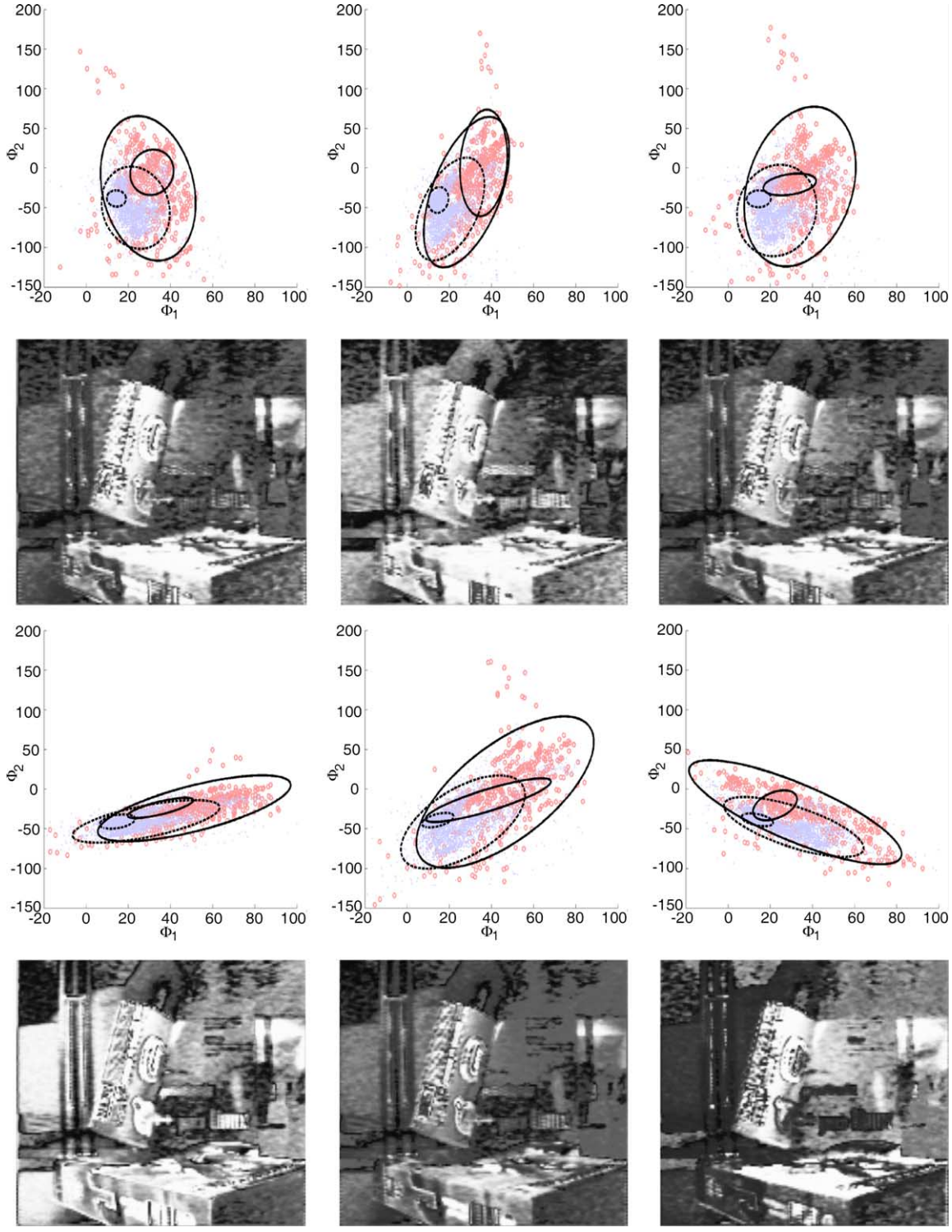

Fig. 5. Several hypotheses and their respective $p(\mathcal{C}|\mathbf{y})$ map, corresponding to the abrupt illumination transition presented in Fig. 1.

method can be rewritten as

$$\mathcal{R}_t = (I - \Delta t \mathcal{Q})^{-1}\left(\mathcal{R}_{t-1} + \Delta t \mathbf{F}_{\text{ext}}(\mathcal{R}_{t-1})\right) \tag{6}$$

where $\mathcal{R} = [(u_1, v_1, 1), \ldots, (u_{N_P}, v_{N_P}, 1)]^T$ contains the homogeneous coordinates of the N_P discretized points of the snake, $\mathcal{Q}$ is a $N_P \times N_P$ pentadiagonal matrix including the α and β parameters, and I is the N_P-identity matrix.

