Variational Mixture Smoothing for Non-Linear Dynamical Systems

Cristian Sminchisescu **Allan Jepson**
University of Toronto, Department of Computer Science
Artificial Intelligence Laboratory, 6 King's College Road, Toronto, Canada, M5S 3G4
{*crismin,jepson*}*@cs.toronto.edu, www.cs.toronto.edu/*{˜*crismin,*˜*jepson*}

Abstract

We present an algorithm for computing joint state, smoothed, *density estimates for* non-linear dynamical systems *in a Bayesian setting. Many visual tracking problems can be formulated as probabilistic inference over time series, but we are not aware of* mixture smoothers *that would apply to weakly identifiable models, where multimodality is persistent rather than transient (*e.g. *monocular 3D human tracking). Such processes, in principle, exclude iterated Kalman smoothers, whereas flexible MCMC methods or sample based particle smoothers encounter computational difficulties: accurately locating an exponential number of probable joint state modes representing high-dimensional trajectories, rapidly mixing between those or resampling probable configurations missed during filtering. In this paper we present an alternative, layered, mixture density smoothing algorithm that exploits the accuracy of efficient optimization within a Bayesian approximation framework. The distribution is progressively refined by combining polynomial time search over the embedded network of temporal observation likelihood peaks, MAP continuous trajectory estimates, and Bayesian variational adjustment of the resulting joint mixture approximation. Our results demonstrate the effectiveness of the method on the problem of inferring multiple plausible 3D human motion trajectories from monocular video.*

Keywords: *non-linear non-Gaussian systems, variational approximation, mixture models, high-dimensional search, constrained optimization, monocular 3D body tracking, Kinematic Jump Sampling.*

1 Introduction

Many visual tracking problems can be naturally formulated as probabilistic inference over the hidden states of a dynamical system. In this framework, we work with time series of system state vectors linked by probabilistic dynamic transition rules, and for each state we also have observations and define an observation model. The parameter space consists of the joint state vectors at all times. This trajectory through states is probabilistically constrained both by dynamics and by the observation model. To the extent that these are realistic statistical models, Bayes-law propagation of a probability density for the true state is possible. **Filtering** computes the optimal distribution of states conditioned only by the past whereas **smoothing** finds the optimal state estimate at each time given both past and future observations and dynamics. For non-linear dynamics and observation models under Gaussian noise, this can be computed using iterated Kalman filtering and smoothing. However, for many tracking problems involving clutter and complex models this methodology is not applicable. For general multimodal distributions under non-Gaussian dynamics and observation models, direct MCMC methods or particle filters/smoothers [10, 13, 15, 12, 19, 16] result. These algorithms naturally represent uncertainty, but are less efficient for weakly identifiable high-dimensional models where multimodality is persistent rather than transient.[1] In such cases, there is, theoretically, an exponential number of trajectories for any single observation sequence, and many of these may be probable. Accurately locating them, or sampling new trajectories through temporal states missed during filtering is a major computational challenge. An additional difficulty is that none of the methods give multiple MAP estimates or a similar compact multimodal approximation. This may be useful in its own right for many applications (*e.g*. visualization, high-level analysis and recognition) where the mean state or other expectation calculations may be uninformative or, at least, not the only desired output.

We are not aware of prior work that computes a mixture approximation for smoothing general non-linear non-Gaussian dynamical systems. In this paper, we propose such an algorithm that exploits the accuracy of efficient discrete and continuous optimization within a variational approximation setting. Our method estimates a compact mixture distribution over joint temporal states, and can be efficiently used in tandem with mixture filtering methods [2, 23, 24, 28]. To sidestep the difficulties associated with random high-dimensional initialization, the algorithm cascades several layers of computation. We use dynamic programming, sparse robust non-linear optimization, and variational adjustment, in order to progressively refine a Kullback-Leibler approximation to the true joint state posterior, given an entire observation sequence. The resulting density model is compact and principled, allowing accurate sampling of alternative trajectories, as well as general Bayesian expectation calculations. We finally demonstrate the algorithm on the difficult prob-

[1] Consider *e.g.* 3D monocular human tracking where for known link (body segment) lengths, the strict non-observabilities reduce to twofold 'forwards/backwards flipping' ambiguities for each link, at each timestep [24].

lem of inferring smooth trajectories that reconstruct different plausible 3D human motions in complex monocular video.

Many existing tracking or smoothing solutions attempt to limit multimodality using learned dynamical models [1, 11, 5, 20, 21]. While these may stabilize the estimates, the distracting likelihood peaks are only down-weighted, but rarely disappear. The state space volume, and therefore the theoretical search complexity remains unchanged, and the generality of the tracker may be significantly reduced. In an upcoming paper [22], we propose a non-linearly embedded density propagation algorithm that restricts visual inference to low-dimensional manifolds learned from motion data that is typical of the context where the model will be used. It is likely, however, that under realistic imaging conditions, at least a mild degree of multimodality will still persist during visual inference, for any interesting (*i.e*. sufficiently flexible) motion model or low-dimensional state representation. The algorithm we propose remains useful in such contexts. Another application is the accurate reconstruction of general 3D biological motion for computer graphics or animation.

