Variational Bayesian Tracking: Whole Track Convergence for Large-scale Ecological Video Monitoring

Jacqueline Christmas, Richard Everson, Rolando Rodríguez-Muñoz, Tom Tregenza

Abstract— **Variational Bayesian approximations offer a computationally fast alternative to numerical approximations for Bayesian inference. We examine variational Bayesian methods for filtering and smoothing continuous hidden Markov models, in particular those with sharply-peaked, nonlinear observations densities. We show that, by making variational updates in the correct order, robust convergence to the tracked state may be achieved.**

We apply the whole track convergence algorithm to tracking wild crickets in video streams and describe how animals may be identified from the characteristics of their tracks. We also show how identifying alphanumeric tags may be read under poor lighting conditions.

I. Introduction

MONITORING entire populations of animals by video is now possible and brings the exciting possibility of relating genetic variation of individuals to their behaviour in the wild, not just the sterile laboratory [1]. Such projects generate hundreds of thousands of hours of video data, bringing with it the challenge of automatic tracking and identification of the animals. Bayesian sequential estimation is a state of the art technique for estimating the unknown hidden state of the tracked object. The Bayesian formulation promises not only an estimate of the most likely or mean state, but also a posterior density which describes the uncertainty in the estimate. However, in many situations nonlinear and non-Gaussian observations models lead to analytically intractable integrals. While a number of methods, especially particle filters (e.g. [2], [3]) are available for approximating the integrals, many of them are computationally expensive because they rely on sampling. An attractive method for inference in this case is the variational Bayes approximation, in which the posterior distribution is approximated by a simpler parameterised distribution found by minimising the Kullback-Leibler divergence between the true distribution and the approximation. In particular, the widely used mean field approach assumes independence (conditioned on the observations) of the posterior distributions for each of the variables involved, effectively breaking links between variables which are not directly dependent.

As Turner and Sahani have shown [4], the independence assumptions inherent in the mean field models appears to break the temporal dependencies that drive the filtering and smoothing algorithms, preventing the propagation of uncertainty through time and leading to unrealistically precise estimates of the hidden state. In this paper we show that the mean field approximation may be used to successfully track, via filtering and smoothing, provided that the variational updates are made in an appropriate order. We call this "whole track convergence".

We illustrate whole track convergence on a toy problem, which may be treated semi-analytically, and we show how it may be used for tracking crickets monitored using a network of IP cameras. Tracking in these circumstances is difficult because of relatively low video resolution and the fact that the animals being tracked are approximately the same size as grass blown by the wind. We also describe how tags affixed to the back of the tracked crickets are read under poor lighting conditions.

II. Background

We consider the Bayesian estimation of sequential hidden Markov models. The hidden state at time t is denoted by $\mathbf{x}_t$ and the observation by $\mathbf{y}_t$. The hidden Markov model is then described by a state transition probability $p(\mathbf{x}_{t+1} \,|\, \mathbf{x}_t)$ and the likelihood of an observation conditioned on the state $p(\mathbf{y}_t \,|\, \mathbf{x}_t)$. As is well known, given a probability density for the state having made observations up to time t, $p(\mathbf{x}_t \,|\, \mathbf{y}_1, \ldots, \mathbf{y}_t) \equiv p(\mathbf{x}_t \,|\, Y_t) \equiv \alpha(\mathbf{x}_t)$, the prediction of the state at time $t+1$ is:

$$p(\mathbf{x}_{t+1} \,|\, Y_t) = \int p(\mathbf{x}_{t+1} \,|\, \mathbf{x}_t)\alpha(\mathbf{x}_t)\, d\mathbf{x}_t \qquad (1)$$

On making a new observation, the predicted state is corrected using Bayes' rule:

$$\begin{aligned}\alpha(\mathbf{x}_{t+1}) &= p(\mathbf{x}_{t+1} \,|\, Y_{t+1}) &(2)\\ &= p(\mathbf{x}_{t+1} \,|\, y_{t+1}, Y_t) \propto p(\mathbf{y}_{t+1} \,|\, \mathbf{x}_{t+1})p(\mathbf{x}_{t+1} \,|\, Y_t) &(3)\end{aligned}$$

Using these prediction and correction equations sequentially allows probability densities describing the hidden state to be updated as new data becomes available. If all the observations $Y_T = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$ are available, information from later times may be used to augment the state estimate at earlier times. This is accomplished through the recursive update of $\beta(\mathbf{x}_t) \equiv p(\mathbf{x}_t \,|\, \mathbf{y}_{t+1}, \ldots, \mathbf{y}_T)$ as follows:

$$\beta(\mathbf{x}_t) = \int \beta(\mathbf{x}_{t+1})p(\mathbf{y}_{t+1} \,|\, \mathbf{x}_{t+1})p(\mathbf{x}_{t+1} \,|\, \mathbf{x}_t)\, d\mathbf{x}_{t+1} \qquad (4)$$

The probability of the hidden state given all the observed data is then simply found from $p(\mathbf{x}_t \,|\, Y_t) = \alpha(\mathbf{x}_t)\beta(\mathbf{x}_t)$.

Jacqueline Christmas and Richard Everson are with the Department of Computer Science, The University of Exeter, Exeter, UK; Rolando Rodríguez-Muñoz and Tom Tregenza are with the Centre for Ecology & Conservation, The University of Exeter, Penryn, UK (email: {J.T.Christmas, R.M.Everson, T.Tregenza, R.Rodriguez-Munoz}@exeter.ac.uk).

This work was supported by a grant for the Natural Environment Research Council of the UK

Although formally straightforward, carrying out these forward and backward recursions in all but the simplest cases (e.g. when the state and observation densities are Gaussian which results in the Kalman filter) is difficult. The principal obstacle is computing the denominator in Bayes' rule (3), which normalises the posterior density $p(\mathbf{x}_{t+1} \,|\, Y_{t+1})$. Particle filters and related sampling methods allow arbitrary densities to be handled, but the computational expense of sampling is often prohibitive.