Iterating over Eq. (6) the snake is locally fitted to the image edges, governed only by its internal and external forces.

However, previous to local fitting stage we perform a global deformation in order to find the suitable translation and orientation of the snake. For this fitting, the following additional constraint of affine deformation is introduced to Eq. (6):

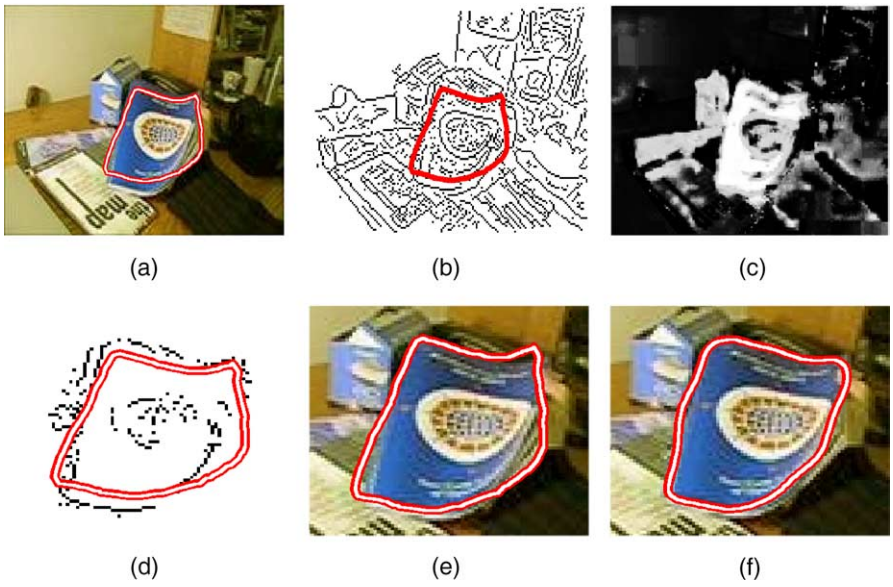

Fig. 6. Global and local fitting procedures: (a) Original image and contour from previous iteration. (b) Edge features image. The process of contour fitting in such an image is quite difficult because of noisy edges. (c) Foreground a posteriori probability map obtained using the color module. (d) Refined edge image (foreground detail). (e) Contour fitted after global deformations. (f) Contour fitted after local deformations.

$$\mathcal{R}_t = \mathcal{R}_{t-1}\mathcal{H}_A = \mathcal{R}_{t-1}\begin{bmatrix} a_{11} & a_{12} & v_1 \\ a_{21} & a_{22} & v_2 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Combining Eqs. (6) and (7), we obtain the following iterative procedure for the affine snake deformation:

(1) $\mathcal{H}_A = (\mathcal{J}^T\mathcal{J})^{-1}\mathcal{J}^T(\mathcal{R}_{t-1} + \mathbf{F}_{\text{ext}}(\mathcal{R}_{t-1}))$ where $\mathcal{J} = \mathcal{R}_{t-1} - \Delta t \mathcal{S}\mathcal{R}_{t-1}$
(2) Normalize $\mathcal{H}_A$ using the component $\mathcal{H}_A(3,3)$ Set $\mathcal{H}_A(3,1) = \mathcal{H}_A(3,2) = 0$
(3) $\mathcal{R}_t = \mathcal{R}_{t-1}\mathcal{H}_A$

Steps 1–3 are iterated until the convergence of R_t and R_{t-1}. In Fig. 6, we show the results of the global and local fitting in a non-rigid movement.

