

Adaptive Color Model for Figure-Ground Segmentation in Dynamic Environments*

Francesc Moreno-Noguer and Alberto Sanfeliu

Institut de Robòtica i Informàtica Industrial, UPC-CSIC
Llorens Artigas 4-6, 08028, Barcelona, Spain
{fmoreno, asanfeliu}@iri.upc.es

Abstract. In this paper we propose a new technique to perform figure-ground segmentation in image sequences of scenarios with varying illumination conditions. Most of the algorithms in the literature that adapt color, assume smooth color changes over time. On the contrary, our technique formulates multiple hypotheses about the next state of the color distribution (modelled with a Mixture of Gaussians -*MoG*-), and validates them taking into account shape information of the object. The fusion of shape and color is done in a stage denominated 'sample concentration', that we introduce as a final step to the classical CONDENSATION algorithm. The multiple hypotheses generation, allows for more robust adaptations procedures, and the assumption of gradual change of the lighting conditions over time is no longer necessary.

1 Introduction

Color is a visual cue that is commonly used in computer vision applications, such as object detection and tracking tasks. In environments with controlled lighting conditions and uncluttered background, color can be considered a robust and invariant cue, but when dealing with real scenes with changing illumination and confusing backgrounds, the apparent color of the objects varies considerably over time. Thus, an important challenge for any tracking system to work in real unconstrained environments, is the ability to accommodate these changes. 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. [2]). But in practice, these methods work mostly on 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, usually based on weighting functions of previous color distributions [6][7][8]. The drawback in all of 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 is not enough to cope with general scenes, where the dynamics of the color distribution might follow an unknown and unpredictable path.

* 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.



Fig. 1. Example frames of a sequence with time-varying color illuminant and its corresponding color distributions (in normalized rg color space).

In order to cope with these drastic changes, we propose a framework that uses multiple hypotheses about the future state of the color distribution. In a previous work [5], we have suggested a similar multihypotheses framework to track objects in which color could be approximated by an unimodal distribution, represented by a histogram. The main contribution of the present paper consists of applying the fusion of shape and color information in a final stage of the CONDENSATION algorithm (that we call ‘sample concentration’) in order to deal with multicolored objects. To achieve this, the color of the object (and background) has been approximated by a MoG , which number is automatically initialized by an unsupervised algorithm. At each iteration, an offline learned dynamic model will generate the hypotheses about probable future states of the Gaussian mixture, that will be weighted depending on ‘the quality’ of the a posteriori probability map of the object computed with each of them.

A detailed description of the method is explained in the following sections. In Section 2 the object color model and initialization is given. The process of adjusting the color parameters over time to a dynamic model is explained in Section 3. And next, in Section 4, the complete tracking algorithm and model adaption is described in detail. Results and conclusions are presented in Sections 5 and 6, respectively.

2 Color Model

In order to represent the color distribution of a monochrome object, color histograms have been demonstrated to be an effective technique (e.g. [5]). However, when the object to be modeled contains regions with different color, 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 Gaussian Mixture models. The conditional probability for a pixel \mathbf{x} belonging to a multi-colored object \mathcal{O} is expressed as a sum of M_o Gaussian components:

$$p(\mathbf{x}|\mathcal{O}) = \sum_{j=1}^{M_o} p(\mathbf{x}|j) P(j) \quad (1)$$

Similarly, the background color will be represented by a mixture of M_b Gaussians.

2.1 Model Order Selection

Similar to the problem of selecting the number of bins in histogram models, using MoG conceals the challenge of choosing the number of Gaussian components that

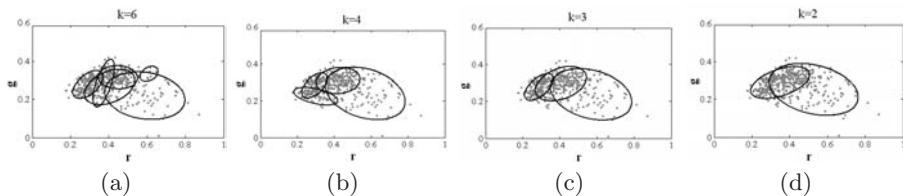


Fig. 2. Fitting a *MoG* to color data in *rg* colorspace. (a) Initialization with $K = 6$ components. (b),(c),(d) are three intermediate steps. The best estimate is the one corresponding to $K = 4$ components.

better adjust the data. In [7], the model order is selected by iteratively applying the EM algorithm and splitting those components having lower a posteriori probability $p(\mathcal{O}|\mathbf{x})$. We have observed that this method generates too many components, some of them unnecessary, increasing the computational cost of the segmentation stage.

