Integrating RJMCMC and Kalman filters for multiple object tracking

Paula Craciun, Mathias Ortner, Josiane Zerubia

Integrating RJMCMC and Kalman filters for multiple object tracking

Paula CRĂCIUN[1] [*], Mathias ORTNER[2], Josiane ZERUBIA[1]

[1]INRIA, Ayin team
2004 Route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex, France

[2]Airbus Defence and Space
31 rue des Cosmonautes, 21402 Toulouse Cedex 4, France

Paula.Craciun@inria.fr, Mathias.Ortner@astrium.eads.net
Josiane.Zerubia@inria.fr

Résumé – Dans cet article, nous proposons d'intégrer le filtre de Kalman à un échantillonneur de Monte Carlo par chaîne de Markov à sauts réversibles (RJMCMC) pour améliorer la procédure d'optimisation dans le cas de suivi d'objets multiples. Nous proposons l'utilisation d'un noyau de perturbation dédié qui utilise le filtre de Kalman pour générer de multiples objets dans une seule itération. Nous démontrons que ce noyau permet de réduire considérablement le temps de mélange de la chaîne de Markov, par rapport à l'échantillonneur RJMCMC standard. Nous montrons les résultats obtenus sur deux séquences biologiques synthétiques et deux séquences simulées de télédétection sur la ville de Toulon, France.

Abstract – In this paper, we propose to integrate the Kalman filter with the reversible jump Markov Chain Monte Carlo (RJMCMC) sampler to improve the optimization procedure in the case of multiple object tracking. We propose the use of a dedicated perturbation kernel that uses the Kalman filter to generate multiple objects in a single iteration. We demonstrate that this kernel reduces considerably the mixing time of the Markov chain, as compared to the standard RJMCMC sampler. We show results on two synthetic biological sequences and two simulated remotely sensed data sets of the city of Toulon, France.

1 Introduction

Multiple object tracking is required in many vision applications such as traffic control, security and surveillance systems and is increasingly used in biological settings due to recent developments in microscopy techniques. Object tracking in video sequences requires the processing of a large amount of data and is generally time-expensive. The applications can be broadly divided into two groups : classical applications where the objects exhibit an independent behavior w.r.t. other objects (i.e. boats can be considered independent from cars on a road) and applications in which objects interact with each other (i.e. natural systems, road traffic, etc.). Moreover, the multiple object tracking problem becomes even more challenging when the objects are identical. Different techniques for solving the tracking problems in vision are available in literature. Luo et al. [7] offers a comprehensive review.

Recently, we have proposed a new spatio-temporal marked point process model for tracking small, rigid objects in high resolution images [2]. We show very good detection and tracking results for synthetic biological data as well as remotely sensed sequences. The model is based on defining a dedicated energy function that is highly non-convex. The solution is found by minimizing this energy function using a suitable batch-optimization scheme based on RJMCMC sampler. This approach is motivated by the low temporal frequency of the sequences ($\leq$ 1Hz).

In this paper we propose to combine the Markov chain dynamics with a sequential filtering technique, i.e. the Kalman filter, to obtain a hybrid optimization procedure that integrates the strengths of both approaches. More precisely, we build a new perturbation kernel that uses the Kalman filter to generate a sequence of objects in a single interation. We apply this optimization procedure to minimize the energy of the model described in [2]. We show results on biological and remotely sensed image sequences and compare our sampler with the sampler previously proposed in [2].

The paper is organized as follows : section 2 briefly presents the model used for multiple object tracking. We present the sampler in section 3 and show results in section 4. Finally, conclusions are drawn in section 5 and future work is outlined.