2 Related Research

There is a large literature on *non-linear* filtering and smoothing, using both Kalman [9] and Monte-Carlo methods [13, 15, 16]. Kalman filtering and smoothing [9] is not applicable for non-Gaussian systems. Particle smoothers [13, 15] are based on forward filtering, followed by smoothing that reweights existing particles, in order to better reflect future evidence. Introduced mostly to tackle the erroneous mean state estimates under transient multimodality [13], the algorithms may not scale well under strong multimodality, where a large number of trajectories have high probability, or when probable temporal states have been missed or eliminated prematurely during filtering. Direct full sequence MCMC methods [16] iteratively generate particle smoother style proposals in their transition kernel. This makes the sampling of new states possible at the price of more expensive step updates, but the methods do not provide an explicit multi-modal representation and fast mixing is difficult. A more compact and efficient approximation that retains modeling generality would be useful.

Bayesian variational methods are one possible class of solutions [14, 8, 18] that typically construct approximations that decouple some of the dependencies present in the original model (*e.g*. mean field). The switching state space model [8, 18] is designed only for piece-wise linear, Gaussian dynamical systems. In general, the variational methods have to sidestep suboptimal modeling and high-dimensional initialization, both of which are problematic. The algorithm we propose is also formulated in a variational setting. Here we use a fully coupled mixture approximation, initialized based on layered, time efficient processing: dynamic programming, polynomial time trajectory search over the network of temporal observation likelihood peaks (initialized from filtering), and local continuous MAP trajectory refinement. Our final approximation is a mixture of Gaussians, and it can be also used to improve mixing in MCMC simulations [25].

Several multiple hypothesis methods exist for filtering 3D human motion [5, 20, 23, 21, 24] but less attention has been given to similar methods for smoothing. Most of the proposed methods compute a single point estimate. Howe *et al* [11] use a dynamical prior obtained from motion capture data and assume 2D joint tracks over an entire time series to compute a 3D joint position MAP estimate. Brand [1] similarly learns a HMM representation from motion capture data and estimates a MAP trajectory, based on time series of human silhouette inputs. DiFranco *et al* [6] propose an interactive system based on a batch optimizer of a Gaussian observation model consisting of 2D human joint correspondences and 3D given human pose key-frames that help disambiguate multimodality resulting from monocular reflective limb ambiguities. This is essentially an iterated Kalman smoother with a better second order step update.

3 Formulation

Consider a non-linear, non-Gaussian, dynamical system having temporal state $\mathbf{x}_t$, $t = 1...T$, prior $p(\mathbf{x}_1)$, observation model $p(\mathbf{r}_t|\mathbf{x}_t)$ with observations $\mathbf{r}_t$, and dynamics $p(\mathbf{x}_t|\mathbf{x}_{t-1})$. Let $\mathbf{X}_t = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t)$ be the model joint state estimated over a time series $1...t$, $t \leq T$, based on observations encoded as $\mathbf{R}_t = (\mathbf{r}_1, \ldots, \mathbf{r}_t)$. The joint probability of all observations and hidden states can be factored (assume for now $\mathbf{X} = \mathbf{X}_T$ and $\mathbf{R} = \mathbf{R}_T$ for notational simplicity) as:

$$\mathcal{P}(\mathbf{X}, \mathbf{R}) = p(\mathbf{x}_1) \prod_{t=2}^{T} p(\mathbf{x}_t|\mathbf{x}_{t-1}) \prod_{t=1}^{T} p(\mathbf{r}_t|\mathbf{x}_t) \qquad (1)$$

For smoothing and other sequence calculations, we are however interested in $\mathcal{P}(\mathbf{X}|\mathbf{R}) = \mathcal{P}(\mathbf{X}, \mathbf{R})/\mathcal{P}(\mathbf{R})$. For multimodal temporal observation likelihood distributions $p(\mathbf{r}_t|\mathbf{x}_t)$, the joint $\mathcal{P}(\mathbf{X}, \mathbf{R})$ may contain an exponential number of modes. We seek a tractable approximating distribution $q^{\boldsymbol{\theta}}(\mathbf{X})$, parameterized by $\boldsymbol{\theta}$, that minimizes the relative entropy: $D(q^{\boldsymbol{\theta}}||\mathcal{P}) = \int_{\mathbf{X}} q^{\boldsymbol{\theta}}(\mathbf{X}) \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}|\mathbf{R})}$.

We further consider the variational free energy $\mathcal{F}(q^{\boldsymbol{\theta}}, \mathcal{P})$, which is a simple modification to $D(q^{\boldsymbol{\theta}}||\mathcal{P})$ that does not change its minimum structure:

$$\mathcal{F}(q^{\boldsymbol{\theta}}, \mathcal{P}) = D(q^{\boldsymbol{\theta}}||\mathcal{P}) - \log \mathcal{P}(\mathbf{R}) \qquad (2)$$

$$= \int_{\mathbf{X}} q^{\boldsymbol{\theta}}(\mathbf{X}) \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}|\mathbf{R})} - \int_{\mathbf{X}} q^{\boldsymbol{\theta}}(\mathbf{X}) \log \mathcal{P}(\mathbf{R}) \qquad (3)$$

$$= \int_{\mathbf{X}} q^{\boldsymbol{\theta}}(\mathbf{X}) \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})} \qquad (4)$$

In the above $\log \mathcal{P}(\mathbf{R})$ does not depend on $\boldsymbol{\theta}$ and does therefore not count for the optimization. Minimizing the varia-

tional free energy w.r.t. $\boldsymbol{\theta}$ is equivalent to refining the approximation $q^{\boldsymbol{\theta}}(\mathbf{X})$ to $\mathcal{P}(\mathbf{X}|\mathbf{R})$. The effectiveness of this procedure depends on a good design and initialization of $q^{\boldsymbol{\theta}}(\mathbf{X})$. We use a fully coupled mixture approximation $q^{\boldsymbol{\theta}}(\mathbf{X}) = \sum_i q_i^{\boldsymbol{\theta}}(\mathbf{X})$, with Gaussian components $q_i^{\boldsymbol{\theta}}(\mathbf{X})$. Methods to initialize its parameters are given in §4 and §5.