We suggest the use of the method proposed in [3], based on a *Minimum Message Length* (MML) criteria that is implemented by a modified EM algorithm. This algorithm performs much more stable and generates an initial set with a lower number of gaussian components than in [7]. In Fig. 2, we show several steps of the fitting process. The algorithm begins with a large number of components (introduced by the user), and iteratively performs an annihilation of those components that are not supported by the data.

2.2 Figure-Ground Segmentation

In order to segment the object of interest from the background we model both color distributions by *MoGs* and compute the a posteriori probability that a pixel \mathbf{x} belongs to the object using the Bayes rule:

$$p(\mathcal{O}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{O})P(\mathcal{O})}{p(\mathbf{x}|\mathcal{O})P(\mathcal{O}) + p(\mathbf{x}|\mathcal{B})P(\mathcal{B})} \quad (2)$$

where \mathcal{B} refers to the background, and $P(\mathcal{O})$, $P(\mathcal{B})$ represent the prior probability of object and background, respectively. These values are approximated to the expected area of the object in the search region (Fig. 3).

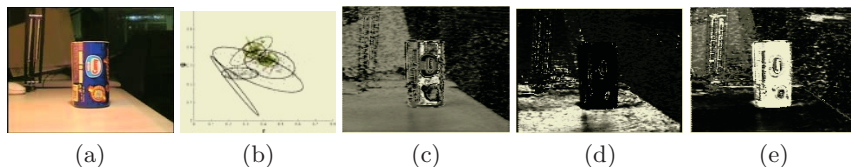


Fig. 3. *MoGs* of \mathcal{O} (the can) and \mathcal{B} , and probability density maps. (a) Original image. (b) Crosses and dashed lines correspond to \mathcal{B} pixels and \mathcal{B} Gaussian components, respectively, and points and continuous lines are \mathcal{O} pixels and Gaussians. (c) $p(\mathbf{x}|\mathcal{O})$ (d) $p(\mathbf{x}|\mathcal{B})$ (e) $p(\mathcal{O}|\mathbf{x})$. Brighter points correspond to more likely pixels.

2.3 Model Paramaterization

Once we have learnt the initial configuration of the *MoG* for \mathcal{O} and \mathcal{B} , we parameterize them with the following state vector:

$$\mathcal{X}_\varepsilon = [\mathbf{p}_\varepsilon, \mu_\varepsilon, \lambda_\varepsilon, \theta_\varepsilon] \quad (3)$$

where $\varepsilon = \{\mathcal{O}, \mathcal{B}\}$, and $\mathbf{p}_\varepsilon = [p_\varepsilon^{(1)}, \dots, p_\varepsilon^{(M_\varepsilon)}]$ contains the prior probabilities of each component, $\mu_\varepsilon = [\mu_\varepsilon^{(1)}, \dots, \mu_\varepsilon^{(M_\varepsilon)}]$ are the centroids of each gaussian, the eigenvalues of the principal directions are represented by $\lambda_\varepsilon = [\lambda_\varepsilon^{(1)}, \dots, \lambda_\varepsilon^{(M_\varepsilon)}]$ and $\theta_\varepsilon = [\theta_\varepsilon^{(1)}, \dots, \theta_\varepsilon^{(M_\varepsilon)}]$ are the angles between the principal axis of each component with the horizontal. Observe the interest of having a low number of gaussian components in order to reduce the dimensionality of this state vector. The algorithm described in section 2.1 works properly in the sense that allows us to select the lowest number of gaussian components that best represent the data.