The variational Bayesian methodology [5], [6], [7], [8], [9] is an attractive alternative in which the desired posterior density is approximated by a simpler parameterised density. Let Y be the observed data and $\boldsymbol{\theta}$ the set of random variables whose posterior distribution $p(\boldsymbol{\theta} \,|\, Y)$ is sought and $q(\boldsymbol{\theta})$ be the approximating density function. We assume that the approximate posterior density can be factorised into G groups, which are assumed to be independent when conditioned on Y. Thus

$$q(\boldsymbol{\theta}) = \prod_{i=1}^{G} q_i(\boldsymbol{\theta}_i) \tag{5}$$

The log marginal probability of $\mathbf{x}$ may be written as

$$\begin{aligned} \log(p(Y)) &= \overbrace{\int q(\boldsymbol{\theta} \,|\, Y) \log\left(\frac{p(Y, \boldsymbol{\theta})}{q(\boldsymbol{\theta} \,|\, Y)}\right) d\boldsymbol{\theta}}^{\text{negative variational free energy}} \\ &\quad + \overbrace{\int q(\boldsymbol{\theta} \,|\, Y) \log\left(\frac{q(\boldsymbol{\theta} \,|\, Y)}{p(Y \,|\, \boldsymbol{\theta})}\right) d\boldsymbol{\theta}}^{\text{KL divergence}} \qquad (6) \\ &= \mathcal{F}(q) + KL(q(\boldsymbol{\theta} \,|\, Y) \parallel p(\boldsymbol{\theta} \,|\, Y)). \qquad (7) \end{aligned}$$

As indicated, the log marginal probability may be recognised as the sum of the Kullback-Leibler (KL) divergence between the approximate posterior and the true posterior, and the negative variational free energy. Since the KL divergence is non-negative (and zero if and only if $q(\boldsymbol{\theta})$ equals $p(\boldsymbol{\theta} \,|\, Y)$) the negative free energy is a lower bound on the log marginal probability and maximising $\mathcal{F}(q)$ by adjusting the approximate posterior $q(\boldsymbol{\theta} \,|\, Y)$ necessarily minimises $KL(q \parallel p)$ so that q better approximates the posterior.

Attias [8] (see also [10], [11]) exploits the factorisation of the posterior (5) to find a general expression for the maximiser of the negative free energy in a mean-field sense. We seek to maximise the negative variational free energy, $\mathcal{F}(q(\boldsymbol{\theta}))$, with respect to all the $q_i(\boldsymbol{\theta}_i)$. For readability Q_i represents $q_i(\boldsymbol{\theta}_i)$:

$$\begin{aligned} \mathcal{F}(q) &= \int q(\boldsymbol{\theta}) \log(\frac{p(Y, \boldsymbol{\theta})}{q(\boldsymbol{\theta})})\, d\boldsymbol{\theta} \qquad (8) \\ &= \int \left(\prod_{i=1}^{G} Q_i\right) \log(p(Y, \boldsymbol{\theta}))\, d\boldsymbol{\theta}_1, \ldots, d\boldsymbol{\theta}_G \\ &\quad - \int \left(\prod_{i=1}^{G} Q_i\right) \left(\sum_{i=1}^{G} \log(Q_i)\right) d\boldsymbol{\theta}_1, \ldots, d\boldsymbol{\theta}_G \quad (9) \end{aligned}$$

Considering the integral with respect to $\boldsymbol{\theta}_j$ and keeping the remaining $Q_{i \neq j}$ fixed, the negative free energy can be written as

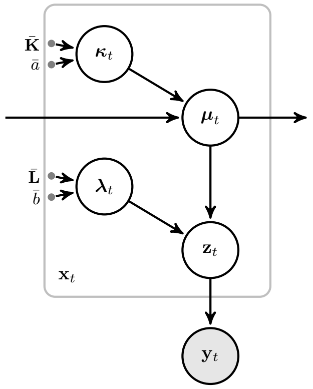


Fig. 1: Graphical model for a composite state model. The composite state comprises an average state $\boldsymbol{\mu}_t$ and a surrogate observation $\mathbf{z}_t$ upon which the true observation depends.

$$\begin{aligned} \mathcal{F}(q) = &\int Q_j \left[\int \log(p(Y, \boldsymbol{\theta})) \prod_{i \neq j} Q_i d\boldsymbol{\theta}_{i \neq j}\right] d\boldsymbol{\theta}_j \\ &- \int Q_j \log(Q_j)\, d\boldsymbol{\theta}_j + const \qquad (10) \end{aligned}$$

where terms that do not depend upon Q_j have been absorbed into the constant. The term in square brackets is the expectation of $\log(p(Y, \boldsymbol{\theta}))$ with respect to each of the Q_j, where $i \neq j$. We denote this $\mathbb{E}_{i \neq j}[\log(p(Y, \boldsymbol{\theta}))]$, and it may be recognised as the negative KL divergence between Q_j and $\mathbb{E}_{i \neq j}[\log(p(Y, \boldsymbol{\theta}))]$; hence the maximum value is zero, which is obtained when

$$\log(Q_j) = \mathbb{E}_{i \neq j}[\log(p(Y, \boldsymbol{\theta}))]. \tag{11}$$

If conjugate priors are chosen for each group, the approximate posterior turns out to have the same functional form as the prior [8], [12] and the variational approximations may thus be found by evaluating (11) for each group in turn. Of course, the hyperparameters of the posterior distribution for one group will generally depend upon the hyperparameters for other groups; consequently the parameters for each group are evaluated cyclically until convergence. This scheme converges to a local maximum of $\mathcal{F}$, thus minimising $KL(q \parallel p)$. As we discuss below, the order in which these updates are made has profound implications for the efficacy of the method.

III. Whole track convergence

In this section we describe a general hidden Markov model for tracking objects in images and investigate the convergence of variational approximations to it.

