NON-RIGID OBJECT LOCALIZATION FROM COLOR MODEL USING MEAN SHIFT

Gaël Jaffré *Alain Crouzil*

Institut de Recherche en Informatique de Toulouse
Université Paul Sabatier - 118 route de Narbonne - 31062 Toulouse Cedex 4 - France
{jaffre, crouzil}@irit.fr

ABSTRACT

This paper deals with non-rigid object localization in an image, from object colors. Our method allows detection in an image of all the objects which correspond to a color model, without a priori information about their number. Our approach consists in creating a binary image, which represents the repartition of the most probable pixels to be part of the object. Considering this image as a cluster in $\mathbb{R}^2$, the object localization is done by finding all the cluster modes. This search is carried out by applying a statistical method: the mean shift procedure. To illustrate our approach, we use sport images, from which we try to detect all the players.

1. INTRODUCTION

Our framework is sport image sequence analysis, especially player tracking. In this study, our topic search is non-rigid object localization in an image, to automatically initialize tracking procedures. This problem is often encountered in tracking applications where search is local (tracking with snakes, mean shift tracking, ...). The search being done in the neighborhood of the object position in the previous frame, initialization is needed for the first frame. Due to many difficulties, it is usually a manual initialization [1, 2].

Our problem presents two main difficulties : the non-rigid and the 3D aspects of the objects. In order to solve both difficulties, we make use of color densities of the objects. In this paper, we assume that objects have a discriminant color, i.e. their color is characteristic of them. Thus, color density is robust to object non-rigidity, partial occlusions, camera zooming and camera position changing.

Among publications about object detection from their color, we were inspired by [3] and [4]. In [3], Comaniciu proposes a face tracking application where, in the first frame, a face is iteratively detected from multiple initializations. In [4], Vandenbroucke proposes a different approach: from learning images, all the players are detected by pixel classification. However, this approach needs various learning data, and no results with noisy models are presented.

We use both method approaches. First, a classification of the pixels is done: from an object model, a binary image is created, where each pixel represents a belonging measure to the model (0 or 1). This image represents the repartition of the most probable pixels to be part of the searched object. An example is given in Fig. 1.b. This image will be used in the sequel as a model to illustrate the different examples.

Our approach consists in considering this binary image as a cluster in $\mathbb{R}^2$: the task of object localization reduces to the detection of local modes in the cluster. Each mode is associated with an object, which corresponds to the model (a mode is a local density maximum).

(a) Original image. (b) Desired cluster.

Fig. 1. Example of cluster we would like to obtain from white player model. Black points represent pixels with value 1, and white ones those with value 0.

In the sequel, we will no longer use binary images, but weighted binary images, i.e. the belonging measure to the model will not be exactly 0 or 1, but will be in the real interval [0,1].

This paper is structured in four parts. First, we detail the statistical tools used to estimate the cluster density and to search the cluster modes. Then, we discuss the used methods to create the cluster which represents the most probable pixel repartition. Besides, we show how we can extract the object coordinates from this cluster. In the last part, we present experiments of the application of our method on sport images.

2. STATISTICAL TOOLS

Our approach consists in considering binary images as clusters in $\mathbb{R}^2$. In this section, we will present the statistical methods used in this paper.

2.1. The Mean Shift Procedure

The mean shift is a nonparametric estimator of density gradient, proposed in 1975 by Fukunaga [5], in order to apply it for pattern recognition problems. However, it was really used only by Cheng [6] in 1995, then by Comaniciu and Meer [7] since 1997.

Let $\{\mathbf{x}_i\}_{i=1\ldots n}$ be an arbitrary set of n points in $\mathbb{R}^d$. The mean shift vector computed with kernel K and kernel bandwidth h is given by [7]

$$M_h(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}_i K\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right)}{\sum_{i=1}^{n} K\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right)} - \mathbf{x}$$

In this paper, we only deal with mean shift procedure in case of Epanechnikov kernel [5]. The mean shift vector definition becomes

$$M_h(\mathbf{x}) = \frac{1}{n_{\mathbf{x}}} \sum_{\mathbf{x}_i \in S_h(\mathbf{x})} \mathbf{x}_i - \mathbf{x}$$

(a) White player. (b) Red player. (c) Noisy white player.