3 Learning the Dynamical Model

One of the stages of the tracking algorithm, consists of propagating the state vector from Eq. 3, in order to generate multiple hypotheses about the future configuration of the *MoG*. In order to formulate these hypotheses we formalize a dynamic motion model in terms of an auto-regressive Markov process. We model color dynamics as a 2nd order process, represented by the expression:

$$\mathcal{X}_{\varepsilon,t} = A_0 \mathcal{X}_{\varepsilon,t-2} + A_1 \mathcal{X}_{\varepsilon,t-1} + D_0 + B_0 \mathbf{w}_t \quad (4)$$

where the matrices A_0, A_1 represent the deterministic component of the model, D_0 is a fixed offset, and $B_0 \mathbf{w}_t$ is the stochastic component, with \mathbf{w}_t a vector of standard normal random variables with unit standard deviation and $B_0 B_0^T$ is the process noise covariance. The parameters A_0, A_1, B_0 and D_0 are learned a priori using the MLE algorithm described in [1]. In order to generate the training data we use a hand-segmented sequence, where the initial *MoG* configuration is fitted using the unsupervised algorithm described in Section 2.1 and to fit the subsequent components we use the EM algorithm. In Fig. 4 we show the evolution of the training parameters for the \mathcal{O} and \mathcal{B} color distributions.

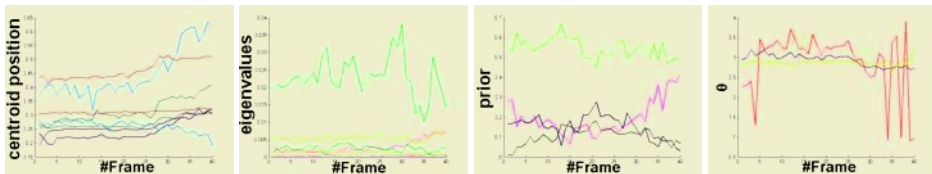


Fig. 4. Evolution of the Foreground parameters for the color distribution of a hand-segmented training sequence. There are shown the results for the 4 components of the *MoG*.

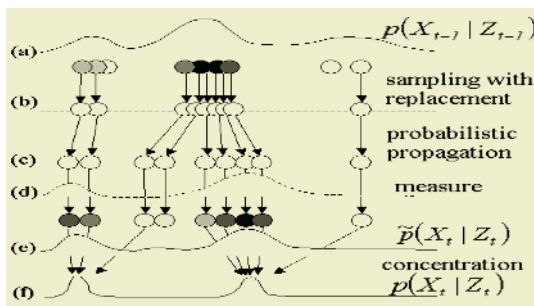


Fig. 5. One iteration of the implemented tracking algorithm for the one-dimensional case. The weight of each sample is represented by its gray level. The classical implementation of the CONDENSATION algorithm uses the steps (a)-(e). In our algorithm, we have added the ‘concentration’ step, where the samples are redirected to the local maxima.

4 The Tracking Algorithm

The basic steps of the tracking algorithm follow the typical procedure of the particle filters, but we introduce a modification similar to the idea presented in the algorithm ICONDENSATION [4], and in order to ‘direct’ the search for the next iteration we concentrate the future hypotheses on those areas of the state-space containing more information about the posterior probability $p(\mathbf{x}|\mathcal{O})$ (Fig. 5). Moreover, in this final stage we fuse object color and shape information. Next, we will briefly describe each one of the steps of the tracking algorithm:

1. **Probability Density Function of Color Distribution.** At time t , there are available from the previous iteration a set of N samples $\mathcal{S}_{t-1}^{(n)}$ ($n = 1, \dots, N$) with the same structure than \mathcal{X} (eq. 3), parameterizing N color distributions (Fig. 5a). Each sample has an associated weight $\pi_{t-1}^{(n)}$. The whole set represents an approximation of the a posteriori density function $p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})$ where $\mathcal{Z}_{t-1} = \{z_0, \dots, z_{t-1}\}$ is the history of the measurements. The goal of the algorithm consists of constructing a new sample set $\{\mathcal{S}_t^{(n)}, \pi_t^{(n)}\}$ for time t .
2. **Sampling from $p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})$.** The next step in the estimation of $p(\mathcal{X}_t|\mathcal{Z}_t)$ consists of sampling with replacement N times the set $\{\mathcal{S}_{t-1}^{(n)}\}$, where each element has probability $\pi_{t-1}^{(n)}$ of being chosen (step (b) from Fig. 5). This, will give us a set $\{\mathcal{S}'_t^{(n)}\}$ of *MoGs* parameterizations. Those samples having higher weights may be chosen several times, so the new set can have identical copies of elements. On the other hand, those distributions having lower weights may not be chosen.
3. **Probabilistic Propagation of the Samples.** Each sample $\mathcal{S}'_t^{(n)}$ is propagated according to the dynamic model learnt during the training stage (eq. 4):

$$\tilde{\mathcal{S}}_t^{(n)} = A_0 \mathcal{S}'_{t-2}^{(n)} + A_1 \mathcal{S}'_{t-1}^{(n)} + D_0 + B_0 \mathbf{w}_t \quad (5)$$