In common with many image tracking problems, the observations probability is highly nonlinear and sharply peaked. We therefore follow Vermaak et al. [13] by writing the state $\mathbf{x}_t$ as a composite of an average state $\boldsymbol{\mu}_t$ together with a surrogate, intermediate observation $\mathbf{z}_t$, which depends upon

$\boldsymbol{\mu}_t$ and the state noise variable $\boldsymbol{\lambda}_t$.

$$p(\mathbf{z}_t \,|\, \boldsymbol{\mu}_t, \boldsymbol{\lambda}_t) = \mathcal{N}(\mathbf{z}_t \,|\, \langle\boldsymbol{\mu}_t\rangle, \langle\boldsymbol{\lambda}_t\rangle) \tag{12}$$

Figure 1 shows the graphical model for a single time t; this is coupled to earlier and later times through the state transition density, which we model here as a simple diffusion:

$$p(\boldsymbol{\mu}_t \,|\, \boldsymbol{\mu}_{t-1}, \boldsymbol{\kappa}_t) = \mathcal{N}(\boldsymbol{\mu}_t \,|\, \boldsymbol{\mu}_{t-1}, \boldsymbol{\kappa}_{t-1}^{-1}). \tag{13}$$

The precisions $\boldsymbol{\kappa}_t$ and $\boldsymbol{\lambda}_t$ are assigned conjugate Wishart priors

$$p(\boldsymbol{\kappa}_t; \bar{\mathbf{K}}, \bar{a}) = \mathcal{W}(\boldsymbol{\kappa}_t; \bar{\mathbf{K}}, \bar{a}) \tag{14}$$
$$p(\boldsymbol{\lambda}_t; \bar{\mathbf{L}}, \bar{b}) = \mathcal{W}(\boldsymbol{\lambda}_t; \bar{\mathbf{L}}, \bar{b}). \tag{15}$$

Note that the noise Wishart prior over the state noise precision may be integrated out to show that state transition density is a multi-variate Student-t density, which allows occasional large changes in the state.

The state is assigned a prior

$$p(\boldsymbol{\mu}_0) = \mathcal{N}(\boldsymbol{\mu}_0 \,|\, \bar{\boldsymbol{\mu}}_0, \bar{\kappa}_0^{-1}). \tag{16}$$

The surrogate observation $\mathbf{z}_t$ serves to separate the problematic likelihood from the variational inference for the variables $\boldsymbol{\mu}_t$, $\boldsymbol{\lambda}_t$ and $\boldsymbol{\kappa}_t$, which then only require the expectations $\langle\mathbf{z}_t\rangle$ and $\langle\mathbf{z}_t\mathbf{z}_t^T\rangle$ for their calculation. Like Vermaak et al. [13] we estimate these expectations using importance sampling as described below.

The observations probability is now given by $p(\mathbf{y}_t \,|\, \mathbf{z}_t)$. With $\mathbf{z}_t$ defining the region (e.g. an oriented ellipse) of the video frame $\mathbf{y}_t$ that falls within the boundaries of the tracked object (i.e. the *foreground*), we define different probability distributions for some attribute(s) of the foreground and background pixels, giving

$$p(\mathbf{y}_t \,|\, \mathbf{z}_t) = \prod_{i \in \text{fg}} p_f(y_{t,i}) \prod_{j \in \text{bg}} p_b(y_{t,j}) \tag{17}$$

where $y_{t,j}$ is the jth pixel of $\mathbf{y}_t$.

The mean field variational Bayes method approximates the posterior as the factorised distribution:

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_T \,|\, Y_T) \approx \prod_{t=1}^{T} q(\boldsymbol{\mu}_t) q(\mathbf{z}_t) q(\boldsymbol{\lambda}_t) q(\boldsymbol{\kappa}_t) \tag{18}$$

The approximate posteriors are found in terms of "forward" variables $q^{\alpha}(\cdot)$, which depend upon the variables at the previous time, and "backward" variables $q^{\beta}(\cdot)$ which depend on variables at the following time. These are given as follows:

$$\begin{aligned} q^{\alpha}(\boldsymbol{\mu}_t) &= \mathcal{N}(\boldsymbol{\mu}_t \,|\, \mathbf{m}_t^{\alpha}, \mathbf{S}_t^{\alpha}) && (19)\\ \mathbf{S}_t^{\alpha} &= \Big(\langle\boldsymbol{\lambda}_t\rangle + \langle\boldsymbol{\kappa}_t\rangle\Big)^{-1} \\ \mathbf{m}_t^{\alpha} &= \mathbf{S}_t^{\alpha}\Big(\langle\boldsymbol{\lambda}_t\rangle\langle\mathbf{z}_t\rangle + \langle\boldsymbol{\kappa}_t\rangle\langle\boldsymbol{\mu}_{t-1}\rangle\Big) \\ q^{\beta}(\boldsymbol{\mu}_t) &= \mathcal{N}(\boldsymbol{\mu}_t \,|\, \mathbf{m}_t^{\beta}, \mathbf{S}_t^{\beta}) && (20)\\ \mathbf{S}_t^{\beta} &= \langle\boldsymbol{\kappa}_{t+1}\rangle^{-1} \\ \mathbf{m}_t^{\beta} &= \langle\boldsymbol{\mu}_{t+1}\rangle \\ q(\boldsymbol{\mu}_t) &= \mathcal{N}(\boldsymbol{\mu}_t \,|\, \mathbf{m}_t, \mathbf{S}_t) && (21)\\ \mathbf{S}_t &= \Big((\mathbf{S}_t^{\alpha})^{-1} + (\mathbf{S}_t^{\beta})^{-1}\Big)^{-1} \\ \mathbf{m}_t &= \mathbf{S}_t\Big((\mathbf{S}_t^{\alpha})^{-1}\mathbf{m}_t^{\alpha} + (\mathbf{S}_t^{\beta})^{-1}\mathbf{m}_t^{\beta}\Big) \\ q(\boldsymbol{\mu}_0) &= \mathcal{N}(\boldsymbol{\mu}_0 \,|\, \mathbf{m}_0, \mathbf{S}_0) && (22)\\ \mathbf{S}_0 &= (\langle\boldsymbol{\kappa}_1\rangle + \bar{\boldsymbol{\kappa}}_0)^{-1} \\ \mathbf{m}_0 &= \mathbf{S}_t\Big(\langle\boldsymbol{\kappa}_1\rangle\langle\boldsymbol{\mu}_1\rangle + \bar{\boldsymbol{\kappa}}_0\bar{\boldsymbol{\mu}}_0\Big) \end{aligned}$$