6. Tracking algorithm

In this Section, we will use the tools described previously to explain in detail the whole method for tracking rigid and non-rigid objects in a cluttered environment and under changing illumination. The basic steps of the tracking algorithm follow the particle filter procedure, but we introduce a modification to the classic CONDENSATION algorithm (analogous to the ICONDENSATION technique [2]), and in order to 'direct' the search for the next iteration we add a final stage that concentrates the future hypotheses on those areas of the state-space containing more information about $p(\mathcal{O}|\mathbf{y})$ (see Fig. 7). Moreover, in this final stage we fuse object color and shape information to obtain precise results about object pose. Next, we present the steps of our algorithm:

(1) Probability density function of the color point set: at time t, a set of N samples $\mathcal{S}_{t-1}^{(n)}$ $(n=1, \ldots, N)$ with the same structure as $\mathcal{G}$ (Eq. (4)), is available from previous iteration. This set, parameterizes N color distributions. Each sample has an associated weight $\pi_{t-1}^{(n)}$ and a classification $\mathcal{Y}_{t-1}^{(n)} = [\mathcal{Y}_{\mathcal{O},t-1}^{(n)}, \mathcal{Y}_{\mathcal{B},t-1}^{(n)}]^T$ of the image colorpoints in the foreground and background sets. The whole set represents an approximation to $p(\mathcal{G}_{t-1}|\mathcal{Z}_{t-1})$ where $\mathcal{Z}_{t-1} = \{\mathbf{Z}_0, \ldots, \mathbf{Z}_{t-1}\}$ is the history of the images. The algorithm aims to construct a new sample set $\{\mathcal{S}_t^{(n)}, \pi_t^{(n)}\}$ to estimate $p(\mathcal{G}_t|\mathcal{Z}_t)$.
(2) Sampling from $p(\mathcal{G}_{t-1}|\mathcal{Z}_{t-1})$: A sampling with replacement is performed N times on the set $\{\mathcal{S}_{t-1}^{(n)}\}$, where each element has probability $\pi_{t-1}^{(n)}$ of being chosen. This will give us a set $\{\mathcal{S}_{t-1}^{\prime(n)}\}$.
(3) Probabilistic propagation of the samples: each sample $\mathcal{S}_{t-1}^{\prime(n)}$ is propagated to $\tilde{\mathcal{S}}_t^{(n)}$, using the dynamic model explained in Section 4. Note that, as it was pointed out, the dynamic model is not directly applied to the *MoG*'s parameters of the color distributions, but rather, to the associated color distribution points. Subsequently, the predicted color distribution points are used to compute the corresponding *MoG*'s parameters.
(4) Measure and weight: each element $\tilde{\mathcal{S}}_t^{(n)}$, has to be weighted according to some measured features. Based on the propagated *MoG* samples $\tilde{\mathcal{S}}_t^{(n)}$ we compute $p(\mathcal{O}|\mathbf{y})$ for the whole image using the Bayes rule (Eq. (3)). With this probability map, we assign the following weight to each sample:

$$\pi_t^{(n)} = \frac{\sum_{y\in W} p(\mathcal{O}|\mathbf{y})}{N_w} - \frac{\sum_{y\notin W} p(\mathcal{O}|\mathbf{y})}{\overline{N_w}}$$

where W is the interest region around the previous object position (where we predict the object will be), and N_w, $\bar{N}_w$

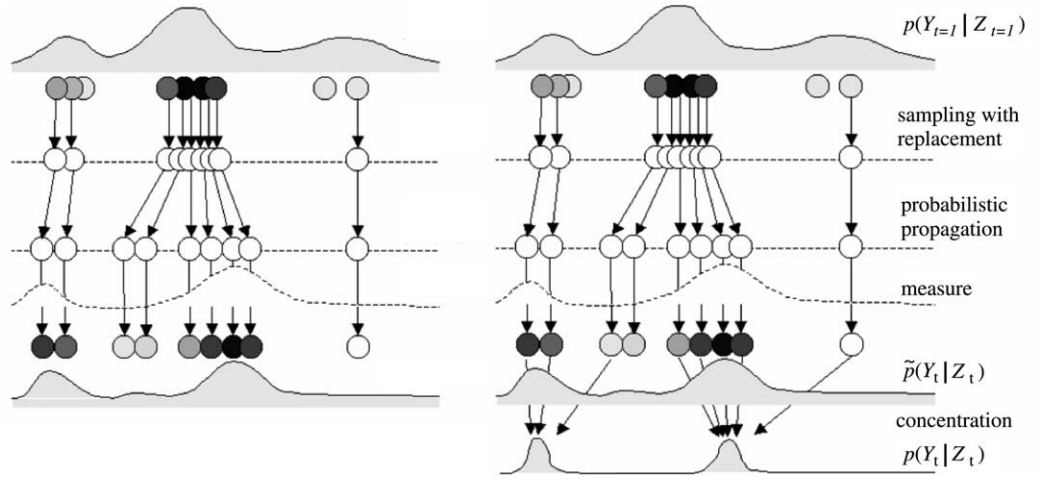

Fig. 7. Left: steps of the classic CONDENSATION algorithm (Figure adapted from [10]). Right: in our implementation, we have included a final stage called 'sample concentration'.

are the number of image pixels in and out of this interest region, respectively.

