Object Detection in Video via Particle Filters

Jacek Czyz
Communications Laboratory, Université catholique de Louvain
1348 Louvain-la-Neuve, Belgium
czyz@tele.ucl.ac.be

Abstract

We propose an object detection method using particle filters. Our approach estimates the probability of object presence in the current image given the history of observations up to current time. To do so, object presence is modelled by a two-state Markov chain, and the problem is translated into sequential Bayesian estimation which can be solved by particle filters. The observation density, required by the particle filter is based on selected discriminative Haar-like features that were introduced by Viola and Jones [7] for object detection in static images. We illustrate the approach on the problem of face detection. Experiments on real video sequences show the feasability of the approach.

1. Introduction

Object detection in images has received considerable attention in the past decades, probably because reliable object detection systems are required as a front-end in numerous applications. For example, face detection is the first stage of many human computer interaction systems. Object detection deals with determining if an instance of a given class of objects (for examples cars, faces, etc.) is present or not in an image. Successful object detection systems are based on the learning of object appearance using large collections of examplars.

Many systems take only into account the information contained in one image to detect the object. This paper uses the particle filter framework for detecting objects in video. We consider that object presence is a discrete random variable that can be modelled as a two-state Markov chain. To perform object detection, we propose to compute the probability of the object to be visible in the current frame, given the history of all frames up to the current time step. In order to estimate this probability we formulate the problem as recursive Bayesian estimation and solve it using sequential Monte Carlo or particle filter techniques [1].

The approach is illustrated in the context of face detection. Although in principle any static face detection algorithm could be used to create the observation density required by the particle filter (PF), we illustrate the detection using the Viola-Jones face detector [7]. This face detection method is based on the learning of a small discriminative subset from a large set of Haar-like rectangular features. These features can be efficiently evaluated by a few arithmetic operations on the so-called integral image, which makes it very suitable for computationally intensive methods like particle filters. Combination of rectangular features and PF's has been done before, yet with other purposes. In [3], the output of the Viola-Jones detector is used to construct a relevant importance density in the context of tracking multiple targets. Micilotta and Bowden [2] and Yang et al. [9] use the rectangular features in the PF observation density but they focus on pure tracking, detection being obtained by other means. Wang et al. [8] focus on online selection of discriminative rectangular features for robust tracking in clutter.

Our work is related, in spirit, to the propagation of detection probability proposed in [6]. In their work, the authors create a face probability map over each frame using a static face detector resembling the one of Schneiderman and Kanade [5]. Samples that hypothetise face presence are initialised around local maxima of the map and propagated to next frames using a prediction and update model. The update model is based on the probability maps computed for each frame. Face appearence and disappearance are treated by devoted procedures on the samples. In contrast, our approach is defined strictly in the Bayesian framework. Face appearance and disappearence are therefore managed naturally through the object presence random variable. The resulting method is simpler and has less parameters than in [6].

The paper is organised as follows. In the next section, we formulate the problem of detection in video and we present the models adopted in this paper. Section 3 presents the particle filter solution to the detection problem. Section 4 is devoted to experiment and Section 5 concludes the paper.

2. Problem formulation

Let us model the event "object is present" and "object is not present" by a discrete random variable E with $E = 0$ when the object is not visible and $E = 1$ when the object is visible, Classically the detection is done in a given signal $\mathbf{z}_k$ at the time step k by comparing the probability that the object is present given the input signal, i.e. $P(E = 1|\mathbf{z}_k)$ with the probability that the object is absent $P(E = 0|\mathbf{z}_k)$. This is equivalent to comparing the likelihood ratio

$$l(\mathbf{z}_k) = \frac{p(\mathbf{z}_k|E = 1)}{p(\mathbf{z}_k|E = 0)}$$

to a threshold. Most of object detection methods proposed in the literature use pattern recognition and machine learning techniques to learn a discriminative function which approximates the likelihood ratio. The learning is done on training data with both positive and negative example images. In order to actually detect the face, image sub-windows that potientially contain the face are first extracted and the discriminative function is applied to these sub-windows. Motion or color cues can be used to guide the sub-window extraction and concentrate the attention of the detector on image regions with high probability of face presence. Finally the discriminative function or *score* $s_k(\mathbf{z}_k)$ is simply compared to a threshold, and it is decided that the face is present if $s_k > t$ and not present if $s_k < t$.