The remaining variables are associated with the state and observational noise:

$$\begin{aligned} q(\boldsymbol{\kappa}_t) &= \mathcal{W}(\boldsymbol{\kappa}_t \,|\, \mathbf{K}_t, a_t) && (23)\\ a_t &= \bar{a} + 1 \\ \mathbf{K}_t &= \Big[\bar{\mathbf{K}}^{-1} + \langle\boldsymbol{\mu}_t\boldsymbol{\mu}_t^T\rangle + \langle\boldsymbol{\mu}_{t-1}\boldsymbol{\mu}_{t-1}^T\rangle \\ &\qquad - \langle\boldsymbol{\mu}_t\rangle\langle\boldsymbol{\mu}_{t-1}\rangle^T - \langle\boldsymbol{\mu}_{t-1}\rangle\langle\boldsymbol{\mu}_t\rangle^T\Big]^{-1} \\ q(\boldsymbol{\lambda}_t) &= \mathcal{W}(\boldsymbol{\lambda}_t \,|\, \mathbf{L}_t, b_t) && (24)\\ b_t &= \bar{b} + 1 \\ \mathbf{L}_t &= \Big[\bar{\mathbf{L}}^{-1} + \langle\mathbf{z}_t\mathbf{z}_t^T\rangle + \langle\boldsymbol{\mu}_t\boldsymbol{\mu}_t^T\rangle \\ &\qquad - \langle\mathbf{z}_t\rangle\langle\boldsymbol{\mu}_t\rangle^T - \langle\boldsymbol{\mu}_t\rangle\langle\mathbf{z}_t\rangle^T\Big]^{-1} \end{aligned}$$

Standard derivations show that

$$\langle\boldsymbol{\kappa}_t\rangle = \mathbf{K}_t a_t \text{ and } \langle\boldsymbol{\lambda}_t\rangle = \mathbf{L}_t b_t \tag{25}$$

The expectations with respect to $\mathbf{z}_t$ cannot be calculated analytically, mainly because the likelihood $p(\mathbf{y}_t \,|\, \mathbf{z}_t)$ is nonlinear. However, following Vermaak et al. [13] importance sampling from $q(\mathbf{z}_t) = \mathcal{N}(\mathbf{z}_t \,|\, \langle\boldsymbol{\mu}_t\rangle, \langle\boldsymbol{\lambda}_t\rangle)$ may be used. If $\mathbf{z}_t^{(i)}$ is the ith sample (from a total of M) and $w_t^{(i)}$ is the corresponding weight:

$$w_t^{(i)} = \frac{p(\mathbf{y}_t \,|\, \mathbf{z}_t^{(i)})}{\sum_{j=1}^{M} p(\mathbf{y}_t \,|\, \mathbf{z}_t^{(j)})}, \tag{26}$$

then approximations to the expectations can be calculated as follows:

$$\langle\mathbf{z}_t\rangle \approx \sum_{i=1}^{M} w_t^{(i)}\mathbf{z}_t^{(i)} \quad \langle\mathbf{z}_t\mathbf{z}_t^T\rangle \approx \sum_{i=1}^{M} w_t^{(i)}\mathbf{z}_t^{(i)}(\mathbf{z}_t^{(i)})^T \tag{27}$$

We note that care must be taken to ensure that sufficiently many importance samples are drawn so that the average is not dominated by the single largest weight. In the work reported here we use 200 samples in the initial variational iterations and 20 as the iterations near convergence.

A. State uncertainty

Turner and Sahani have pointed out that the mean field approximation fails to correctly propagate the state uncertainty from one time step to the next [4]. That is, the uncertainty represented by $p(\mathbf{x}_{t-1} \,|\, Y_{t-1})$ is not properly factored into the $q(\mathbf{x}_t)$ which means that $q(\mathbf{x}_t)$ is too narrow. Although variational methods are well known to yield approximations to the posterior distribution that are more compact than the true posterior, this failure to propagate uncertainty may result in extremely compact distributions. Indeed examining (19), (20) and (21) shows that the variance $\mathbf{S}_t$ of $q(\boldsymbol{\mu}_t)$ (the approximation to $p(\boldsymbol{\mu} \,|\, Y_T)$) does not involve the variances of the state uncertainties at neighbouring times, namely $\mathbf{S}^{\alpha}_{t-1}$ and $\mathbf{S}^{\beta}_{t}$. There is, nonetheless, some coupling of the uncertainty at neighbouring times through the coupling of the parameters in (19) – (22), however, the state uncertainty is seriously reduced.

We show elsewhere [14] that the state uncertainty may be more effectively propagated using a structured variational approximation. Here we show that, despite the failure to propagate state uncertainty, the mean field variational approximation converges well to the entire track.

B. Order of variational updates

As with most variational schemes, the systems of coupled equations (19)–(25) cannot be solved exactly. Instead they are solved by cyclically updating the parameters for one variable using the current values for the others. This is repeated until convergence.

At first sight, a natural way to update the variables is to cyclically update all the variables for $t = 1$ until they are converged, before proceeding to $t = 2$ and so on. Since variables at time t depend only on variables at the previous time, it appears that the estimates for t can be "polished" before proceeding to $t{+}1$. Updating in this order is analogous to the forward sweep in the standard forward recursions (c.f., (1) and (2)). Variables from the backward sweep are similarly updated analogously to the backward recursions, after which the forward and backward sweep estimates are combined.