(5) Sample concentration: in the last stage of our algorithm, we concentrate the samples around the local maxima, so that in the subsequent iteration the hypotheses are formulated around these more likely regions of the state space. In our case, this is absolutely necessary because the state vector $\mathcal{G}$ has high dimensionality, and if we let the samples move freely, uniquely governed by the dynamic model, the number of hypotheses needed to find the samples representing a correct color configuration, is extremely high.

The concentration is performed by taking the sample with maximum weight, $\pi_t^* = \max\{\pi_t^1,\ldots,\pi_t^{(n)}\}$ and based on the a posteriori map generated by this sample, the object of interest is accurately segmented from the image using the deformable model fitting procedure explained in Section 5. The various substeps of this stage, can be summarized as follows:

(a) Using morphologic operations on the probability map image, a coarse approximation of object shape is obtained that allows us to eliminate noisy edges from the original image (Fig. 6b,c,d).
(b) The contour of the object in the previous iteration, is used as initialization of an affine snake, that is adjusted (only by affine deformations) to the image of refined edges (Fig. 6e) in order to solve the global deformation. Next, to cope with non-rigid deformations the process is repeated with a non-affine snake (Fig. 6f).
(c) Once the boundary of the object has been accurately detected, the color estimates are refined. Inner image pixels are separated from outer pixels and the vector $\mathcal{Y}_t^* = [\mathcal{Y}_{\mathcal{O},t}^*, \mathcal{Y}_{\mathcal{B},t}^*]^T$ is generated. Mixtures of Gaussians are fitted

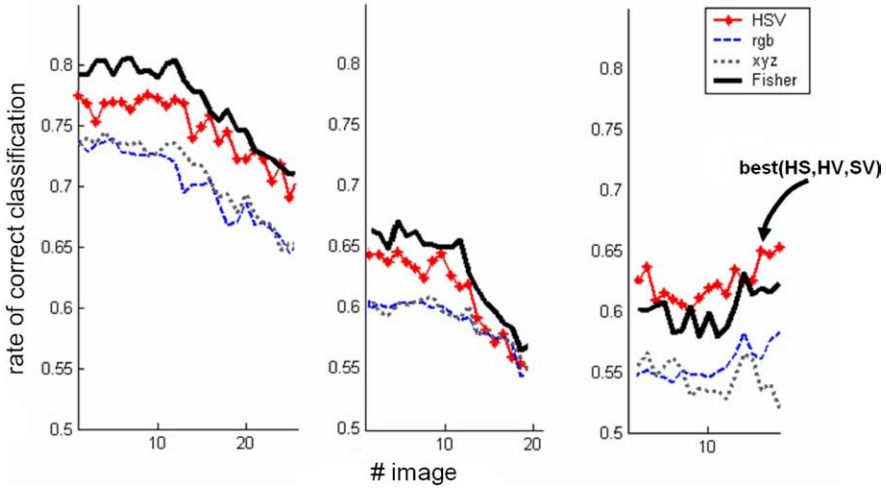

Fig. 8. Classification results of image pixels into classes $\mathcal{O}$ and $\mathcal{B}$ using different colorspaces. The horizontal axis represents the image index for each one of the experiments, and the vertical axis represents the percentage of pixels correctly classified. Note that the results for the *HSV*, *rgb* and *xyz* colorspaces, correspond to the best result obtained when we pick two of the three components of these colorspaces.

to these color distributions (using the EM algorithm), giving a state vector $\mathcal{S}_t^*$, around which samples $\left\{\tilde{\mathcal{S}}_t^{(n)}\right\}$ are 'concentrated' with the equation $S_t^{(n)} = (1-a)\tilde{S}_t^{(n)} + a\mathcal{S}_t^*$, where the parameter a governs the level of concentration. Similarly, weights $\left\{\pi_t\ (n)\right\}$ and distributions $\left\{\tilde{\mathcal{Y}}_t^{(n)}\right\}$ associated to these samples, are concentrated around π_t^* and $\mathcal{Y}_t^*$.