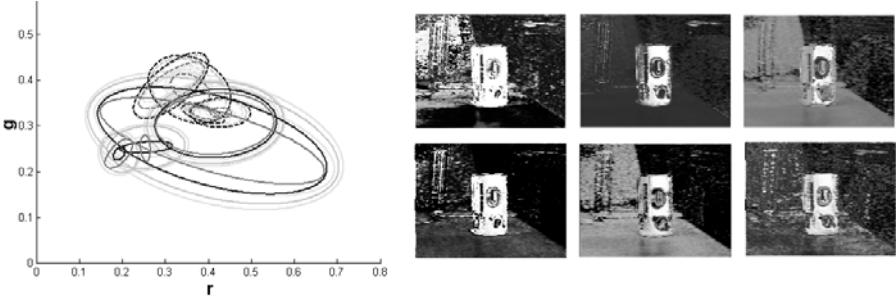


Fig. 6. Left: Ten samples of *MoG* from the set $\{\mathcal{S}_{t-1}^{(n)}\}$. The gray level is proportional to the weight of the samples. Right: Posteriori probability maps $p(\mathcal{O}|\mathbf{x})$, computed with different *MoGs*.

4. **Measure and Weight.** In this step, each element $\tilde{\mathcal{S}}_t^{(n)}$ has to be weighted according to some measured features. From the propagated samples $\tilde{\mathcal{S}}_t^{(n)}$ we construct the corresponding *MoG*, that are used to compute the probability maps $p(\mathcal{O}|\mathbf{x})$, for each sample (Fig. 6). The goal is to assign higher weights to the samples $\tilde{\mathcal{S}}_t^{(n)}$ generating ‘better’ segmentations of the tracked object. This is done assigning to each sample $\tilde{\mathcal{S}}_t^{(n)}$ the following weight:

$$\pi_t^{(n)} = \frac{\sum_{\mathbf{x} \in W} p(\mathcal{O}|\mathbf{x})}{N_w} - \frac{\sum_{\mathbf{x} \notin W} p(\mathcal{O}|\mathbf{x})}{\overline{N_w}} \quad (6)$$

where W is the interest region around the previous object position (where we predict that will be the object), and $N_w, \overline{N_w}$ are the number of image pixels in and out respectively, of this interest region.

5. **Sample Concentration.** In the last stage of the algorithm (Fig. 5f) we concentrate the samples around the local maxima, so that in the following iteration the hypotheses are formulated around these more likely regions of the state space. In our case, this is absolutely necessary because our state vector \mathcal{X} has high dimensionality (proportional to the number of gaussian components), 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 with the following steps:



Fig. 7. Steps to extract the exact position of the object fusing color segmentation and accurate adjustment by deformable contours (commented in the text).

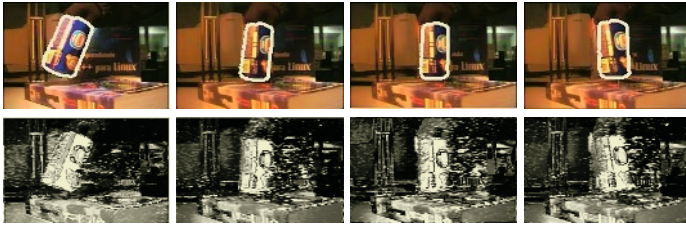


Fig. 8. Fitted contour and $p(\mathcal{O}|\mathbf{x})$ map on a sequence with gradual change of illuminant and object position.

- (a) The maximum from the set of weights $\{\pi_{t-1}^{(n)}\}$, $n = 1, \dots, N$ is taken, and using morphologic operations over its probability map image (Fig. 7b), a coarse approximation of the object shape is obtained.
- (b) With this rough shape, we eliminate noisy edges from the original image (Fig. 7c,d).
- (c) The contour of the object in the previous iteration, is used as initialization of a snake, that is adjusted to the previous edge image (Fig. 7e). The fusion of color segmentation and shape information increases the robustness of the system, because even when the color hypotheses give a highly rough estimation, they can be corrected using the contour information.
- (d) Once the object has been accurately detected (Fig. 7f), its color distribution is extracted. A *MoG* is fitted to this distribution (using the EM algorithm), giving a state vector \mathcal{S}_t^* . Samples $\tilde{\mathcal{S}}_t^{(n)}$ are ‘concentrated’ on this new distribution as follows:

$$\mathcal{S}_t^{(n)} = (1 - a)\tilde{\mathcal{S}}_t^{(n)} + a\mathcal{S}_t^* \quad (7)$$

where the parameter a governs the level of concentration. In our experiments we have set $a = 0.8$.