However, it is found that iterating estimates to convergence for each time before proceeding to the next marked inhibits the ability of the variational scheme to converge to the correct result. This may be viewed as a consequence of the underestimated state uncertainty: if the parameter estimates for time t result in a narrow $q(\mathbf{x}_t)$, the range of likely $\mathbf{x}_{t+1}$ is restricted and the estimates for $q(\mathbf{x}_{t+1})$ may converge to a local optimum, near to $q(\mathbf{x}_t)$, but far from the true track.

Instead, we find that it is effective to update in the following order, summarised in Algorithm 1.

This iterative procedure updates the variational distributions $q^{\alpha}(\boldsymbol{\mu}_t)$ for $\boldsymbol{\mu}_t$ and $\mathbf{x}_t$ for each t in turn in a forward sweep, followed by the variational distributions $q^{\beta}(\boldsymbol{\mu}_t)$ in a backward sweep. Note that in these only a single update of the parameters is made at each t. This allows the approximations to the state for all times to converge together, rather than attempting to completely converge estimates for the distributions for a single time at once. We dub this *whole track convergence.* In practice we find that this allows the state to be initialised quite crudely (i.e., vague priors) because the estimates converge along the entire track.

Algorithm 1 Smoother tracking algorithm

```
initialise variables
while not converged do
    ——— forward sweep ———
    for $t = 1$ up to $T$ do
        estimate $\langle \mathbf{x}_t \rangle$ and $\langle \mathbf{x}_t \mathbf{x}_t^T \rangle$ using (27)
        calculate $q^{\alpha}(\boldsymbol{\mu}_t)$ using (19)
    end for
    ——— backward sweep ———
    $q^{\beta}(\boldsymbol{\mu}_T) = q^{\alpha}(\boldsymbol{\mu}_T)$
    for $t = T - 1$ down to 1 do
        calculate $q^{\beta}(\boldsymbol{\mu}_t)$ using (20)
    end for
    —— combine forward and backward estimates ——
    for $t = 1$ up to $T$ do
        calculate $q(\boldsymbol{\mu}_t)$ using (21)
    end for
    calculate $q(\boldsymbol{\mu}_0)$
    ——— remaining variables ———
    for $t = 1$ up to $T$ do
        calculate $q(\boldsymbol{\lambda}_t)$ using (24)
        calculate $q(\boldsymbol{\kappa}_t)$ using (23)
    end for
end while
```

C. Illustration: toy model

We illustrate the whole track convergence on a relatively simple "toy" model, which nonetheless retains the characteristics of more realistic tracking situations.

In this model, each "image" is a one-dimensional column of pixels. Background pixels have intensities drawn from a Gamma density $y_{t,j} \sim \mathcal{G}(1, 0.1)$, while foreground pixel intensities are drawn from a "brighter" Gaussian distribution: $y_{t,j} \sim \mathcal{N}(100, 30^2)$. As shown in Figure 2, the centre of the foreground pixels follows a sinuous path, while the width of the foreground object oscillates more rapidly. In addition, there is a discontinuity in the track, a period when the object is not observed at all and a period when a second "distractor" object is present.

We model the state as the two dimensional vector consisting of the location of the centre of the tracked object and its width. The parameters governing the state and observational noise distributions are chosen as $\bar{a} = D, \bar{\mathbf{K}} = \boldsymbol{\Sigma}^{-1}/D$, and $\bar{b} = D, \bar{\mathbf{L}} = \boldsymbol{\Sigma}^{-1}/D$ where $\boldsymbol{\Sigma} = \mathrm{diag}(10^2, 5^2)$ and $D = 2$ is the dimension of the state.

Figure 3 compares the convergence on the toy data by iterating variables for each t until convergence with whole track convergence (Algorithm 1). The lefthand panel shows the centre of the converged track and its width using whole track convergence. The centre and righthand panels show

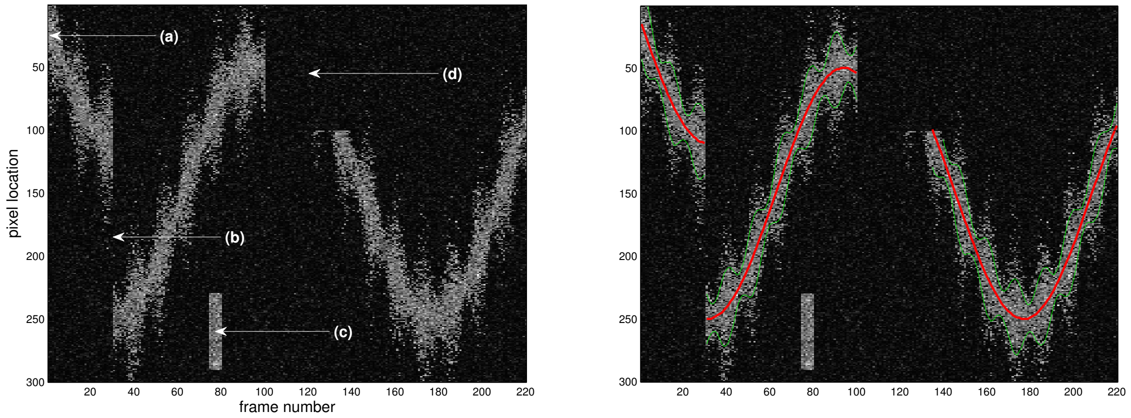
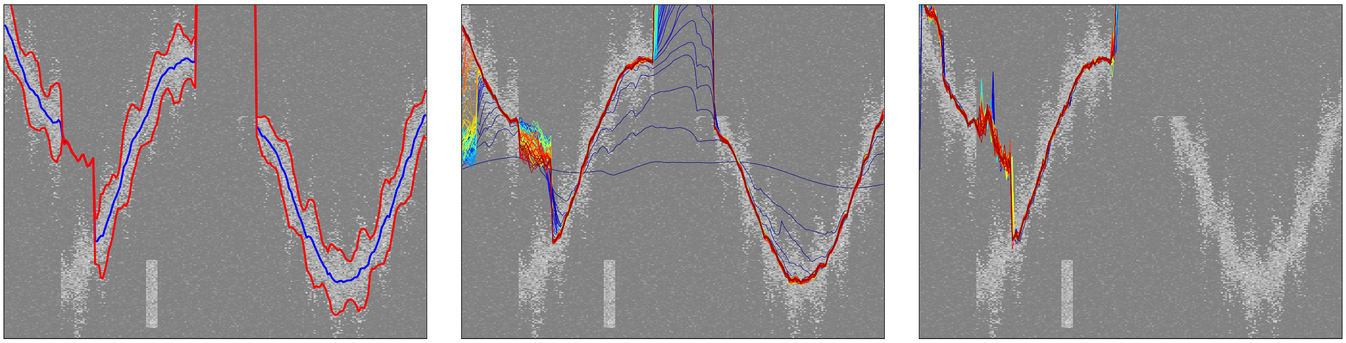