Fig. 2. Player models used in the examples of this paper.

where $S_h(\mathbf{x})$ is the sphere centered on $\mathbf{x}$, of radius h and containing $n_{\mathbf{x}}$ data points. The mean shift vector has the direction of the gradient of the density estimate at $\mathbf{x}$. The mean shift procedure is obtained by successive computations of the mean shift vector $M_h(\mathbf{x})$, and translation of the sphere $S_h(\mathbf{x})$ by $M_h(\mathbf{x})$. The procedure is guaranteed to converge [7].

2.2. Mode Research

In [5], Fukunaga proposes an algorithm of mode seeking using the mean shift procedure. He applies the procedure to each point: when two data points converge to the same final position, they are considered to belong to the same cluster.

However, complexity is $O(n^2)$, where n is the number of data points, what provides huge computational time. In order to be usable to quickly initialize tracking procedure, complexity of mode seeking must be reduced. First, we use images as clusters, so data are in a regular grid, providing an efficient computation of the mean shift procedure [7]. To even reduce complexity, we use the approach presented in [8]: instead of applying the mean shift procedure to every point, we only apply it to a subset. This is carried out as described below.

(1) Define a tessellation of the cluster with m spheres of radius h:

- the distance between two neighbors should not be smaller than h,
- (optional) the number of points inside the sphere should not be below a threshold T_1.

(2) Apply the mean shift procedure to the sphere centers. The different points of convergence are the modes of the cluster.

(3) Merge modes whose distance is less than h.

(4) Discard modes whose density estimate is below a fraction T_2 of maximal density estimate.

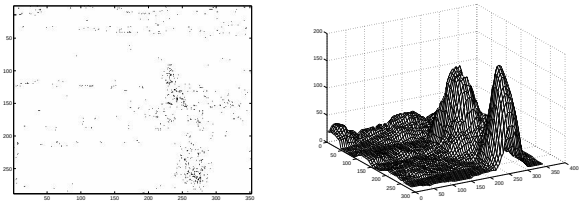
3. CLUSTER CREATION FROM IMAGES

The color models we used in this paper are presented in Fig. 2.

3.1. Binary Images

The most naive approach consists in creating binary image with the following way: each pixel whose color appears in the model has the value 1, and each pixel whose color does not appear has the value 0. Example is given in Fig. 3, where result seems satisfactory: density estimate of the cluster presents two maxima which can be distinguished, and which correspond to the "centers" of the two players. However, this approach is very limited, because on real data the color model is often slightly different from the color of the object in the image. Moreover, results are very sensitive to noise in the model, as we will see in section 3.3.

To deal with slight changes in color, histograms can be used. To compare two pixels, we must now compare their index in the



(a) Cluster. (b) Density estimate.

Fig. 3. Cluster from the naive approach, with the model of Fig. 2.a.

histogram instead of directly comparing their color or gray level. The problem consists in finding the optimal bin number. A small one will provide too many points in the cluster, whereas a too large one will not allow important changes in color.

Vandenbroucke proposes another approach to get binary images from models, with soccer images [4]. Learning data are drawn from various player models, for which many color representation systems are used. Finally, he only keeps the system which provides the best separation between the two classes of pixels according to the player soccer suits. Obtained clusters have a very good quality (like the desired models shown in Fig. 1.b). However, this method needs many learning models, many color system changes, and particularly noise-free models (noise is manually removed).