This approach takes only into account the information contained in the current frame for making the decision. It overlooks the fact that the frames are contiguous in the sequence as noted in [6]. It is likely that performance would be improved by using information from several frames. To do so, we can model the presence of the object using a Markov chain with two values: $E = 0$ and $E = 1$. The position and size of the object can be included in a unknown random vector $\mathbf{x}_k$. The probability of object presence given frame $\mathbf{z}_k$ is the marginal of the joint probability of object presence and the object position and size given the observation, i.e.

$$P(E_k = 1|\mathbf{z}_{1:k}) = \int p(E_k = 1, \mathbf{x}_k|\mathbf{z}_{1:k}) d\mathbf{x}_k. \qquad (1)$$

Bayesian sequential estimation allows to find $p(E_k, \mathbf{x}_k|\mathbf{z}_{1:k})$ recursively.

In order to solve the sequential estimation problem we must specify a motion model, i.e. the evolution of the state through the transition probability density function (pdf) $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, and a measurement likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$, i.e. the link between the state and the current measurement. The next three subsections describe the modelling of object motion, of object appearance and disappearance, and the measurement likelihood function.

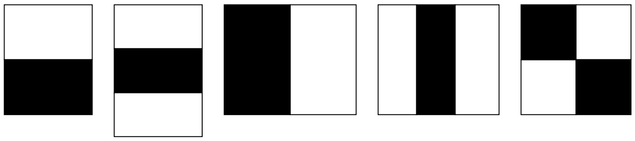


Figure 1. The five types of rectangular features.

2.1. State vector and dynamic model

The state vector $\mathbf{x_k}$ of an object at frame k typically consists of parameters of the image sub-window that potentially contains the face.

For simplicity we adopt the random walk model $\mathbf{x}_k = [x_k\ y_k\ S_k]^T$, where (x_k, y_k) denotes the upper left corner of the rectangular image region (in our case a square) used for extracting the sub-window, S_k denotes the square size. Note that other variables can be added, such as velocities and scale change rate, depending on the application. The state dynamics is described by the linear model: $\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{w}_{k-1}$, where $\mathbf{w}_{k-1}$ is process noise, assumed to be white, zero-mean Gaussian, with covariance matrix $\mathbf{Q}$. The transition pdf is therefore

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_{k-1}, \mathbf{Q}).$$

2.2. Object presence

The presence random variable $E \in \{0, 1\}$ is modelled by an 2-state Markov chain, whose transitions are specified by a 2×2 transitional probability matrix (TPM)

$$\mathbf{\Pi} = \begin{bmatrix} 1 - P_a & P_a \\ P_d & 1 - P_d \end{bmatrix},$$

where

$$P_a = Pr\{E_k = 0|E_{k-1} = 1\} \qquad (2)$$

$$P_d = Pr\{E_k = 1|E_{k-1} = 0\} \qquad (3)$$

are the probabilities of object appearance and disappearence respectively.