4 Multiple Embedded Trajectory Optima using Dynamic Programming

Let the temporal observation likelihood density be approximated by mixtures: $p(\mathbf{r}_t|\mathbf{x}_t) = \sum_{i=1}^{N_t} \pi_t^i m_t^i(\mathbf{x}_t, \boldsymbol{\mu}_t^i, \boldsymbol{\Sigma}_t^i)$, $t = 1...T$, where m_t^i are observation likelihood modes (Gaussian or heavy tail distributions), π_t^i are mixing proportions, N_t are the number of modes at time t. In practice, this representation can be efficiently computed in tandem with a filtering method. For continuously optimized filters like [2, 23, 24], a mixture for $p(\mathbf{r}_t|\mathbf{x}_t)$ is estimated anyway, as a necessary substep during the computation of $p(\mathbf{x}_t|\mathbf{R}_t)$ (we will use KJS [24] for the work here). For discrete particle filters [13, 3, 28] this may involve local optimization on samples from $p(\mathbf{x}_t|\mathbf{R}_{t-1})$ or on the centers of its fitted mixture [28]. Regard the observation likelihood modes as nodes of an *embedded network* that approximates $\mathcal{P}(\mathbf{X}, \mathbf{R})$. Each m_t^i is a node having value equal to its observation likelihood $p(\mathbf{r}_t|m_t^i)$. It connects with all the components j in the previous and next timesteps through links that are the dynamic probabilities $p(m_t^i|m_{t-1}^j)$ and $p(m_{t+1}^j|m_t^i)$. The values for the inter-mode dynamics and mode observation likelihood can be obtained by integrating the point-wise dynamics and observation likelihood over the support of each mode: $p(m_{t+1}^j|m_t^i) = \int_{\mathbf{x}_{t+1}} \int_{\mathbf{x}_t} m_{t+1}^j(\mathbf{x}_{t+1}) m_t^i(\mathbf{x}_t) p(\mathbf{x}_{t+1}|\mathbf{x}_t)$ and $p(\mathbf{r}_t|m_t^i) = \int_{\mathbf{x}_t} m_t^i(\mathbf{x}_t) p(\mathbf{r}_t|\mathbf{x}_t)$, and $p(m_1^i) = \int_{\mathbf{x}_1} m_1^i(\mathbf{x}_1) p(\mathbf{x}_1)$. Given the parametric temporal mixture representation, this operation can be performed efficiently either by sampling or by using analytic approximations for particular functional forms of m_t^i or p (*e.g.* a Bhattachayya distance for Gaussians).[2] Notice that the mixture weights π_t^i are not necessary for the construction of the embedded network.

Each embedded trajectory is a sample from $\mathcal{P}(\mathbf{X}, \mathbf{R})$ but we seek a reduced set that is representative of $\mathcal{P}$ among exponentially many possible paths. We select a tractable intuitive approximation and consider the $N_1 N_T$ most probable trajectories between any possible pairs of observation likelihood modes in the initial and final timesteps. To compute these efficiently with dynamic programming (DP), we exploit the embedded network sparsity (each mixture set at t only connects with the previous ones at timestep $t-1$ and with the next ones at timestep $t+1$) and use Johnson's algorithm [4]. This applies multiple Dijkstra computations at each node in the network to compute single source most probable paths (each of these forms a tree rooted at the source node [4]). The process has complexity $\mathcal{O}(V^2 \log V + VE)$ for a network with V vertexes and E edges. For our problem this can be further reduced since only the most probable paths between nodes of the initial and final timesteps are desired. Given an upper bound $M \geq N_i, \forall i$ and T timesteps, the complexity of this computation is $\mathcal{O}(M^3(T-1) + M^2 T \log MT)$ for a Fibonacci heap implementation.

[2] The embedded network is not a HMM. In a HMM, the number and the set of possible values for the states is the same, temporally. Here, the number of modes at each timestep depends on the uncertainty of the observation likelihood, and their corresponding means can take different continuous values.

5 Continuous MAP Refinement

Trajectories obtained with DP are globally optimal only w.r.t. a *fixed network* of observation likelihood peaks (obtained using filtering or some importance proposal distribution). True optimal smoothing can be obtained by re-estimating the joint hidden state $\mathbf{X}$ based on the full observation sequence $\mathbf{R}$. Because the model is non-linear and non-Gaussian, we have to follow a general approach and directly optimize $\mathcal{P}(\mathbf{X}, \mathbf{R})$. The DP solutions provide good quality, fast initialization to an otherwise difficult high-dimensional search problem. Based on these, trajectories are refined to obtain optimal modes (MAP) using efficient sparse second order continuous methods [7, 26]. (While the DP results are also aposteriori maxima w.r.t. the embedded network, for brevity we use the names DP and MAP to differentiate between the discrete and the continuous optimization results) An ascent direction is chosen by solving the regularized subproblem:

$$(\mathbf{H} + \lambda \mathbf{W})\, \delta\mathbf{X} = \mathbf{g} \quad \text{subject to} \quad C_{bl} \cdot \mathbf{X} < 0 \qquad (5)$$