2 Multiple object tracking model

The 3D image cube is modeled as a bounded set $\mathcal{K} = [0, I_{h_{max}}] \times [0, I_{w_{max}}] \times \{t_1, \ldots, t_T\}$ and denote $x = (c_h, c_w, t, a, b, \omega, l)$ an ellipse of $\mathcal{K} \times \mathcal{M}$, where (c_h, c_w) represents the location of the ellipse within the image, t is the frame number, $a \in [a_{min}, a_{max}]$ and $b \in [b_{min}, b_{max}]$ denote the length of the semi-major axis and semi-minor axis respectively, $\omega \in [0, \pi]$

[*] The first and last authors would like to thank Airbus Defence & Space for the partial funding of this research.

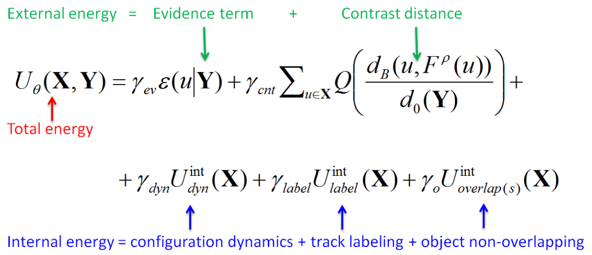
External energy = Evidence term + Contrast distance
$U_\theta(\mathbf{X},\mathbf{Y}) = \gamma_{ev}\varepsilon(u|\mathbf{Y}) + \gamma_{cnt}\sum_{u\in\mathbf{X}} Q\left(\frac{d_B(u,F^\rho(u))}{d_0(\mathbf{Y})}\right) +$
Total energy
$+\gamma_{dyn}U^{\text{int}}_{dyn}(\mathbf{X}) + \gamma_{label}U^{\text{int}}_{label}(\mathbf{X}) + \gamma_o U^{\text{int}}_{overlap(s)}(\mathbf{X})$
Internal energy = configuration dynamics + track labeling + object non-overlapping

FIGURE 1 – The energy term used for detecting and tracking objects. Further details can be found in [2].

is the orientation of the ellipse and l is its label. A configuration of ellipses $\mathbf{x}$ is an unordered set of ellipses in $\mathcal{K}\times\mathcal{M}$: $\mathbf{x} = \{x_1, \ldots, x_{n(\mathbf{x})}\}, x_i \in \mathcal{K}\times\mathcal{M}$, where $n(\mathbf{x}) = card(\mathbf{x})$ denotes the number of ellipses in the configuration. All ellipses with the same label form one trajectory. Finally, a marked point process X is a collection of random configurations on the same probability space $(\Omega, \mathcal{A}, \mathbb{P})$ [9].

The Gibbs family of processes is used to define the energy of the process as follows :

$$f_\theta(X = \mathbf{X}|\mathbf{Y}) = \frac{1}{c(\theta|\mathbf{Y})}\exp^{-U_\theta(\mathbf{X},\mathbf{Y})} \qquad (1)$$

where :

- $\mathbf{X} = \{\mathbf{x_1} \cup \mathbf{x_2} \cup \cdots \cup \mathbf{x_t} \cup \cdots \cup \mathbf{x_T}\}$ is the configuration of ellipses, with $\mathbf{x_t}$ being the configuration of ellipses at time t ;
- $\mathbf{Y}$ represents the 3D image cube ;
- θ is the parameter vector ;
- $c(\theta|\mathbf{Y}) = \int_\Omega \exp^{-U_\theta(\mathbf{X},\mathbf{Y})}\mu(d\mathbf{X})$ is called the normalizing constant, with Ω being the configuration space and $\mu(\cdot)$ being the intensity measure of the reference Poisson process ;
- $U_\theta(\mathbf{X},\mathbf{Y})$ is the energy term.

Using the MAP criterion, the most likely configuration of ellipses that corresponds to the global minimum of the energy is found as follows :

$$X \in arg\max_{\mathbf{X}\in\Omega} f_\theta(X = \mathbf{X}|\mathbf{Y}) = arg\min_{\mathbf{X}\in\Omega}[U_\theta(\mathbf{X},\mathbf{Y})]. \qquad (2)$$

The parameter vector $\theta = \{\theta_{ext}, \theta_{int}\}$ is composed of the parameter vectors for the external and internal energy terms, namely θ_{ext} and θ_{int} respectively. An efficient way to learn this parameter vector θ from the data sets can be found in [2]. In this paper, we used the energy term which is depicted in Figure 1 and described in detail in [2] for multiple object tracking.