2.3. Observation model

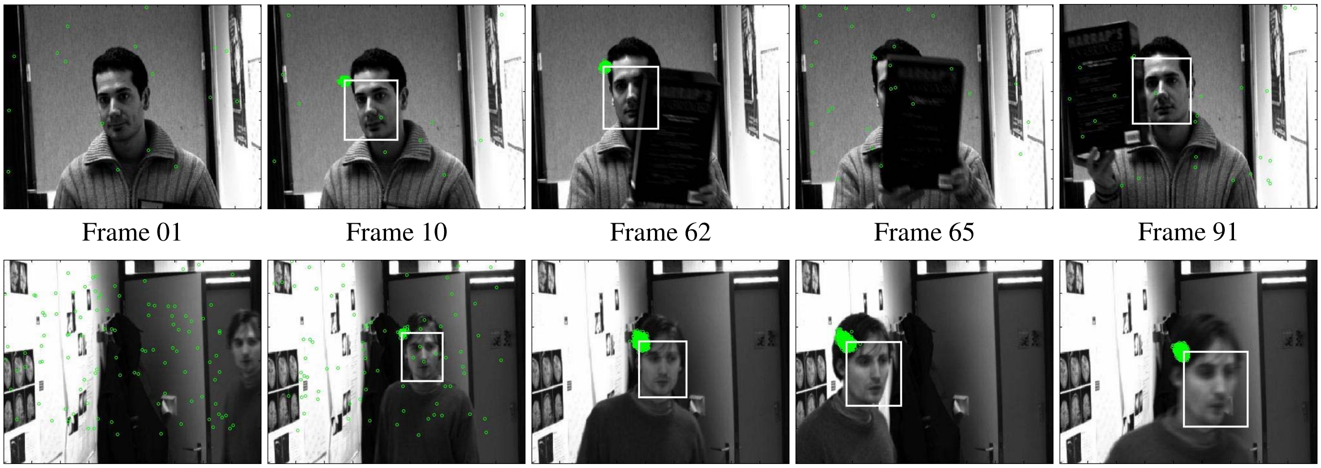
Like in [9, 2], we adopt an observation density based on the rectangle features introduced by Viola and Jones [7]. However, in [9, 2] the features are used to track the object through the sequence, object detection being obtained by other means.

The observation density or measurement likelihood function relates the currently observed image to a model of the object. The proposed density is based on the Viola-Jones static face detector [7]. The detector discriminates between face and non-face images by computing the response of a small subset of discriminative rectangular features. Five types of features, depicted on Figure 2, are used. The response $f_i(I)$ of a feature f_i on an image I is the difference between the sum of pixel values in the white area and sum of the pixel values in the black area. The subset of discriminative features is obtained from a large set of rectangular features by an adaBoost ensemble learning method. Each feature f_i comes with a threshold t_i and a coefficient α_i (which reflects the discriminative power of the feature) obtained during the learning phase. The classification in the face or non-face class is based on the following score

$$s(I) = \sum_{i=1}^{n_f} \alpha_i u(f_i(I) - t_i), \qquad (4)$$

where n_f is the number of selected features and $u(.)$ is the unit step function. Note that the selected rectangular features must be adapted in scale if the sub-window size is different from the original feature size used in the feature selection step (which was chosen 24x24). Instead of making a hard decision by comparing s to a threshold as in the static case, the probabilistic formulation allows to take into account the “soft” likelihood of face presence. The score s can be interpreted as a measure of similarity between the observed image I and a face model implicitely learned through the adaBoost feature selection process. We therefore adopt the following measurement likelihood function

$$p(\mathbf{z}_k|\mathbf{x}_k) \propto \exp\left(\beta s_k(I) + \gamma\right) \qquad (5)$$



Frame 01 Frame 10 Frame 62 Frame 65 Frame 91

Frame 15 Frame 51 Frame 93 Frame 109 Frame 151

Figure 2. Face detection results. Small circles show the x and y component of particles (i.e. the upper left corner of the region) with $E_k^n = 1$. Once $P > 0.5$ the face is decided to be present and a white box is drawed using the estimated state vector.

where I is the sub-image extracted from $\mathbf{z}_k$ at the location specified in the state vector $\mathbf{x}_k$, and β and γ are design parameters that are set using training sequences. With this definition, the likelihood function takes large values on image regions that have a face appearance and small values on image regions that do not have a face appearance. For the results reported below, we selected $n_f = 200$ features using a training set of 4000 faces images and 7000 of non-faces images.

3. Particle filter for detection

In the following we briefly outline the conceptual solution to detection in image sequences in the Bayesian framework, for the models described in the previous section. We then present the particle filter that implements this solution.