3.2. Weighted Binary Images

Our clusters are obtained from binary images. Each pixel has only two possibilities: either it belongs to the model, or it does not. The weighted binary image allows more degrees in model belonging. Largest values correspond to most probable pixels, whereas lower ones correspond to unlikely pixels. Cluster is now obtained by taking only the pixels with strict positive value, each value corresponding to a weight: thus, most probable pixels will have a greater weight than unlikely pixels.

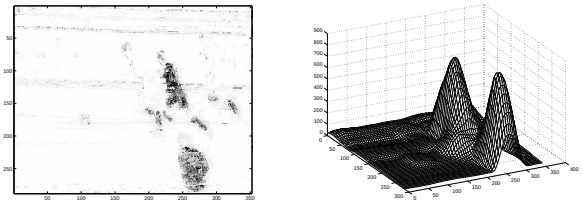
Our method to create cluster is the one proposed by Swain and Ballard in [9]: the measure of the object presence in image is defined from what is called the backprojected image.
Let us denote by

- $\{\mathbf{x}_i\}_{i=1\ldots n}$ the location of the n pixels in the image,
- $\{q_k\}_{k=1\ldots m}$ the m-bin color histogram (linearized color histogram with m bins) of the image,
- $\{p_k\}_{k=1\ldots m}$ the m-bin color histogram of the model.

We also define the function $c : \mathbb{R}^2 \rightarrow \{1 \ldots m\}$ which associates to the pixel at location $\mathbf{x}_i$ the index $c(\mathbf{x}_i)$ of the histogram bin corresponding to the color of that pixel. The ratio histogram $\{r_k\}_{k=1\ldots m}$ is defined as $r_k = \min\left(\frac{p_k}{q_k}, 1\right)$. It associates to each color a belonging measure to the model. It will allow the object presence in the image to be measured. Its backprojection onto the image associates the value $r_{c(\mathbf{x}_i)}$ to the pixel $\mathbf{x}_i$, for all $i = 1 \ldots n$. Since the ratio histogram emphasizes the predominant colors of the target while diminishing the presence of clutter and background colors, the backprojected image $\left\{r_{c(\mathbf{x}_i)}\right\}_{i=1\ldots n}$ represents a spatial measure of the object presence.

In order to compute the mean shift vector for all $\mathbf{x} \in \mathbb{R}^d$, we use a definition which takes the weights into account

$$M_h(\mathbf{x}) = \frac{\sum_{\mathbf{x}_i \in S_h(\mathbf{x})} w(\mathbf{x}_i)\, \mathbf{x}_i}{\sum_{\mathbf{x}_i \in S_h(\mathbf{x})} w(\mathbf{x}_i)} - \mathbf{x}$$



(a) Cluster. (b) Density estimate.

Fig. 4. Backprojected image.

where $S_h(\mathbf{x})$ is the sphere of radius h and center $\mathbf{x}$, and $w(\mathbf{x}_i)$ is the weight associated to the point $\mathbf{x}_i$.

Backprojected image obtained from our soccer image is given in Fig. 4. We computed the histogram in the RGB space with 16 bins for each color.

3.3. Comparison with Noisy Models

With synthetic models, we have similar results with naive approach and backprojected image. However, this difference becomes obvious as models become noisy. In order to have a synthetic model from an image of the object, we must select the image area where the object is, then manually remove the background [4]. This model could be used for various images.