with $\mathbf{g} = \frac{d\mathcal{P}}{\mathbf{dX}}$, $\mathbf{H} = \frac{d^2\mathcal{P}}{\mathbf{dX}^2}$, $\mathbf{W}$ is a symmetric positive definite damping matrix and λ is a dynamically chosen weighting factor. C_{bl} are hard rectangular prior state constraints (*e.g.* joint angle limits, replicated at all time-steps). For our problem, the joint state Hessian $\mathbf{H}$ is block tridiagonal. The observations couple to the current time state, and fill the diagonal blocks $\mathbf{H}_t = \frac{d^2 p(\mathbf{r}_t|\mathbf{x}_t)}{\mathbf{dx}_t^2}$, whereas the first-order Markovian dynamics couples to the previous and next states, and fills the off-diagonal blocks $\mathbf{H}_{t,t-1} = \frac{d^2 p(\mathbf{x}_t|\mathbf{x}_{t-1})}{\mathbf{dx}_t^2}$. The lower and upper triangular factors are also block tridiagonal, the inverse is however dense. [3]

To efficiently compute the state update, the Hessian is decomposed by recursive steps of reduction, where blocks of variables are progressively eliminated by partial factorization. At each linearization point, the forwards reduction gives filtering. It progressively computes, for each timestep, the opti-

[3] For kinematic modeling, the Hessian has, in general, a secondary tridiagonal structure embedded in each block. This is produced by the simply coupled kinematic chains of the limbs, where each link couples with the previous and next ones. Body parts at the root of the hierarchy, *e.g.* the torso, however induce denser couplings.

Variational Mixture Smoothing Algorithm

Input: Temporal set of mixture approximations for $p(\mathbf{r}_t|\mathbf{x}_t) = \sum_{i=1}^{N_t} \pi_t^i m_t^i(\mathbf{x}_t, \boldsymbol{\mu}_t^i, \boldsymbol{\Sigma}_t^i), t = 1...T$.
Output: Joint mixture approximation of $\mathcal{P}(\mathbf{X}|\mathbf{R}) = \sum_{i=1}^{N_1 N_T} q_i^{\boldsymbol{\theta}}(\mathbf{X})$.

1.(§4) Build the embedded network G
For $t = 1...T-1, i = 1...N_t, j = 1...N_{t+1}$:
— $w_t^{ij} \leftarrow p(m_{t+1}^j|m_t^i)p(\mathbf{r}_{t+1}|m_{t+1}^j)$.
— If $(t = 1)$ $w_1^{ij} \leftarrow w_1^{ij} p(\mathbf{r}_1|m_1^i)p(m_1^i)$

G has nodes m_t^i and weights w_z^{ij} with $t = 1...T, i = 1...N_t, j = 1...N_t, z = 1...T-1$. A weighted path between any two nodes is the product of all intermediate weights.

2. (§4) Compute most probable weighted paths $\mathbf{X}_k^{dp}, k = 1...N_1 N_T$ between modes $m_1^i, i = 1...N_1$ and $m_T^j, j = 1...N_T$ in G.

3. (§5) Estimate local MAP modes and covariances ($\xi_k = \mathbf{X}_k^{map}, \boldsymbol{\Lambda}_k = (\mathbf{H}_k^{map})^{-1}$), $k = 1...N_1 N_T$, using DP initialization $\mathbf{X}_k^{dp}, k = 1...N_1 N_T$ (without loss of generality assume no duplicate local optima are found). The size of $\xi_k, \boldsymbol{\Lambda}_k$ is $(T \cdot n)$ and $(T \cdot n)^2$, where $n = \dim(\mathbf{x})$.

4. (§6) Initialize the variational mixture approximation $\sum_{i=1}^{N_1 N_T} q_i^{\boldsymbol{\theta}^0}(\mathbf{X})$ using $\boldsymbol{\theta}^0 = (\boldsymbol{\theta}_1^0, \ldots, \boldsymbol{\theta}_{N_1 N_t}^0)$ with $\boldsymbol{\theta}_k^0 = (\rho_k, \xi_k, \boldsymbol{\Lambda}_k)$, $k = 1...N_1 N_T$, or alternatively use best B modes for computational efficiency. The mixing proportions for components are computed as $\rho_k = \mathcal{P}(\mathbf{X}_k^{map}, \mathbf{R}) / \sum_{i=1}^{N_1 N_T} \mathcal{P}(\mathbf{X}_i^{map}, \mathbf{R})$.

5. (§6) Optimize variational bound $\mathcal{F}$ (6) by updating variational parameters $\boldsymbol{\theta}$ using (9).

Figure 1: The steps of our Variational Mixture Smoothing Algorithm for high-dimensional multimodal distributions.

mal current state estimate given all previous observations and dynamics. The corresponding recursion by back-substitution gives smoothing [9, 27]. Filtering is the first half-iteration of a general nonlinear optimizer. For non-linear dynamics and non-Gaussian observation models, the local MAP state trajectory is estimated by successive passes of filtering, smoothing and relinearization of $\mathcal{P}(\mathbf{X}, \mathbf{R})$.