3 Optimization

The energy depicted in Figure 1 is clearly not convex. It is easy to construct examples that have two virtually equal minima, separated by a wall of high energy values. The main reason that drives the energy to be non-convex is the dependence caused by the high-order physical constraints of the model.

The target distribution is the posterior distribution of $\mathbf{X}$, i.e. $\pi(\mathbf{X}) = f(\mathbf{X}|\mathbf{Y})$, which is defined on a union of subspaces of different dimensions. We use the reversible jump Markov Chain Monte Carlo (RJMCMC) sampler, developed by Green [4], to sample the posterior distribution. RJMCMC is a widely known optimization method for non-convex energy functions and an unknown number of objects [8]. It uses a mixture of perturbation kernels $Q(\cdot,\cdot) = \sum_m p_m Q_m(\cdot,\cdot)$, $\sum_m p_m = 1$ and $\int Q_m(\mathbf{X},\mathbf{X}')\mu(d\mathbf{X}') = 1$, to create tunnels through the walls of high energy.

We use simulated annealing [8] to find a minimizer of the energy function. The density function in eq. 1 can be rewritten as :

$$f_{\theta,i}(X = \mathbf{X}|\mathbf{Y}) = \frac{1}{c_{Temp_i}(\theta|\mathbf{Y})}\exp^{-\frac{U_\theta(\mathbf{X},\mathbf{Y})}{Temp_i}} \qquad (3)$$

where $Temp_i$ is a temperature parameter that tends to zero when i tends to infinity. If $Temp_i$ decreases in logarithmic rate, then X_i tends to a global optimizer of $f_{\theta,i}$. In practice however, a logarithmic law is not computationally feasible and hence, a geometric law is used instead.

A mapping $R_m(\cdot,\cdot) : \mathcal{C}\times\mathcal{C} \to (0,\infty)$, called the Green ratio, is associated to each of these perturbation kernels. This ratio is designed in order to ensure the balance of the Markov chain and obtain its ergodic convergence towards the desired distribution. At iteration i, the proposition $X_i = \mathbf{X}'$ is accepted with probability $\alpha_m = \min(1, R_m(\mathbf{X},\mathbf{X}'))$. Otherwise $X_i = \mathbf{X}$ [8].

Therefore, the efficiency of this iterative process depends on the variety of the perturbation kernels used. We propose below a new perturbation kernel that it specifically adapted to the multiple object tracking problem.

3.1 Birth and Death using a Kalman filter

As opposed to the classical birth and death kernel, we propose a problem-specific birth and death kernel that uses a Kalman filter within the birth step. This allows us to create tracklets (i.e. ellipses with the same label in consecutive frames) in a single step. The Kalman filter dates back to 1960, when R. E. Kalman described a recursive solution to the discrete-data linear filtering problem [6]. This filter became very popular and multiple variations and extensions of it have been designed to adjust the filter to diverse problems [1, 5].

The Kalman filter (KF) is applied to estimate the state of an object, where the state is assumed to be linearly Gaussian distributed in time. The continuity of the motion serves as a strong prediction criterion in object tracking. To generate new tracklets, we model the system as linear Gaussian, with the state parameters of the Kalman filter given by the ellipse location, its velocity, its size and its orientation. For a single object, the discrete-time dynamic equation is given by :

$$\mathbf{KX}_{t+1} = \mathbf{F}\cdot\mathbf{KX}_t \qquad (4)$$

where the state vector is given by $\mathbf{KX} = [c_x, c_y, \dot{c_x}, \dot{c_y}, a, b, \omega]$, where c_x and c_y are the predicted coordinates of the ellipse, $\dot{c_x}$ and $\dot{c_y}$ are the velocities in the respective direction, a and b are the semi-major and semo-minor axis of the ellipse and ω is the orientation ; and F is the a priori known transition matrix, where we set $dt = 1$ in our experiments, as given below :