7. Experimental results

We initially compare the class discrimination power of the Fisher colorspace to that of other colorspaces (two components of the *rgb*, *HSV* or *xyz*). To quantify the notion of class separability, a constant number of Gaussians are fitted to foreground and background distributions of hand segmented images, for each one of the colorspaces. Next, according to Eq. (3) we segment the same images, assigning each pixel $\mathbf{y}$ to the class with maximal $p(\mathcal{O}|\mathbf{y})$ The hand segmented image is used as ground truth to evaluate the rate of correctly classified points. We have performed this test on three different video sequences, undergoing a change of illumination. In Fig. 8, we plot the results of these experiments, where the vertical axis represents the percentage of pixels well classified. Taking into account the mean of all the sequences, the best results are obtained with the Fisher colorspace with a 68.1% of pixels correctly classified, followed by the *HV* components of *HSV* colorspace with a 67.1% rate. Although the difference is not significant, the main advantage of the Fisher colorspace is that it directly provides the best linear transformation of the data. If we used some other color space (*rgb*, *HSV* or *xyz*), we would not know a priori which combination of 2D coordinates are the best for a specific problem of fore/background segmentation. In Table 1, we show the results of the complete experiment.

Next, two different experimental results are presented in order to illustrate the robustness of our system to several changing conditions of the environment. Since the algorithm has been implemented in an interpretative language (MATLAB), we cannot discuss time performance issues, instead we focus on the effectiveness of the method. Time performance depends linearly on the number of hypotheses used to estimate the color distributions.

In the first experiment, we track the boundary of a bending book (non-rigid motion) in a video sequence where the lighting conditions change smoothly from natural lighting to yellow lighting. In this case, as the displacement of the color distribution in color space was relatively small, we have used 'only' 5 hypotheses. Fig. 9 shows some frames of the sequence with the obtained results, the corresponding edge images and the a posteriori probability maps of the foreground (the book). The sequence of edge images contains a lot of noisy boundaries that pose difficulties for the tracking process and for the adjustment of a deformable model to the edges of the object. However, the integration with color information gives a first estimate of the object position, that allows us to eliminate many false edges and reduce the complexity of the deformable model fitting procedure.

Whereas in the first experiment, we demonstrate the need for integration of the different vision modules, in the second experiment, we demonstrate the need for a multihypotheses model to face abrupt changes in the illuminant. For this experiment, we have computed the prediction of the color distribution using 20.hypotheses. In Fig. 10, we compare the results obtained using a smooth color dynamic model and our multihypotheses model, for a rigid object moving in an environment in which the lighting changes abruptly. The *MoG* for frame t predicted by the smooth model, is based on a weighting function $\mathcal{G}_t = (1-a)\mathcal{G}_{t-2} + a\mathcal{G}_{t-1}$, where $\mathcal{G}$ is the parameterization of the color distribution and a is the mixing factor. Results prove the inability of the smooth color model to predict the change (the a posteriori probability map of the foreground region does not discriminate between fore and background, Fig. 10e) whereas a good result is obtained with the method proposed in the paper (where simple morphologic operations over the a posteriori probability map, allow obtain a good estimation of the object position, Fig. 10f). In Fig. 11, we show similar results for the contour tracking of a non-rigid object under an abrupt change of illumination.

Table 1
Details of the results presented in Fig. 8

Colorspace	Seq. 1		Seq. 2		Seq. 3		Mean	
	μ	σ	μ	σ	μ	σ	μ	σ
Fisher	0.770	0.033	0.630	0.035	0.602	0.016	0.681	0.082
HV	0.748	0.025	0.609	0.035	0.624	0.016	0.671	0.071
HS	0.706	0.033	0.587	0.017	0.554	0.012	0.628	0.072
Rg	0.709	0.031	0.589	0.018	0.545	0.014	0.627	0.076
Xy	0.701	0.030	0.587	0.019	0.557	0.013	0.627	0.069
Rb	0.705	0.033	0.589	0.017	0.546	0.009	0.626	0.074
Gb	0.707	0.032	0.588	0.019	0.543	0.009	0.626	0.076
Yz	0.694	0.032	0.583	0.020	0.557	0.013	0.622	0.067
Xz	0.694	0.029	0.582	0.020	0.557	0.013	0.622	0.060
Sv	0.628	0.045	0.571	0.022	0.561	0.013	0.592	0.045

Each column represents the average over all images in a single experiment, and the last column is the mean of the three experiments. Every value, is the percentage of pixels correctly classified (mean and variance), using a particular colorspace.