6 Variational Updates for Mixtures

Given, MAP modes of $\mathcal{P}(\mathbf{X}, \mathbf{R})$ (computed in §5) having mixing proportions, means and covariances unfolded in parameter vector $\boldsymbol{\theta}_i = (\rho_i, \xi_i = \mathbf{X}_i^{map}, \boldsymbol{\Lambda}_i = (\mathbf{H}_i^{map})^{-1})$, $i = 1...N_1 N_T$, we construct an approximating mixture distribution with augmented parameter space $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{N_1 N_T})$, $q^{\boldsymbol{\theta}} = \sum_i q_i^{\boldsymbol{\theta}}$, with $q_i^{\boldsymbol{\theta}} \sim \rho_i \mathcal{N}(\mathbf{X}, \xi_i, \boldsymbol{\Lambda}_i)$. (Each component $q_i^{\boldsymbol{\theta}}$ indexes in the global state $\boldsymbol{\theta}$ for its parameters $\boldsymbol{\theta}_i$) The variational free energy is:

$$\mathcal{F}(q^{\boldsymbol{\theta}}, \mathcal{P}) = \int_{\mathbf{X}} q^{\boldsymbol{\theta}}(\mathbf{X}) \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})} \quad (6)$$

$$= \sum_i \int_{\mathbf{X}} q_i^{\boldsymbol{\theta}}(\mathbf{X}) \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})} \quad (7)$$

$$= \sum_i < \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})} >_{q_i^{\boldsymbol{\theta}}} \quad (8)$$

The mixture parameters can be optimized by computing the gradient of the variational free energy:

$$\frac{d\mathcal{F}}{\mathbf{d}\boldsymbol{\theta}} = \sum_i \int_{\mathbf{X}} q_i^{\boldsymbol{\theta}}(\mathbf{X}) g_i^{\boldsymbol{\theta}}(\mathbf{X}) \left(1 + \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})}\right) \quad (9)$$

$$= \sum_i < g_i^{\boldsymbol{\theta}}(\mathbf{X}) \left(1 + \log \frac{q^{\boldsymbol{\theta}}(\mathbf{X})}{\mathcal{P}(\mathbf{X}, \mathbf{R})}\right) >_{q_i^{\boldsymbol{\theta}}} \quad (10)$$

$$g_i^{\boldsymbol{\theta}}(\mathbf{X}) = \frac{d \log q_i^{\boldsymbol{\theta}}(\mathbf{X})}{\mathbf{d}\boldsymbol{\theta}} \quad (11)$$

where $g_i^{\boldsymbol{\theta}}(\mathbf{X})$ are the gradients of the individual mixture component Gaussian quadratic forms w.r.t. parameter subsets $\boldsymbol{\theta}_i = (\rho_i, \xi_i, \boldsymbol{\Lambda}_i)$ of $\boldsymbol{\theta}$. The mixture has to obey two internal constraints: *(1)* on the mixture coefficients $\sum_{i=1}^{N_1 N_T} \rho_i = 1$; and *(2)* on the positive definiteness of component covariance matrix $\boldsymbol{\Lambda}_i, i = 1...N_1 N_T$. These can be easily enforced by reparameterization using softmax for the mixing proportions: $\alpha_k = \exp(\rho_k) / \sum_i \exp(\rho_i)$, and Cholesky decomposition for the covariance matrices: $\boldsymbol{\Lambda}_i^{-1} = \mathbf{L}^\top \mathbf{L}$ where $\mathbf{L}$ is upper triangular with elements l_{ij}, the diagonal terms are positive, *e.g.* $l_{ii} = \exp(d_i)$ and $|\boldsymbol{\Lambda}_i|^{-1/2} = \prod_i l_{ii}$. The newly introduced variables α_i, d_i and l_{ij} $(j > i)$ are now unconstrained real numbers. The smoothing algorithm is summarized in fig. 1.

7 Experiments

We show experiments that involve estimating a joint mixture distribution for smoothing 3D articulated human motion in monocular video (fig. 9).

The human model consist of 32 d.o.f. kinematic skeleton controlled by angular joint state variables, covered by 'flesh' built from superquadric ellipsoids with global deformations. The state space has priors controlling joint angle limits, and body part non self-intersection constraints, included as additional terms in the likelihood [23]. **The Observation Likelihood** is based on a robust combination of intensity-based alignment metrics, silhouette, and normalized edge distances

[23]. **Filtering** is based on Kinematic Jump Sampling (KJS) [24], which is a density propagation method involving locally optimized covariance based random sampling [23, 24] with a domain-specific deterministic sampler based on skeletal reconstruction using inverse kinematics.

The experiments we show are based on the analysis of a 2.5s, 120 frame sequence of *monocular* video involving agile complex motion of a human subject in a cluttered scene (see fig. 9). KJS filtering uses up to 8 modes per timestep. Mixtures for $p(\mathbf{r}_t|\mathbf{x}_t)$, here containing up to 8 modes, are estimated during the computation of the filtered $p(\mathbf{x}_t|\mathbf{R}_t)$ [2, 23]. The flow of processing is the one in fig. 1: observation likelihood modes are assembled in an embedded network where all most probable trajectories between pairs of modes in the initial and final timestep are computed using dynamic programming. These are refined non-linearly to obtain trajectory modes of $\mathcal{P}(\mathbf{X}, \mathbf{R})$. The modes and covariances (inverse Hessians at maxima) are used to initialize a variational mixture approximation of $\mathcal{P}(\mathbf{X}|\mathbf{R})$ that is refined based on the updates in (9). Data analysis for these steps is described next.