5 Results

In this section, three different experimental results are presented in order to illustrate the robustness and different capabilities of our system. In the first experiment (Fig. 8) we show how the method is able to face a gradual change of illumination and object position. The second experiment (Fig. 9) corresponds to a sequence with an abrupt change of both position and illumination. Fig. 9a and Fig. 9b correspond to the two consecutive frames presented to the algorithm. The a posteriori map of the best hypothesis (Fig. 9c) is used to discriminate false edges and fit a deformable contour (Fig. 9d,e). In this experiment we have constrained the fitting process to affine deformations. Finally, in the third experiment (Fig. 10) we show the performance of our system in a natural and cluttered scene, where we track the movement of an hippopotamus in the water. Observe that although the high level of noise and clutter from the scene the algorithm is able to perform a good tracking.

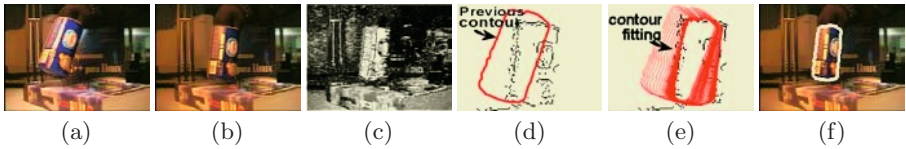


Fig. 9. Results of an abrupt change of illuminant and object position (commented in text).

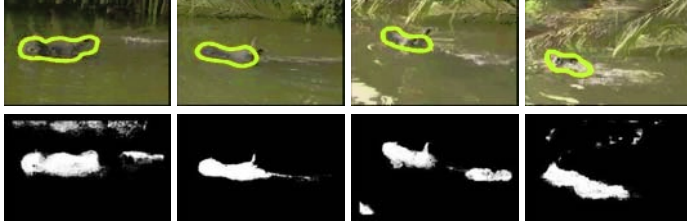


Fig. 10. Fitted contour and $p(\mathcal{O}|\mathbf{x})$ map of a natural sequence.

6 Conclusions

In this paper we have presented a new approach to the multi-colored object tracking under varying illumination environments that dynamically accommodates the color and shape of the object of interest. The main contribution of this work is the fusion of the shape and color information in the probabilistic framework offered by the particle filter formulation. We also introduce the concept of ‘concentration’ in the last stage of the CONDENSATION algorithm, what makes the system able to cope with a state vector of high dimensionality.

References

1. A.Blake, M.Isard, “Active contours”, *Springer*, 1998.
2. G.D.Finlayson, B.V.Funt, K.Barnard, “Color Constancy under Varying Illumination”, *Proc. ICCV*, pp.720-725, 1995.
3. M.A.T.Figueiredo, A.K.Jain, “Unsupervised Learning of Finite Mixture Models”, *Trans.PAMI*, Vol.24, num.3, pp.381-396, 2002.
4. M.Isard, A.Blake “Icondensaton: Unifying Low-Level and High-Level Tracking in a Stochastic Framework”, *Proc. ECCV*, Vol.1, pp.893-908, 1998.
5. F.Moreno-Noguer, J.Andrade-Cetto, A.Sanfeliu “Fusion of Color and Shape for Object Tracking under Varying Illumination”, *Proc.IBPRIA, LNCS 2652, Springer*, pp.580-588, 2003.
6. L.Sigal, S.Sclaroff, V.Athitsos, “Estimation and Prediction of Evolving Color Distributions for Skin Segmentation under Varying Illumination”, *Proc.CVPR*, Vol.2, pp.152-159, 2000.
7. Y.Raja, S.McKenna, S.Gong, “Colour Model Selection and Adaption in Dynamic Scenes”, *Proc.ECCV*, Vol.1, pp.460-475, 1998.
8. J.Yang, W.Lu, A.Waibel, “Skin-color modeling and adaption”, *Proc.ACCV*, Vol.2, pp.687-694, 1998.