Fig. 2: Left: One-dimensional example track with four points of interest labelled: (a) the first frame in which the object must be acquired, (b) a discontinuity in the track, (c) a patch of foreground pixels that are not on the track but which have a higher likelihood than the track in those frames, and (d) a region where the object is not observed. Right: true track.



(a) Converged track and width

(b) Whole track location convergence

(c) Convergence of polishing updates

Fig. 3: Whole track convergence. (a) Whole track converged location and object width. (b) Whole track location convergence (c) Polishing convergence. Initial iterations are shown in blue, final iterations in red.

the convergence of the variational iterations for whole track convergence and the polishing scheme. The initial iterations are shown in blue, while the final iterations are coloured red. The polishing iterations converge to the initial part of the track and are able to converge to both sides of the discontinuity, however when the observations are missing the polishing updates fail to locate the track on the other side. In contrast the whole track convergence scheme is initialised with a track approximately in the centre of the “frame”. As the updates continue the track converges at all t. As the figure shows the iterations successfully locate the track at $t = 1$ despite being initialised distant from the true location. Convergence of the whole track means that it converges to both sides of the missing data.

IV. Tracking crickets

We used the whole track convergence smoothing algorithm described above to detect and track the crickets approximately 20,000 hours of video footage. The crickets tracked are a flightless species, *Gryllus campestris*, which lives in grassy meadows in Northern Spain. Both sexes spend nearly all their time outside a burrow. Burrows for an entire population of ≈ 200 crickets were monitored by IP cameras and video feeds recorded on disk [1]. Recordings were made in natural sunlight during the day and by infra-red light at night.

We model the cricket by an oriented ellipse, parameterised by its location, bearing and the lengths of its axes, so that the state is a 5-dimensional vector. Cricket motion is modelled by a diffusive state transition density (13) with Wishart priors over the diffusion covariance (14) so that the marginalised state transition density is a heavy-tailed Student-t density, allowing for occasional large jumps in the state (for example, when the cricket is surprised and bolts to the burrow).

Particularly during the day, the background is subject to rapid and spatially non-uniform changes in illumination. In addition the grass surrounding the burrow moves in the breeze, creating a noisy background. A few frames from a daylight video are shown in Figure 4. To help separate the moving cricket from the background we run a version of principal component analysis (PCA) on the video frames being analysed. This version of PCA models temporal correlations through auto-regression of the latent variables; in addition

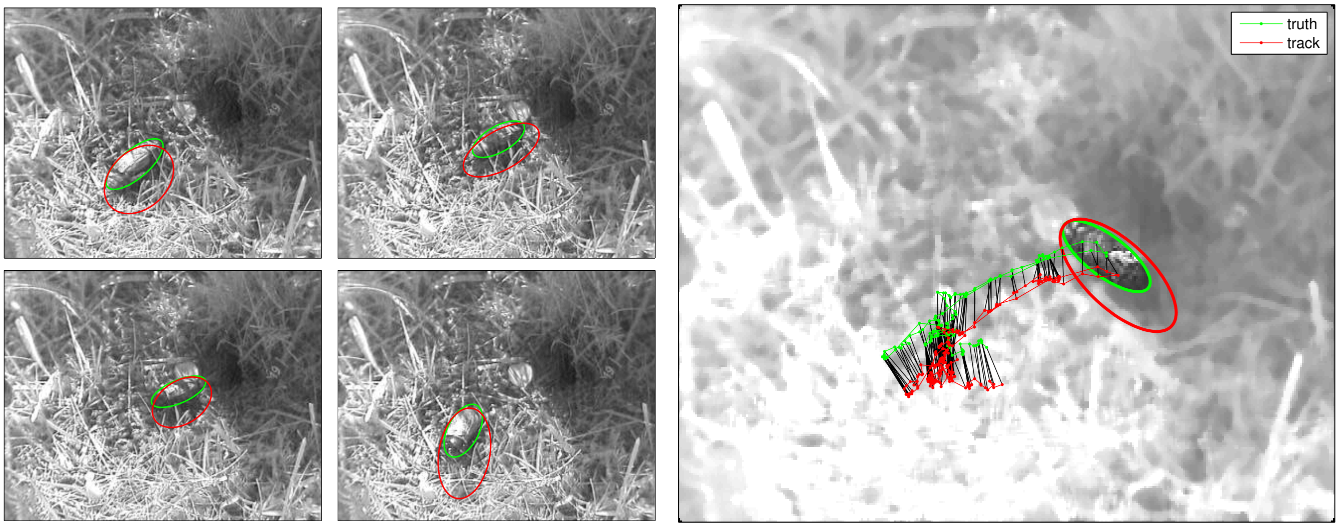


Fig. 4: Left: Four randomly selected frames from a much longer video sequence showing the manually identified cricket (outlined in green) and the result of our tracking algorithm (red). The cricket burrow is visible as a darker patch towards the upper right. Right: Manually identified (green) and automatically tracked (red) cricket locations for 210 frames. Locations at corresponding times are joined by black lines and the background picture has been bleached for clarity.