The embedded network structure (node values and internode edges, with the observation likelihood modes being nodes) is estimated, as explained in §4, based on a subsampled time series having T=47 steps. We compute 64 most probable paths corresponding to trajectories between all possible pairs of nodes at times 1 and T. In fig. 2, we show, for each node, the probability that it is visited by the different probable paths. The nodes at all times are unfolded on the x axis (temporal modes are sequentially assigned a unique number). The probabilities are all positive, but we flip sign at the beginning of each new timestep for visualization. Each node m_t^i can be visited by generally N_1 possible trajectories (here $N_{\{t=1\}} = 8, N_{\{T=47\}} = 8$), each initiated at a different starting mode m_1^j (the highest probable paths routed at m_1^j form a tree). However, the 'visiting probabilities' for a mode could be negligible, *e.g.* because it has low likelihood or very low dynamic transition probabilities w.r.t. probable modes at times $t, (t+1)$. Let the corresponding path probabilities to the mode be p_{tj}^i. We compute the probability that m_t^i is visited by some trajectory initiated at m_1^j as: $\sum_j p_{tj}^i / \sum_i \sum_j p_{tj}^i$, plotted in fig. 2. The trajectory distribution is highly multimodal. Occasionally, there are 'bottlenecks' at timesteps where the observation likelihood mixture collapses to fewer components, producing spikes up or down in fig. 2. This also leads to fewer and more probable trajectories, *e.g.* for modes indexed $100 - 150$. Some of these correspond to timesteps where the tracked subject has both arms in front of his face. Many reflective ambiguities of the arms become improbable, due to the presence of physical body non-penetration priors. This is one situation where the physical priors, although locally much broader than the observation likelihood, are more constraining.

Fig. 3 and 4 compare joint angle trajectories for the DP and

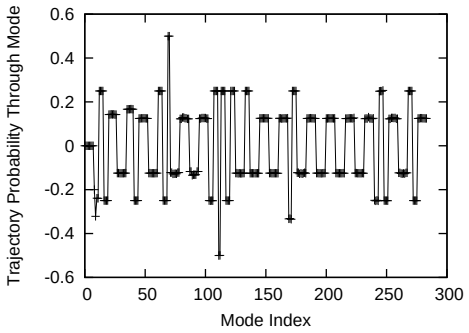

Figure 2: There are *many different* probable trajectories through the embedded network. We show the trajectory probability through modes at all times, unfolded on the x axis (see text). These values are positive, but we flip sign in-between consecutive timesteps for visualization. The temporal trajectory distribution collapses to fewer components in regions where the uncertainty of the observation likelihood diminishes (to fewer modes), but is generally multimodal.

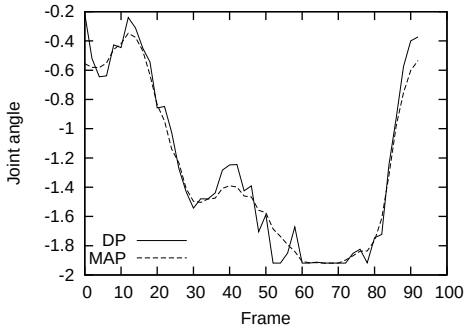

Figure 3: MAP smooths trajectories but correctly preserves prior constraints (joint angle limits and body non self-intersection), *e.g.* frames 60-70.

MAP solutions. MAP significantly improves smoothness and preserves joint angle limits, see *e.g.* fig. 3, frames 60-70. Sequence smoothing can sometimes lead to qualitatively different solutions at particular timesteps w.r.t. the DP results, *e.g.* there is a large change in the state variables in-between the frames 50-60 in fig. 4.

We also compute the average joint state distance between the MAP and the DP solutions for all 64 trajectories in fig. 5. The averaging is done over a radian + meter state space (the distance between trajectory vectors is averaged by the number of timesteps and the number of variables). The difference per state variable is about 2-3 degrees, but many changes are concentrated in only a few temporal states as shown in fig. 3,4. This explains why the DP and MAP trajectories are often qualitatively different.

Fig. 6 shows the MAP trajectory energy *only* (without dynamics), *i.e.* negative log-likelihood product over temporal states in (1). The measurement error is low, only about 4%

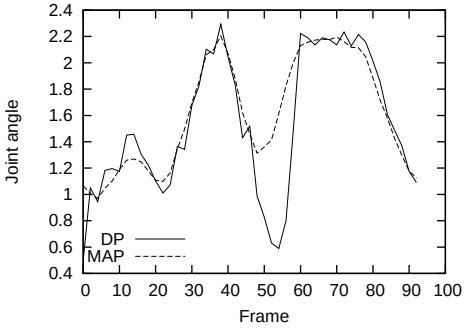

Figure 4: MAP fills-in states missed during filtering, *e.g.* in frames 50-60 configurations are significantly far w.r.t. the DP solution, *i.e.* a different temporal state optimum.

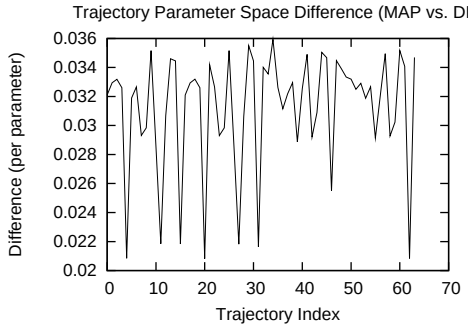

Figure 5: DP and MAP solutions can be qualitatively different. The average change per joint is about 2-3 degrees, but many times the changes are concentrated in only a few temporal state variables, *e.g.* see fig. 3, 4.