The main idea is to estimate object presence and object location and size at the same time. To do so, we first define an augmented state vector

$$\mathbf{y}_k = \begin{cases} E_k & \text{if } E_k = 0, \\ (E_k, \mathbf{x}_k^T)^T & \text{if } E_k = 1. \end{cases} \tag{6}$$

which includes the state vector with the object caracteristics and the presence discrete variable. Given the posterior density $p(\mathbf{y}_{k-1}|\mathbf{z}_{1:k-1})$, and the latest available image $\mathbf{z}_k$ in the video sequence, the goal is to construct the posterior density at time k, $p(\mathbf{y}_k|\mathbf{z}_{1:k})$. Once the posterior pdf $p(\mathbf{y}_k|\mathbf{z}_{1:k})$ is known, the probability P that the object is present in a video sequence at time k is computed using Equation (1).

Note that the object position and size in the frame k can also be computed as the marginal of the pdf $p(\mathbf{y}_k|\mathbf{z}_{1:k})$. These types of problems are referred to as hybrid state estimation because the state vector to be estimated involves both continuous (the subwindow parameters) and discrete-valued variables (the E variable) [4]. The formal Bayesian recursive solution to the hybrid state estimation can be presented as a two step procedure consisting of prediction and update, and can be found in [4] (Chap. 11). This solution can be implemented by a particle filter, which is described in the following.

Particle filters approximate the posterior density $p(\mathbf{y}_k|\mathbf{z}_{1:k})$ by a weighted set of random samples or particles. In our case, a particle of index n is characterized by a certain value of E_k^n variable and, if $E_k^n = 1$, a state vector $\mathbf{x}_k^n$ i.e. $\mathbf{y}_k^n = [E_k^n, \mathbf{x}_k^n]$ for $n = 1, \ldots, N$ where N is the number of particles. The five main steps of this filter are detailed below.

The first step is to simulate the random transitions of E_{k-1}^n to E_k^n based on the TPM $\mathbf{\Pi}$. Step 2 is the prediction step. Given the transitions from the previous step, we distinguish four cases:

- If $E_{k-1}^n = E_k^n = 1$, then we draw $\mathbf{x}_k^n$ from the transition density $p(\mathbf{x}_k|\mathbf{x}_{k-1}^n)$ as in a traditional bootstrap filter.
- If $E_{k-1}^n = 0$ and $E_k^n = 1$, the particle n supports the hypothesis that a face becomes visible at frame k. In this case, the state vector $\mathbf{x}_k^n$ is drawn from a pdf $p_b(\mathbf{x}_k)$ which is assumed to be known. This pdf encodes the prior knowledge of face caracteristics (size and position) when the face becomes visible in the image. For example, $p_b(\mathbf{x}_k)$ can be chosen to have large values in regions where faces are likely to appear: entrances, image borders , etc. or in regions determined using other cues such as skin color or motion. If this knowledge is imprecise, $p_b(\mathbf{x}_k)$ can be modeled as a uniform density.
- If $E_{k-1}^n = 1$ and $E_k^n = 0$, the particle n supports the hypothesis that the face is no more visible (either because it is occluded or it has left the scene). In this case the state vector $\mathbf{x}_k^n$ does not exist, since a particle with $E_k^n = 0$ does not have a state vector attached to it according to Equation (6).
- Nothing has to be done for particles with $E_{k-1}^n = 0$ and $E_k^n = 0$.

Step 3 is the update step. Using (5) the unnormalized importance weights are computed as:

$$\tilde{w}_k^n = \begin{cases} 1, & \text{if } E_k^n = 0 \\ C\exp\left(\beta s_k^n + \gamma\right), & \text{if } E_k^n = 1 \end{cases} \tag{7}$$

where C, β, γ are design parameters and s_k^n is the detection score computed from image $\mathbf{z}_k$ in the region specified by $\mathbf{x}_k^n$.