the observational noise is robustly modelled by a Student-t distribution and variational Bayes is used for inference [15], [16], [17], [18]. The probability of each observed pixel under this PPCA-AR model $p(y_{t,k} \mid Y_T)$ is used to quantify whether a pixel belongs to the foreground or to the background. The observations probability is then

$$p(\mathbf{y}_t \mid \mathbf{x}_t) = \prod_{k \in ellipse} p(y_{t,k} \in \text{fg}) \prod_{k \notin ellipse} p(y_{t,k} \in \text{bg})$$

Figure 4 shows the result of the variational algorithm for 210 frames randomly selected from the middle of a video sequence for which we also established the ground truth. As can be seen from the figure, the automatically extracted tracks closely follow the ground truth tracks except that they are offset by a small amount. This offset arises because in sunlight our algorithm tends to track not only the cricket body, but also the area occupied by its legs and antennae and the cricket's shadow. This means that the tracked ellipse is larger than the manually fitted ellipse (see lefthand panels of Figure 4), resulting in the observed offset.

We emphasise that without the use of the whole track convergence algorithm, very careful initialisation of the cricket's location is required and the variational "polishing" updates easily lose track of the cricket. By contrast, the whole track convergence algorithm effectively locates the cricket throughout the entire video sequence and is able to track it even when the cricket leaves the video frame for a few moments or is stationary and thus blends into the background.

V. Track classification

In addition to crickets the videos also record a wide variety of other fauna such as slugs and spiders. As these are of approximately the same size as crickets and move at comparable speeds to crickets, they too are tracked. In order to determine what sort of animal is being tracked we characterise each track by a (normalised) histogram of the distance moved between frames, the change in bearing between frames and the area of the tracked ellipse. Tracks are then classified to one of six classes using 1-nearest neighbour classification, where the distance between histograms is measured using the Kullback-Leibler divergence.

	Predicted class					
	nothing	cricket	slug	human	camera	other
nothing	76	8	15	19	1	32
cricket	15	88	13	8	0	1
slug	18	2	4	5	0	7
human	3	1	0	8	0	4
camera	2	0	0	1	0	0
other	26	4	9	13	0	14

TABLE I: The 6-way confusion matrix for 397 videos that have been manually classified.

Table I shows the confusion matrix for 397 videos in which the tracks were also manually classified. Although the overall accuracy for classification into the 6 particular classes shown is not great, the leave-one-out classification rate between cricket and non-cricket is 87%. We remark that the automatic classification also identified two video sequences containing (well camouflaged) crickets which had escaped the human observer's attention.

VI. Identifying cricket tags

In order to identify individual crickets, small two-character alphanumeric tags are fixed to the backs of the crickets. Here we briefly outline how the tags may be machine read.

Once a track has been classified as that of a cricket, the tracking ellipses are used as the basis for homing in on the

Algorithm 2 Tag location and identification process

```
**for** some a given video frame **do**
    fit a cricket shape in the region of the tracking ellipse
    **for** each potential ("proposed") tag identification **do**
        try to fit the proposed tag to the extracted frame tag
        calculate the tag probability
    **end for**
    select the proposed tag with the highest probability
**end for**
```

location of the tags and then reading the alphanumeric characters. An overview of the procedure is shown in Algorithm 2. The two steps are described in the following two sections: locating the tag (section VI-A) and identifying it (section VI-B).

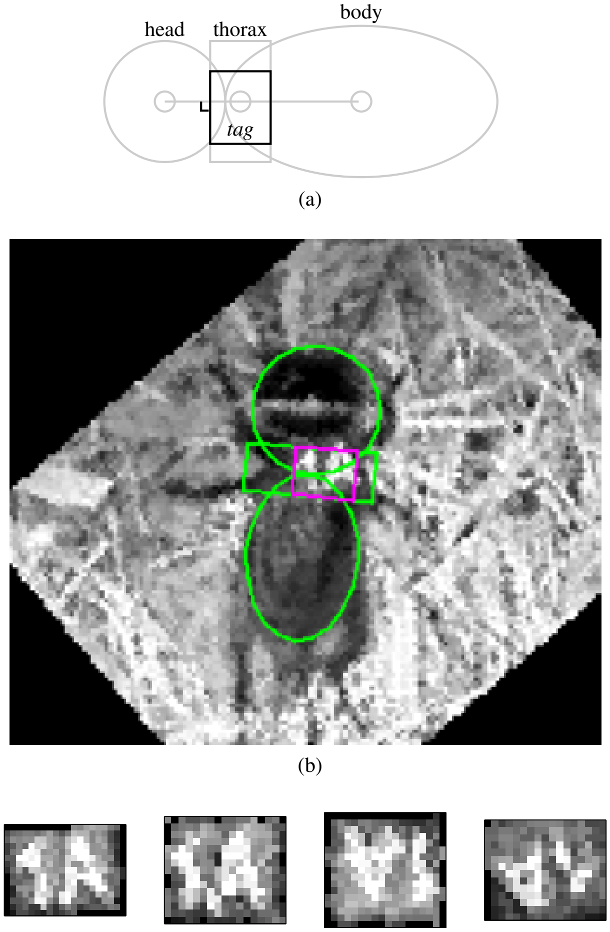
A. Tag location

For a selected frame, an evolutionary algorithm (EA) fits a cricket shape (see figure 5a) to the region centred on the tracking ellipse. Those pixels that fall within the cricket shape, but outside the tag, are assumed to be sampled from one probability distribution (with a low, i.e. dark, mean and small variance); all others are assumed to be sampled from a different probability distribution, with a higher (lighter) mean and wider variance. The fitness function for the EA is the probability of the image given the size and location of the shape, based on these pixel probability distributions. Figure 5b shows a shape fitted to an example frame, with the location of the tag highlighted. Once located, the tag region can be extracted and passed on to be identified. The tag images are often of low quality, as shown in figure 5c.