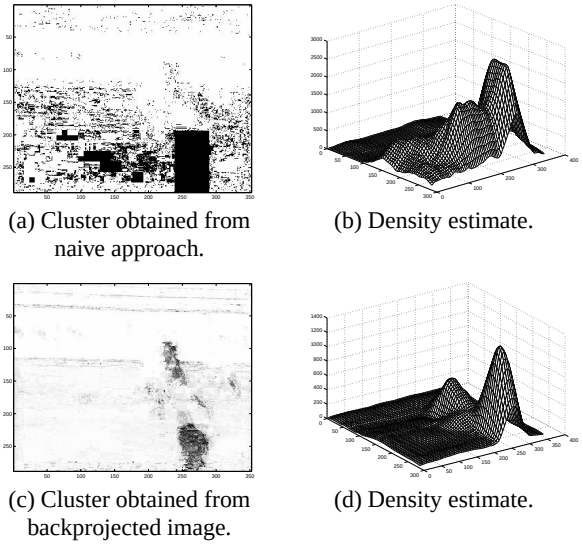
However, we could expect to have a model from an image area, without manual correction. Background pixels will then be part of the model. The methods previously presented have different behaviors in presence of noisy pixels, as we can observe in Fig. 5: with the naive approach, background pixel presence provides a too noisy cluster, which makes player localization impossible. One can see that in the rectangular area where the model was extracted, every pixel was kept for the cluster. On the other hand, with backprojected image, the ratio $\frac{p_k}{q_k}$ allows the background pixel influence to decrease [9], because in the image, the background pixels are more numerous than the object pixels. In the area where the model is extracted, background pixels have low weights, whereas player pixels keep large weights. Density estimate of the backprojected image has two maxima which can be distinguished, the most important being obviously the one which corresponds to the player selected for the model. This last method providing more reliable results on real (noisy) data, we will use it in our applications.

4. OBJECT LOCALIZATION

When the cluster is computed, object localization is carried out by applying the algorithm described in section 2.2, which allows the detection of all the cluster modes, without knowledge about their number. The only a priori information needed is the scale of the cluster, i.e. the radius h we will use in the mean shift procedure. When mode computation is ended, each mode corresponds to an object center.

When there is only one object to find, its position corresponds to the global maximum of cluster density estimate. Even if pixel by pixel density computation is possible, the algorithm of section 2.2 is more efficient in term of time computation because density is only estimated for a few number of points. Moreover, the last step of the algorithm becomes useless, because we only keep the mode with maximum density. Thus, threshold T_2 is not used.

This approach is inspired from [3], where face localization is carried out from various initializations. For each of them, the mean shift procedure is run, in order to compute the corresponding



(a) Cluster obtained from naive approach. (b) Density estimate.

(c) Cluster obtained from backprojected image. (d) Density estimate.

Fig. 5. Noise influence on clusters. Models were drawn from an image area, without background removal. (Fig. 2.c).

mode, and to keep the one with the highest peak. However, we can point out some differences with our approach. First, initialization points are not randomly taken in sufficiently populated region, but are always chosen in the same location, for all images. Besides, the optimization is not the one of density estimate of the backprojected image, but the one of Bhattacharyya coefficient, which measures the similarity between color distributions of the model and an image area. The advantage of using backprojected image instead of Bhattacharyya coefficient is the computational time, which is lower with backprojected image.

When we know the number N of the objects in the image, the threshold T_2 is useless. The method is the same than with only one object, but we keep the N modes with highest density estimate.

When we do not know the number of objects in the image, which usually occurs in applications, we then apply all the steps of the algorithm of section 2.2. With the last step, the modes with a too small density estimate will be removed. In our experiments, we used $T_2 = 25\ \%$. This threshold is not very dependent on the current environment, i.e. we did not have to change it in our different experiments.

5. EXPERIMENTS

We applied this method with different sport images, and results[1] are satisfactory. When we deal with collective sports with two teams, we must run the procedure two times (one by team model).

Fig. 6 and 7 present results of our method with different sport images. Sportsman locations are given by the circle centers. In both cases, we did not know the number of players in the images.

The limits of our method arise when conditions become difficult, as in Fig. 8, where there is a billboard with the same color than white players, and those ones are only represented by few pixels. All the players are detected, but there are two false alarms, the billboard and the referee.

[1]Other results are available on the web site http://www.irit.fr/∼Gael.Jaffre/RECHERCHE/dea.html

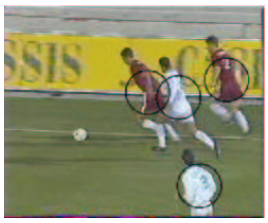


Fig. 6. Soccer player localization.