$$\mathbf{F} = \begin{pmatrix} 1 & 0 & dt & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & dt & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The measurement vector $\mathbf{M}$ is obtained using a simple thresholded frame differencing technique to identify moving objects in every frame. At each time frame, we identify the foreground blobs for which we obtain their center location, width, height and orientation. We then construct a measurement vector $\mathbf{M_t}$, that can be injected into the measurement model of the KF. Accordingly :

$$\mathbf{M}_t = \mathbf{H} \cdot \mathbf{KX}_t + q_t \tag{5}$$

with $q_t \sim \mathcal{N}(0, \mathbf{R_t})$ being white Gaussian noise with covariance matrix $\mathbf{R}_t$ and $\mathbf{H}$ is the measurement function. Since we assume a known model for the dynamics of an object, we can use the KF to predict the position of the object in the next frame. The KF state prediction $\bar{\mathbf{KX}}_{t+1}$ and the state covariance prediction $\bar{\mathbf{P}}_{t+1}$ are defined by :

$$\bar{\mathbf{KX}}_{t+1} = \mathbf{F} \cdot \hat{\mathbf{KX}}_t \qquad \bar{\mathbf{P}}_{t+1} = \mathbf{F} \cdot \hat{\mathbf{P}}_t \cdot \mathbf{F}^T \tag{6}$$

where $\hat{\mathbf{KX}}_t$ and $\hat{\mathbf{P}}_t$ are the estimated state vector and error covariance matrix at time t respectively.
Then, the KF update step is as follows :

$$\mathbf{K}_{t+1} = \bar{\mathbf{P}}_{t+1} \cdot \mathbf{H}^T \left(\mathbf{H} \cdot \bar{\mathbf{P}}_{t+1} \cdot \mathbf{H}^T + \mathbf{R}_{t+1} \right)^{-1} \tag{7}$$

$$\hat{\mathbf{KX}}_{t+1} = \bar{\mathbf{KX}}_{t+1} + \mathbf{K}_{t+1} \left(\mathbf{M}_{t+1} - \mathbf{H}_{t+1} \cdot \bar{\mathbf{KX}}_{t+1} \right) \tag{8}$$

$$\hat{\mathbf{P}}_{t+1} = \left(\mathbf{I} - \mathbf{K}_{t+1} \cdot \mathbf{H} \right) \cdot \bar{\mathbf{P}}_{t+1}. \tag{9}$$

The KF starts with the initial conditions given by $\mathbf{K}_0$ and $\bar{\mathbf{P}}_0$. $\mathbf{K}_t$ is called Kalman gain and defines the updating weight between the new measurements and the prediction from the dynamic model.
The birth and death using a Kalman filter kernel first chooses with probability p_b and $p_d = 1 - p_b$ whether an object u should be added to (birth) or deleted from (death) the configuration. If a death is chosen, the kernel selects one object u in $\mathbf{X}$ and proposes $\mathbf{X}' = \mathbf{X} \setminus u$. However, if a birth is chosen, the kernel generates a new object u and proposes $\mathbf{X}' = \mathbf{X} \cup u$. If the birth is accepted, a Kalman filter is initialized to the location, size and orientation of u. The state at time t is updated using the measurements $\mathbf{M}_t$ and then a prediction step is executed. The kernel generates a new object v based on the state vector $\bar{\mathbf{KX}}_{t+1}$ of the KF and proposes $\mathbf{X}'' = \mathbf{X}' \cup v$. The state of the filter is updated using $\mathbf{M}_{t+1}$ and a new prediction is made. The process is repeated until a birth proposal is rejected. Note that the standard birth and death kernel is a particular case of the proposed perturbation kernel, when the proposal $\mathbf{X}'' = \mathbf{X}' \cup v$ is rejected.