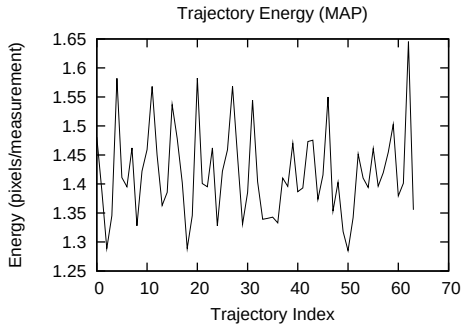

Figure 6: Although MAP smooths trajectories and can perturb them w.r.t. the DP solution, it preserves a low negative joint observation likelihood, the rightmost product in (1).

larger than the one of a filtered fit. This shows that MAP not only smooths the trajectories, but also preserves good image likelihood.

In fig. 7 we show computations of the Hessian matrix eigenspectrum at a local MAP trajectory, for state spaces of increasing dimension. Joint states for the 32 d.o.f. human model are estimated over 1, 8, 47 timesteps (having 32, 256 and 1504 variables respectively). The ratio of largest/smallest singular values decreases from 5616 showing severe ill-conditioning for 1 frame to 2637 for 8 frames and 737 for 47 frames. The advantage of additional constraints provided by longer sequences dominates over the inconvenience of a larger state space. The overall effect is the decrease in estimation uncertainty.

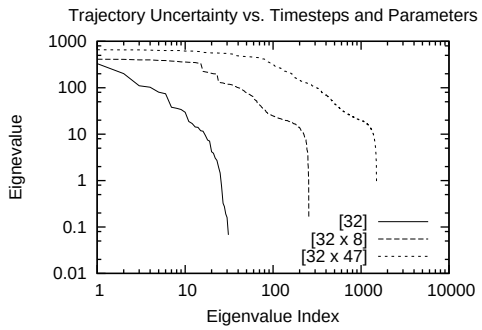

Figure 7: Continuous optimization over long sequences reduces the local MAP uncertainty. Notice the change in the uncertainty structure at a local maximum, for a 32 d.o.f. model optimized over 1, 8 and 47 frames (with joint state having 32, 32x8, 32x47 variables). The largest/smallest eigenvalue ratio decreases from 5616 through 2637 to 737. The state is larger, but the longer sequences are better constrained.

The final smoothing step involves initializing the variational mixture approximation based on the set of MAP modes found using continuous optimization. Fig. 8 shows the decrease in variational free energy over 15 iterations, but a plateau is already reached after about 6-8 refinement steps. For this sequence, we assumed the covariance structure is fixed to the one obtained from MAP (which is however *different* for each mode) and estimate the mixing proportions, component means and one separate positive inflation factor that uniformly rescales the covariances $\mathbf{\Lambda}_k$, for each mode. The inflation factor accommodates broader component spread along directions where several trajectories with reasonable high probability cross. This may correspond to cases where the corresponding state variable couplings are close to a low-lying saddle point between two close state space reflective ambiguities at that particular timestep. Other situations include sharply peaked priors that when composed with an ill-conditioned observation likelihood may lead to biased MAP modes and covariances that underestimate the surrounding volume. This is typical of many ill-posed problems in vision.[4] For the algorithm we present, all mixture parameters

[4] In fact, the MAP estimates do not play the central role in Bayesian inference. They change arbitrarily with reparameterization, and they optimize the density without taking into account the complementary volume information. On the other hand, a KL approximation may occasionally fail to precisely account for narrow peaks. Depending on efficiency constraints, or the expected utility of the density model, one can also follow the steps §4,§5 or §4,§6, or optimize different parameter subsets in §6 (*e.g.* keep the means fixed).

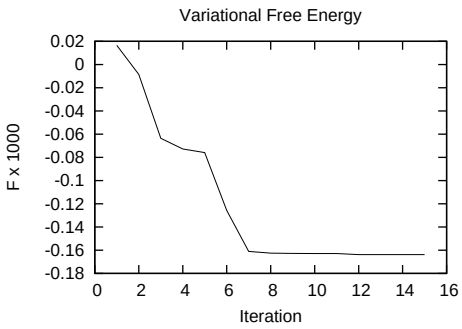

Figure 8: Optimizing the variational bound increases the data likelihood $\mathcal{P}(\mathbf{R})$ and the quality of the approximation $q^{\boldsymbol{\theta}}$ to $\mathcal{P}(\mathbf{X}|\mathbf{R})$. Not much improvement is achieved after 10 iterations.

can vary and, most importantly, the variational free energy provides a cost function for their optimal adjustment. Different subsets of parameters can be selected and optimized based on application constraints.

Finally, fig. 10 shows a couple of trajectories sampled from the smoothed distribution. Although in the beginning of the sequence the two trajectories look qualitatively similar, they diverge significantly during its second half. Noticeable differences are the different tilt of the torso and especially the left arm positioning that follows a trajectory corresponding to a reflective ambiguity w.r.t. the camera. However, both solutions look plausible and have high observation likelihood.

8 Conclusions

We have presented a mixture smoother for non-linear dynamical systems. We are not aware of any prior algorithm that would compute a mixture approximation for smoothing such systems. The one we propose applies to weakly identifiable models, where multimodality is persistent rather than transient. Such models are typical of many visual inference applications like 3D monocular human modeling and tracking, or scene reconstruction using structure-from-motion. Strong multimodality and non-Gaussianity rules out the use of iterated Kalman smoothers, whereas direct MCMC methods or particle-based smoothers may encounter difficulties in accurately locating multiple probable high-dimensional state trajectories or rapidly mixing between them. Our algorithm refines a compact approximation by combining polynomial time search over the network of observation likelihood peaks, local MAP continuous trajectory estimates and Bayesian variational adjustment of the resulting joint mixture representation. We show results that demonstrate the method on the estimation of multiple, smooth, high-quality trajectories that represent plausible articulated 3D human motions in difficult monocular video.