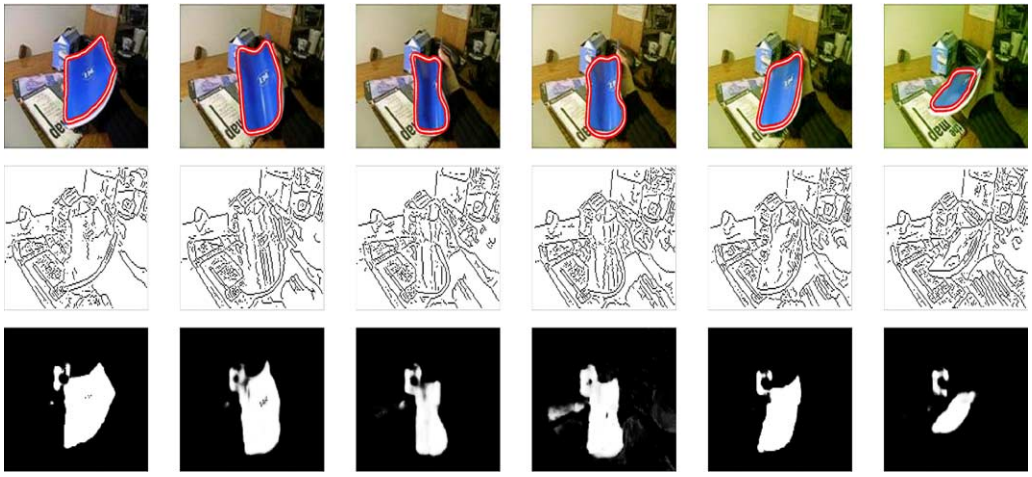

Fig. 9. Tracking results of a non-rigid object in a sequence with smooth lighting changes. First row: tracking results. The tracked contour is superimposed on the original images. Second row: edge map. The task of fitting a deformable model to the contour of the object is extremely difficult because of the presence of noisy edges. Third row: foreground a posteriori map obtained using the proposed multihypotheses color model. This map provides a rough estimate of the object position and removes most of the noisy edges, so that the deformable contour fitting procedure is highly simplified.