(a) Noisy surfer model.

(b) Result.

Fig. 7. Surfer localization from a noisy model.

6. CONCLUSIONS AND DIRECTIONS OF FURTHER RESEARCH

Our method allows non-rigid object localization from a color model, even in the presence of noise, when they have enough pixels. This method can be used to automatically initialize tracking procedures, which are often manually initialized in the first frame [1, 2].

The interest of our approach is the use of the mean shift procedure, which is a nonparametric statistical procedure, robust to noise always present in real data. Mean shift procedure allows quick convergence toward object locations, due to low complexity, low number of iterations and local search.

Another advantage in our method is the low number of parameters: object color model, approximate scale of the objects, and thresholds T_1 and T_2. The threshold T_1 is only used to reduce the number of mean shift procedure uses, it can be removed if we do not want to determine it. The threshold T_2 is only used when we do not know the number of objects in the image. We set it experimentally, and as it is not very dependent on the current environment, we did not change it in all our examples.

So, the user only has to give a color model of the object, and its approximate size. To avoid manually creating a synthetic model, the user can straightforwardly give the center object location: the model will be extracted from neighborhood (which depends on the scale) of the object. Finally, the user may not choose any scale, but select an area in the image which corresponds to an object: the color model will be initialized from the colors of this area, and the scale will be taken from the size of the area. In this case, we can take two different scales h_x and h_y for the height and the width of the object. Background pixels will only have a weak influence, as discussed in section 3.3.

However, a prior knowledge about the scale remains needed. We would like to remove this a priori knowledge, by automatically computing the scale. Thus, we will use the works presented in [10] and [11], where are described clustering methods which use the mean shift procedure, but automatically compute the scale from data.

7. ACKNOWLEDGMENT

This work has been conducted on the behalf of the KLIMT project.

Fig. 8. Soccer player localization : difficult case.

8. REFERENCES

[1] Chris Needham and Roger Boyle, "Tracking multiple sports players through occlusion, congestion and scale," in *Proceedings of the 12th British Machine Vision Conference*, Manchester, UK, Sept. 2001, vol. 1, pp. 93–102.

[2] Janez Perš and Stanislav Kovačič, "Tracking People in Sport: Making Use of Partially Controlled Environment," in *Proceedings of the 9th International Conference on Computer Analysis of Images and Patterns*, Warsaw, Poland, Sept. 2001, pp. 374–382.

[3] Dorin Comaniciu and Visvanathan Ramesh, "Robust Detection and Tracking of Human Faces with an Active Camera," in *Proceedings of the 3rd IEEE International Workshop on Visual Surveillance*, Dublin, Ireland, July 2000, pp. 11–18.

[4] Nicolas Vandenbroucke, Ludovic Macaire, and Jack-Gérard Postaire, "Color image segmentation by supervised pixel classification in a color texture feature space. Application to soccer image segmentation," in *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, Spain, Sept. 2000, vol. 3, pp. 625–628.

[5] Keinosuke Fukunaga and Larry Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 32–40, Jan. 1975.

[6] Yizong Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[7] Dorin Comaniciu and Peter Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002.

[8] Dorin Comaniciu and Peter Meer, "Distribution Free Decomposition of Multivariate Data," *Pattern Analysis and Applications*, vol. 2, no. 1, pp. 22–30, 1999.

[9] Michael Swain and Dana Ballard, "Color Indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, Nov. 1991.

[10] Dorin Comaniciu, "An Algorithm for Data-Driven Bandwidth Selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 281–288, Feb. 2003.

[11] Hong Pan, Stan Li, and Guodong Guo, "Robust Unsupervised Clustering Using Generalized Annealing M-Estimator," in *Proceedings of the 4th Asian Conference on Computer Vision*, Taipei, Taiwan, Jan. 2000, vol. 1, pp. 104–109.