Future Work will explore the use of multiresolution solvers for large dynamic programming problems, as well as tractable mixture approximations that automatically decouple some of the state variables. It would be interesting to study the impact of learned motion models on the trajectory distribution, or derive reconstruction algorithms robust to missing data [22].

Acknowledgments We give special thanks to Kyros Kutulakos and Nigel Morris for kind help with the video capture.

References

[1] M. Brand. Shadow Puppetry. In *ICCV*, pages 1237–44, 1999.

[2] T. Cham and J. Rehg. A Multiple Hypothesis Approach to Figure Tracking. In *CVPR*, volume 2, pages 239–245, 1999.

[3] K. Choo and D. Fleet. People Tracking Using Hybrid Monte Carlo Filtering. In *ICCV*, 2001.

[4] T. Cormen, C. Leiserson, and R. Rivest. *An Introduction to Algorithms*. MIT Press, 1996.

[5] J. Deutscher, A. Blake, and I. Reid. Articulated Body Motion Capture by Annealed Particle Filtering. In *CVPR*, 2000.

[6] D. DiFranco, T. Cham, and J. Rehg. Reconstruction of 3-D Figure Motion from 2-D Correspondences. In *CVPR*, 2001.

[7] R. Fletcher. Practical Methods of Optimization. In *John Wiley*, 1987.

[8] Z. Gahramani and G. Hinton. Variational learning for switching state space models. *Neural Computation*, 2001.

[9] A. Gelb, editor. *Applied Optimal Estimation*. MIT Press, 1974.

[10] N. Gordon, D. Salmond, and A. Smith. Novel Approach to Non-linear/Non-Gaussian State Estimation. *IEE Proc. F*, 1993.

[11] N. Howe, M. Leventon, and W. Freeman. Bayesian Reconstruction of 3D Human Motion from Single-Camera Video. *NIPS*, 1999.

[12] M. Isard and A. Blake. A Smoothing Filter for CONDENSATION. In *ECCV*, 1998.

[13] M. Isard and A. Blake. CONDENSATION – Conditional Density Propagation for Visual Tracking. *IJCV*, 1998.

[14] T. Jaakkola and M. Jordan. Improving the Mean Field Approximation via the use of Mixture Distributions. *Learning in Graphical Models*, 1998.

[15] G. Kitagawa. Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. *J. Comput. Graph. Statist.*, 1996.

[16] R. Neal, M. Beal, and S. Roweis. Inferring State Sequences for Non-Linear Systems with Embedded Hidden Markov Models. In *NIPS*, 2003.

[17] R. Neal and G. Hinton. A View of EM that justifies Incremental, Sparse, and other Variants. In *Learning in Graphical Models*, 1998.

[18] V. Pavlovic, J. Rehg, T. Cham, and K. Murphy. A Dynamic Bayesian Approach to Figure Tracking using Learned Dynamical Models. In *ICCV*, 2001.

[19] S. Scott. Bayesian Methods for Hidden Markov Models. Recursive Computing in the 21st Century. *J. Amer. Stat. Association*, 97, 2002.

[20] H. Sidenbladh, M. Black, and D. Fleet. Stochastic Tracking of 3D Human Figures Using 2D Image Motion. In *ECCV*, 2000.

[21] H. Sidenbladh, M. Black, and L. Sigal. Implicit Probabilistic Models of Human Motion for Synthesis and Tracking. In *ECCV*, 2002.

Figure 9: Tracking a 2.5s video sequence containing agile motion in clutter. *First row:* original Sequence, *Second row:* one probable model state sequence projected onto image at selected time-steps. See fig. 10 for alternative 3D trajectories.

Figure 10: Variational smoothing generates multiple plausible trajectories. 3D reconstruction results based on a 2.5s video sequence shown in fig. 9. *First row:* One smoothed reconstructed 3D trajectory viewed from a synthetic viewpoint. *Second row:* alternative 3D trajectory. Although in the beginning the two trajectories appear qualitatively similar, they diverge significantly during the second half of the sequence. Note the different tilt of the torso and the significant difference in the left arm positioning that followed a smooth trajectory corresponding to a reflective ambiguity w.r.t. the camera.

[22] C. Sminchisescu and A. Jepson. Generative Modeling for Continuous Non-Linearly Embedded Visual Inference. In *ICML*, Banff, 2004.

[23] C. Sminchisescu and B. Triggs. Estimating Articulated Human Motion with Covariance Scaled Sampling. *IJRR*, 22(6):371–393, 2003.

[24] C. Sminchisescu and B. Triggs. Kinematic Jump Processes for Monocular 3D Human Tracking. In *CVPR*, volume 1, pages 69–76, Madison, 2003.

[25] C. Sminchisescu, M. Welling, and G. Hinton. A Mode-Hopping MCMC Sampler. Technical Report CSRG-478, University of Toronto, submitted to *Machine Learning Journal*, September 2003.

[26] P. Toint and D. Tuyttens. On large-scale Nonlinear Network Optimization. *Mathematical Programming*, 1990.

[27] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle Adjustment - A Modern Synthesis. In Springer-Verlag, editor, *Vision Algorithms: Theory and Practice*, 2000.

[28] J. Vermaak, A. Doucet, and P. Perez. Maintaining Multi-odality through Mixture Tracking. In *ICCV*, 2003.