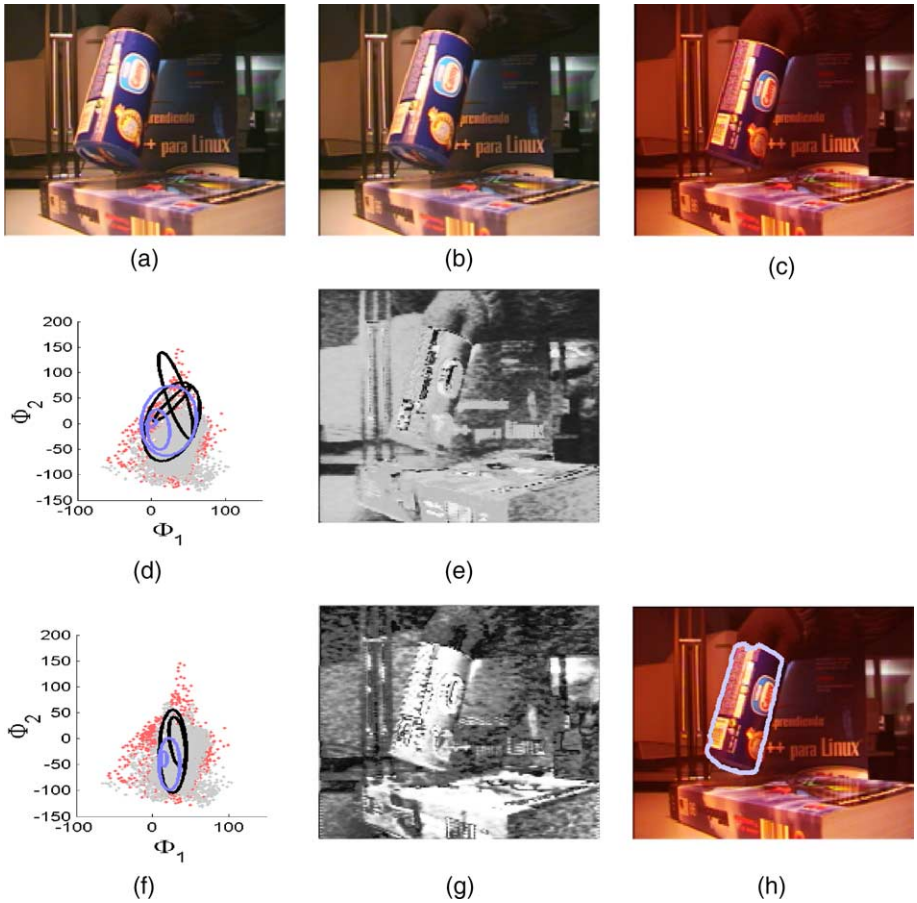

Fig. 10. Performance comparison of a smooth prediction color dynamic model and the multihypotheses one, for an abrupt change in illumination and rigid object motion. (a), (b), (c) Frames $t-2$, $t-1$ and t are three consecutive images of the sequence. Note the abrupt change in illuminant between frames $t-1$ and t. (d) Ellipses correspond to the foreground and background *MoG* predicted with a smooth color dynamic model. The real distributions of points in colorspace are also shown. (e) $p(\mathcal{O}|\mathbf{y})$ map obtained with the smooth model. There is no good discrimination between fore and background. (f) *MoG* of the best sample using the multihypotheses color dynamic model. (g) $p(\mathcal{O}|\mathbf{y})$ map obtained with this color model. There is good fore/background discrimination. (h) Tracking results obtained after using $p(\mathcal{O}|\mathbf{y})$ to eliminate false edges from image and fitting a deformable contour.