B. Tag identification

The set of alphanumeric codes used on tags for a particular year are known, so there is a relatively small set of possibilities (approximately 200 in 2008) for a given video. Of course the tag is not always face-on to the camera, so another EA is used to learn the affine transformation associated with the tag's orientation. For each of the codes known to have been used in the year the video was recorded, the EA compares the transformed extracted tag with the actual tags by superimposing one on the other and calculating a match total probability, based on Gaussian distributions for each pixel, with mean value from the actual tag and a fixed variance. This is facilitated by resizing the originally extracted tag to increase the number of pixels it contains. Figure 6 shows an example of a tag extracted from a video and the resized tag overlaid with the match calculated by the EA.

The low resolution of the tag means that the process does not always identify the correct code. Table II shows some example tags and top three most likely identifications.



(a)

(b)

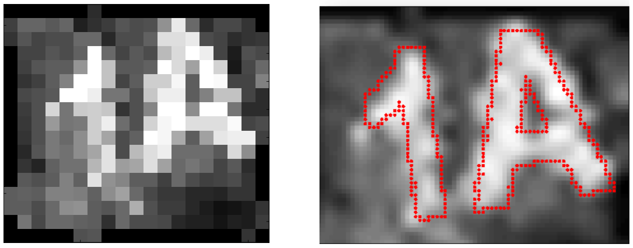
(c)

Fig. 5: (a) The model of the cricket's shape, showing the cricket in grey and the tag as a black rectangle. The shape of the cricket is fixed (the sizes of the parts may vary), but the tag region can be moved left or right within the thorax. A cricket shape is defined by a 10-dimensional vector. The cricket's appendages are ignored. (b) An example of the shape fitted to a cricket. (c) Examples of tags extracted from different frames of the same video, showing the poor resolution.

VII. Conclusion

Although the sequential Bayesian estimation paradigm is very powerful for tracking, it is often analytically intractable and some form of approximation must be used. Sampling methods are often impractical when large volumes of data must be treated and in this paper we have examined the use of variational approximations to the posterior distributions. Although the variational methods may severely underestimate the state uncertainty, we have shown that if the variational updates made in the correct sequence very good approximations to the track may be made. Indeed, because the approximation to the whole track converges for all times together, this method is able to robustly acquire the tracked object, when standard sequential algorithms fail.

The variational approximation makes the tracking of camouflaged animals in a large number of video sequences computationally possible, and we have described a model for effective tracking. We have also described a novel method



(a) original tag

(b) identified tag

Fig. 6: An example tag and its identification.

Tag	Rank 1	Rank 2	Rank 3
	1A	7A	2B
	1A	7A	2A
	2A	2B	7A

TABLE II: Example tags and the three most likely matches for each of them, ranked in order of probability. Each tag was compared with the following possibilities: TX, 1A, 2B, 8C, ZZ, 1B, 7A, 2A.

of identifying the tracked animal from features of the track, and shown how particular individuals may be identified from alphanumeric tags.

As the available spatial and temporal resolution increases, we look forward to increasingly accurate quantitative analysis of populations of wild animals.

References

[1] R. Rodríguez-Muñoz, A. Bretman, J. Slate, C. A. Walling, and T. Tregenza, "Natural and sexual selection in a wild insect population," *Science*, vol. 328, no. 5983, pp. 1269–1272, 2010.

[2] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, 2002.

[3] A. Doucet and A. Johansen, "A tutorial on particle filtering and smoothing: fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovskii, Eds. Oxford University Press, 2011, pp. 656–704.

[4] R. E. Turner and M. Sahani, "Two problems with variational expectation maximisation for time-series models," in *Inference and Learning in Dynamic Models*, D. Barber, A. T. Cemgil, and S. Chiappa, Eds. Cambridge University Press, 2011.

[5] D. Mackay, "Ensemble learning and evidence maximisation," Cavendish Laboratory, University of Cambridge, Tech. Rep., 1995.

[6] ——, "Ensemble learning for hidden Markov models," Cavendish Laboratory, University of Cambridge, Tech. Rep., 1997.

[7] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, vol. 37, no. 2, p. 183, 1999.

[8] H. Attias, "A variational Bayesian framework for graphical models," *Advances in Neural Information Processing Systems*, vol. 12, pp. 209–215, 2000.

[9] H. Lappalainen and J. Miskin, *Advances in Independent Component Analysis*. Berlin: Springer-Verlag, 2000, ch. Ensemble Learning, pp. 75–92.

[10] M. Beal and Z. Ghahramani, "The variational Bayesian EM algorithm for incomplete data: with application to scoring graphical model structures," in *Bayesian Statistics*, vol. 7. Oxford University Press, 2002.

[11] M. Beal, "Variational algorithms for approximate Bayesian inference," Ph.D. dissertation, University College London, 2003.

[12] Z. Ghahramani and M. Beal, "Propagation algorithms for variational Bayesian learning," in *Advances in Neural Information Processing Systems*, vol. 13. MIT Press, 2001, pp. 507–513.

[13] J. Vermaak, N. Lawrence, and P. Pérez, "Variational inference for visual tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003, pp. 773–780.

[14] J. T. Christmas and R. M. Everson, "Variational bayesian smoothing: structured approximations and whole-track convergence," University of Exeter, Tech. Rep., 2013.

[15] J. Christmas, "Robust spatio-temporal latent variable models," Ph.D. dissertation, University of Exeter, 2011. [Online]. Available: http://hdl.handle.net/10036/3051

[16] J. Christmas and R. Everson, "Robust autoregression: Student-t innovations using variational Bayes," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 48–57, Jan 2011.

[17] ——, "Temporally coupled principal component analysis: A probabilistic autoregression method," in *International Joint Conference on Neural Networks*, Barcelona, July 2010.

[18] ——, "Robust probabilistic PCA with autoregression: a variational Bayesian method with Student-t noise," *Submitted to Pattern Recognition*, 2012.