3.2 Non-jumping transformations

Non-jumping transformations are transformations that do not modify the number of objects in the configuration. Such transformations randomly select an object u in the current configuration and then propose to replace it by a perturbed version of the object v : $\mathbf{X}' = (\mathbf{X} \setminus u) \cup v$. Translation, rotation and scale are standard examples of such transformations.

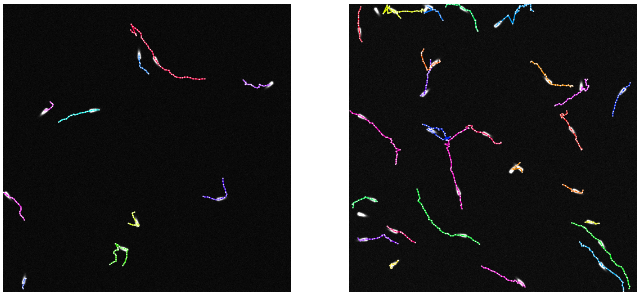

FIGURE 2 – Detection and tracking results on two synthetic biological sequences. The sequences are generated using the Icy software [3]. Left : Tracking results on the first image sequence up to frame 50. Right : Tracking results up to frame 50 of the second image sequence.

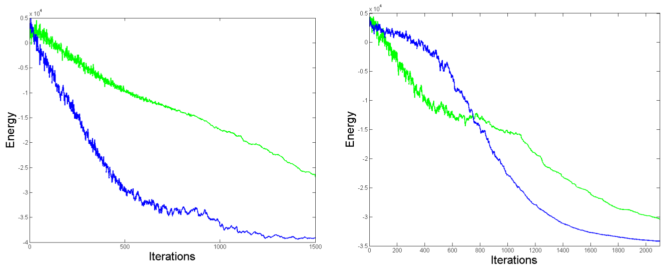

FIGURE 3 – Energy evolution with the number of iterations for the two samplers for the synthetic biological sequences shown in Figure 2. The proposed sampler is shown blue.

4 Experimental results

We have verified the balance of the newly introduced perturbation kernel by successfully simulating Poisson distributions of various intensities. In this paper, we apply our proposed optimization scheme for the detection and tracking of objects on two distinct data sets. The first one consists of two sequences of synthetic biological images of cells. We generated these sequences using the Icy software ([3]). The second data set consists of two simulated remotely sensed image sequences of boats with a spatial ground resolution of 0.5 meter. We compare the proposed optimization scheme with the RJMCMC sampler proposed in [2]. The performance of each sampler is evaluated based on the number of iterations necessary until convergence as well as the final energy level reached.

4.1 Results on synthetic benchmark data sets

Each sequence has 50 frames with a size of 512×512 pixels. The first sequence contains approximately 10 objects per frame,

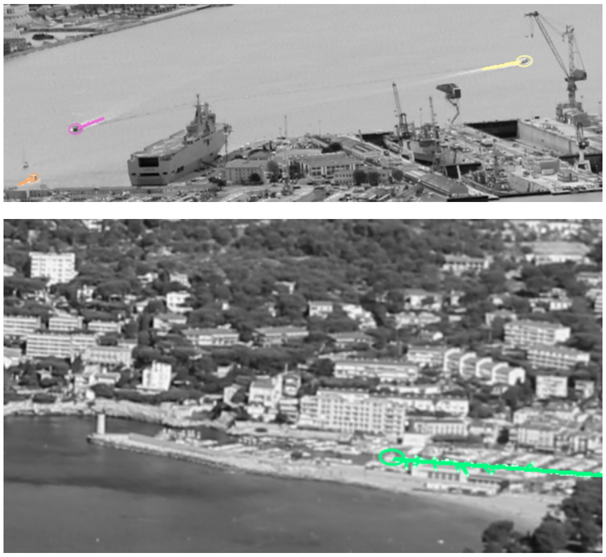

FIGURE 4 – Detection and tracking results on two sequences of simulated satellite images of Toulon. The image sequences are by courtesy of Airbus Defence & Space, France. Top : Tracking results on the first image sequence up to frame 50. Bottom : Tracking results up to frame 150 of the second image sequence.