Step 4 and 5 are the standard normalising and resampling procedures respectively. Finally, the presence of the object is estimated based on (1), where the probability of object presence $P(E = 1|\mathbf{z}_{1:k})$ is computed in the PF as

$$P = \frac{1}{N}\sum_{n=1}^{N}\delta(E_k^n, 1),$$

where $\delta(i, j)$ is the Kronecker delta. The estimate of the state vector of the object is then

$$\hat{\mathbf{x}}_{k|k} = \frac{1}{N_i}\sum_{n=1}^{N}\mathbf{x}_{i,k}^n\ \delta(E_k^n, 1), \tag{8}$$

where $N_i = \sum_{n=1}^{N}\delta(E_k^n, 1)$.

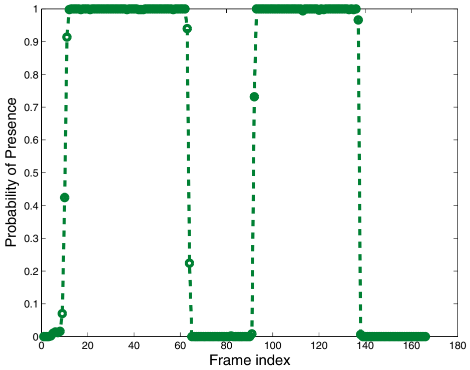


Figure 3. Probability of face presence P estimated by the PF for the sequence of Figure 2 (1st row).

4. Experiments

In this section we present face detection results on two monochromatic sequences, using the proposed method. The image size is 640x480 for both sequences. The transition probabilities P_a and P_d are set to 0.05. For the observation density we used the following parameters: $\beta = 1.25$, $C = 40$ and $\gamma = 2.5$. The density $p_b(\mathbf{x}_{i,k})$ is modeled as a uniform density over the state vector variables which is equivalent to no prior knowledge on where the object is likely to appear. The PF is initialised with all particles in state $E^n = 0$. The first row of Figure 2 shows the results of the detection on the first sequence. 500 particles were used in this example. It can be seen that the subject's face is visible at frame 1 and is detected by the filter at frame 10. At frame 62, the subject's face starts to be occluded by a dark box. The filter decides that the face has disappeared at frame 65. When the face is again visible, it is detected quickly. See Figure 3 for the estimated probability P vs. time index k for this sequence. Note that the camera is not stationary in this sequence.

The second row of Figure 2 shows a more challenging detection example: the face is small, moving and illumination is poor. For that case, 2000 particles were necessary. The subject's face starts to be visible at frame 15 and is detected at frame 51. It is then correctly detected until the end of the sequence despite the large scale variations and the fact that the face is not totally frontal.

5. Conclusion

We have presented an algorithm based on PF's for detecting objects in video. The observation density required by the PF is based on a set of rectangular features selected by an adaboost procedure as introduced in [7]. The approach allows to estimate the probability of face presence in the current frame given the history of all observed frames. This allows to accumulate the likelihood of object presence over several frames. The resulting system is therefore less sensitive to false detections, while being able to detect faces in difficult cases. We are now inverstigating ways to improve the algorithm efficiency by adapting the "cascade detection" [7] to our video-based detection.

References

[1] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[2] A. Micilotta and R. Bowden. View-based location and tracking of body parts for visual interaction. In *British Machine Vision Conference*, pages 849–858, 2004.

[3] K. Okuma, A. Taleghani, N. D. Freitas, J. Little, and D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *Proc. European Conf. Computer Vision*, pages 28–39, 2004.

[4] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, 2004.

[5] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *Int. Conf. on Computer Vision and Pattern Recognition*, pages 1746–1759, 2000.

[6] R. C. Verma, C. Schmid, and K. Mikolajczyk. Face detection and tracking in a video by propagating detection probabilities. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(10):1215–1228, 2003.

[7] P. Viola and M. Jones. Robust real-time object detection. *International Journal on Computer Vision*, 57(2):137–154, 2004.

[8] J. Wang, X. Chen, and W. Gao. Online selecting discriminative tracking features using particle filter. In *Int. Conf. on Computer Vision and Pattern Recognition*, 2005.

[9] C. Yang, R. Duraiswami, and L. S. Davis. Fast multiple object tracking via a hierarchical particle filter. In *Int. Conf. on Computer Vision*, pages 212–219, 2005.