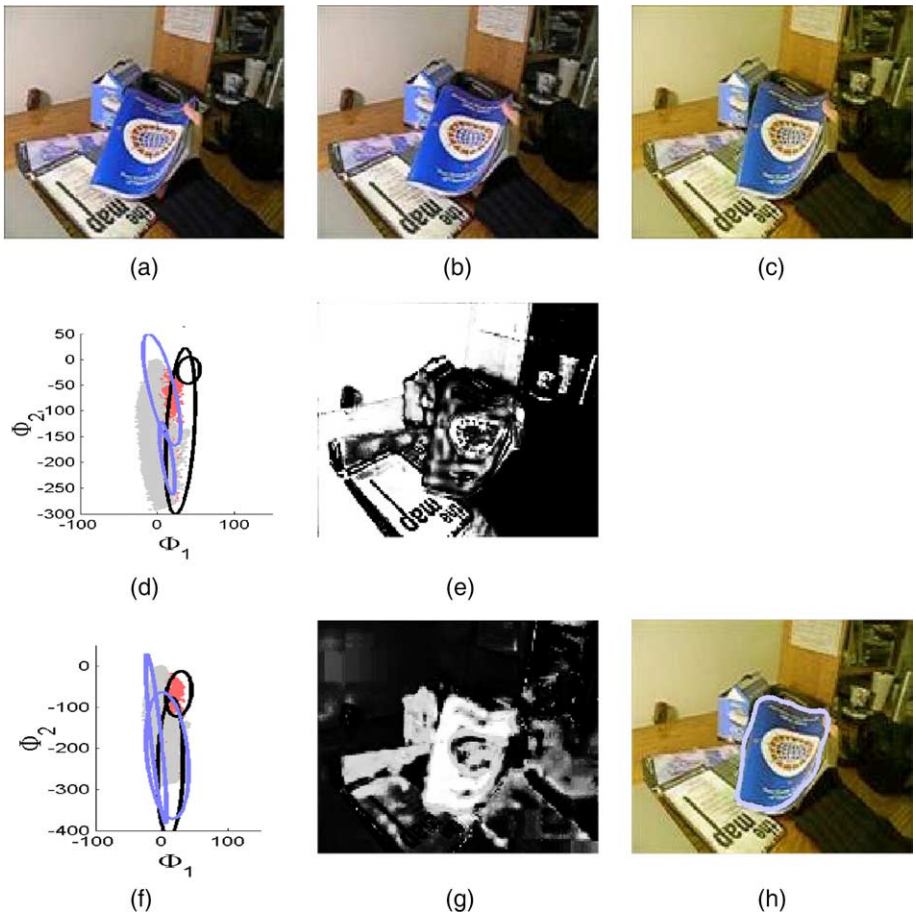

Fig. 11. Performance comparison of a smooth prediction color the multihypotheses one, for an abrupt change in illumination motion. See Fig. 10 for interpretation of results.

8. Conclusions

Most of the methods that adapt color are based on the assumption of smooth change on the color model, so that the predicted color of the target is computed based on a weighting function of previous color distributions. In this paper, we have presented a method where this constraint is no longer needed, and the dynamic model is based on the formulation of multiple hypotheses about the next state of the target color distribution. The best of these hypotheses is used to obtain a rough estimate of the object position, and eliminate false and noisy edges, so that the task of fitting a deformable contour to the object boundary is considerably simplified. Reciprocally, this boundary is used to refine the color estimation. Moreover, we propose the use of the Fisher colorspace, that has a better object/background discrimination performance than typical colorspaces. The algorithm has been used to obtain a precise figure-ground segmentation in rigid and non-rigid objects, moving in an environment with abrupt light changes (where smooth dynamic color models fail). In future work, we plan to integrate the parameters of the Fisher plane into the particle filter formulation, and also adapt the Fisher plane to abrupt changes of illumination. Furthermore, we plan to continue this work by integrating other cues such as texture and optical flow techniques to improve the robustness of the method and apply our multihypotheses framework into tracking of objects in 3D.

Acknowledgements

This work was supported by CICYT projects DPI2001-2223 and DPI2000-1352-C02-01, and by a fellowship from the Spanish Ministry of Science and Technology.

References

[1] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, IEEE Conference on Computer Vision and Pattern Recognition, 1998, pp. 232–237.
[2] M. Isard, A. Blake, Icondensaton: unifiying low-level and high-level tracking in a stochastic framework, Proceedings of the ECCV, 1996, pp. 893–908.
[3] G.D. Finlayson, B.V. Funt, K. Barnard, Color constancy under varying illumination, Proceedings of the International Conference on Computer Vision, 1995, pp. 720–725.
[4] L. Sigal, S. Sclaroff, V. Athitsos. Estimation and prediction of evolving color distributions for skin, segmentation under varying illumination, IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, 2000, pp. 152–159.
[5] J. Yang, W. Lu, A. Waibel, Skin-color modeling and adaption, Proceedings ACCV, vol. 2, 1998, pp. 687–694.
[6] Y. Raja, S. McKenna, S. Gong, Colour model selection and adaption in dynamic scenes, Proceedings of the ECCV, vol. 1, 2000, pp. 460–475.

[7] K. Fukunaga, Introduction to Statistical Pattern Recognition, second ed., Academic Press, San Diego, CA, 1990.
[8] F. Moreno-Noguer, J. Andrade-Cetto, A. Sanfeliu, Fusion of color and shape for object tracking under varying illumination, Proceedings IBPRIA, LNCS 2652, Springer, 2003. pp. 580–588.
[9] K. Nummiaro, E. Koller-Meier, L. Van Gool, An adaptive color-based particle filter, Image and Vision Computing 2 (1) (2003) 99–110.
[10] A. Blake, M. Isard, Active Contours, Springer, 1998.
[11] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, International Journal of Computer Vision 1 (1987) 321–331.
[12] R.O. Duda, P.E. Hart, Pattern Recognition and Scene Analysis, Wiley, New York, 1973.
[13] M.A.T. Figueiredo, A.K. Jain, Unsupervised learning of finite mixture models, IEEE Transaction on Pattern Analysis and Machine Intelligence 24 (3) (2002) 381–396.
[14] T. McInerney, D. Terzopoulos, Deformable models in medical image analysis: a survey, Medical Image Analysis 1 (2) (1996) 91–108.
[15] C. Xu, J.L. Prince, Snakes, shapes, and gradient vector flow, IEEE Transactions on Image Processing 7 (3) (1998) 359–369.