while the second sequence contains approximately 30 objects per frame. Objects can appear or dissapear at any time and location. The objects exhibit a directed uniform motion. The detection and tracking results are shown in Figure 2. The performance of the two samplers is presented in Figure 3. The number of iterations necessary for the proposed sampler to converge is significantly smaller than that of the sampler presented in [2].

4.2 Results on simulated remotely sensed data

In this case, the interest lies in detecting and tracking the moving boats in the scene. The first sequence consists of 50 frames each of 670 × 221 pixels and contains 3 moving boats. The second sequence consists of 150 frames each of 451 × 251 pixels. A single boat is moving throughout this sequence. The detection and tracking results are shown in Figure 4 and the performance of the two samplers is presented in Figure 5. The proposed sampler needs a significantly lower number of iterations until convergence. The birth and death kernel using a Kalman filter not only allows the generation of multiple ellipses in a single iteration, but the ellipses are similar in size and orientation and better reflect the real objects in the scene, which reduces the necessary number of both jumping and non-jumping perturbations.

5 Conclusions

In this paper we have proposed a new optimization scheme that integrates the Kalman filter into a batch-optimization framework based on the RJMCMC sampler and used it in the framework of spatio-temporal marked point processes to detect and track multiple objects in image sequences. This hybrid optimization scheme benefits from the strengths of both approaches. The sequential filtering is used to build more meaningful perturbations which leads to a reduced number of iterations necessary until convergence, while the batch approach ensures the convergence towards a strong minimum and bypasses the typical problems of sequential approaches. We have shown results on two very different data sets in biology and remote sensing. We demonstrated the superiority of our proposed sampler w.r.t. the sampler proposed in [2]. The good performance of the proposed sampler shows the potential of hybrid optimization schemes which we plan to further investigate.

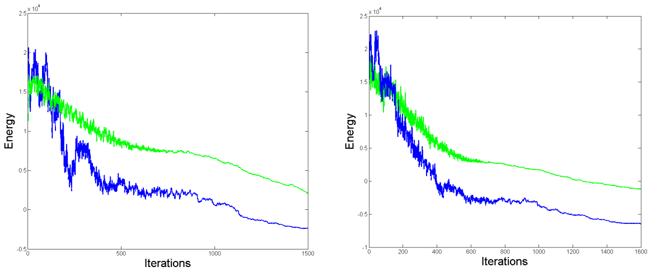

FIGURE 5 – Energy evolution with the number of iterations for the two samplers for the two simulated sequences of remotely sensed images. The proposed sampler is shown blue.

Références

[1] Y. Bar-Shalom and T. Fortmann. *Tracking and Data Association*. Academic Press, San Diego, 1988.

[2] P. Crăciun, M. Ortner, and J. Zerubia. Joint detection and tracking of moving objects using spatio-temporal marked point processes. *Proc. WACV*, pages 177–184, 2015. Available at `https://hal.inria.fr/hal-01104981`.

[3] F. de Chaumont et al. Icy : an open bioimage informatics platform for extended reproducible research. *Nature methods*, 9 :690–696, 2012.

[4] P. Green. Reversible jump Markov Chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4) :711–732, 1995.

[5] A.J. Haug. *Bayesian estimation and tracking : A practical guide*. Wiley, 2012.

[6] R.E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960.

[7] W. Luo, X. Zhao, and T.-K. Kim. Multiple object tracking : A review. *CoRR*, abs/1409.7618, 2014.

[8] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Second edition, 2005.

[9] M.N.M. van Lieshout. *Markov Point Processes and Their Applications*. Imperial College Press, 2000.