Pattern Recognition 44 (2011) 2738–2748

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Generalized darting Monte Carlo

Cristian Sminchisescu [a,b,*], Max Welling [c]

[a] Institute for Mathematics (IMAR), 21 Calea Grivitei Street, 010702 Bucharest, Romania
[b] University of Bonn, Germany
[c] University of California, Irvine, United States

ARTICLE INFO

Article history:
Received 22 September 2008
Received in revised form
18 November 2010
Accepted 5 February 2011
Available online 17 February 2011

Keywords:
Markov chain Monte Carlo
Markov random fields
Darting
Constrained optimization
3D reconstruction
Human tracking

ABSTRACT

One of the main shortcomings of Markov chain Monte Carlo samplers is their inability to mix between modes of the target distribution. In this paper we show that advance knowledge of the location of these modes can be incorporated into the MCMC sampler by introducing mode-hopping moves that satisfy detailed balance. The proposed sampling algorithm explores local mode structure through local MCMC moves (*e.g.* diffusion or Hybrid Monte Carlo) but in addition also represents the relative strengths of the different modes correctly using a set of global moves. This 'mode-hopping' MCMC sampler can be viewed as a generalization of the darting method [1]. We illustrate the method on learning Markov random fields and evaluate it against the spherical darting algorithm on a 'real world' vision application of inferring 3D human body pose distributions from 2D image information.

1. Introduction

It is well known that MCMC samplers have difficulty in mixing from one mode to the other because it typically takes many steps of very low probability to make the trip [2,3]. Recent improvements designed to combat random walk behavior, like Hybrid Monte Carlo and over-relaxation [4,2] do not solve this problem when modes are separated by high energy barriers. In this paper we show how to exploit knowledge of the location of the modes to design a MCMC sampler that mixes properly between them.

We consider two possible scenarios where this advance knowledge is present. In one example we have actively searched for high probability regions using sophisticated optimization methods [5,6]. Given these local maxima, we now desire to collect unbiased samples from the underlying probability distribution. In another example we are given data-cases and aim at learning a model distribution to represent these data as accurately as possible. In this case, the data itself is representative of the low energy mode of a well fitted model.

This paper is organized as follows. In Section 2 we review some popular Markov chain Monte Carlo methods. Then, in Section 3 we introduce the new mode-hopping sampler and some extensions. An additional proof of detailed balance and an auxiliary variable formulation of the method appear in the Appendix. Section 4 explains and illustrates an application to learning Markov random fields, while in Section 5 the generalized darting method is evaluated against the spherical darting method on a 'real world' vision application – learning human models and estimating 3D human body poses from 2D image information.

2. Markov chain Monte Carlo sampling

Imagine we are given a probability distribution $p(\mathbf{x})$ with $\mathbf{x} \in \mathcal{X} \subset R^d$ a vector of continuous random variables. In the following we will focus on continuous variables, but the algorithm is easily extended to discrete state spaces. A very general method to sample from this distribution is provided by Markov chain Monte Carlo (MCMC) sampling. The idea is to start with an initial distribution $p_0(\mathbf{x})$ and design a set of transition probabilities that will eventually converge to the target distribution $p(\mathbf{x})$.

The most commonly known transition scheme is the one proposed in the Metroplis–Hastings (M–H) algorithm, where a target point is sampled from a possibly asymmetric conditional distribution $Q(\mathbf{x}_{t+1}|\mathbf{x}_t)$, where $\mathbf{x}_t$ represents the current sample. To make sure that detailed balance holds, *i.e.* $p(\mathbf{x}_t)Q(\mathbf{x}_{t+1}|\mathbf{x}_t) = p(\mathbf{x}_{t+1})Q(\mathbf{x}_t|\mathbf{x}_{t+1})$, which in turn guarantees that the target distribution remains invariant under Q, we should only accept a certain fraction of the proposed targets:

$$P_{accept} = \min\left[1, \frac{p(\mathbf{x}_{t+1})Q(\mathbf{x}_t|\mathbf{x}_{t+1})}{p(\mathbf{x}_t)Q(\mathbf{x}_{t+1}|\mathbf{x}_t)}\right] \tag{1}$$

[*] Corresponding author.
E-mail addresses: Cristian.Sminchisescu@ins.uni-bonn.de (C. Sminchisescu), Welling@ics.uci.edu (M. Welling).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.02.006

In the most commonly used M–H algorithm, the transition distribution Q is symmetric and independent of the energy surface at location $\mathbf{x}$. This simplifies (1) (the Q factors cancel), but leads to slow mixing due to random walk behavior. It is however not hard to incorporate local gradient information, $dE(\mathbf{x})/d\mathbf{x}$, to improve mixing speed. One could for instance bias the proposal distribution $Q(\mathbf{x}_{t+1}|\mathbf{x}_t)$ in the direction of the negative gradient $-dE(\mathbf{x})/d\mathbf{x}$ and accept using (1):

$$\mathbf{x}_{\tau+1} = \mathbf{x}_\tau - \frac{\Delta\tau^2}{2}\left.\frac{dE(\mathbf{x})}{d\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_\tau} + \Delta\tau\mathbf{n} \tag{2}$$

where $\mathbf{n}$ is a vector of independently chosen Gaussian variables with zero mean and unit variance, and $\Delta\tau$ is the stepsize. When the stepsize becomes infinitesimally small this is called the Langevin method and one can show that the rejection rate vanishes in this limit.[1]

The Langevin method is a special case of a more general sampling technique called Hybrid Monte Carlo (HMC) sampling [4,2,7]. In HMC the particle is given a random initial momentum sampled from a unit-variance isotropic Gaussian density and its deterministic trajectory along the energy surface is then simulated for T time steps using Hamiltonian dynamics. If this simulation has no numerical errors the increase, ΔE, in the combined potential and kinetic energy will be zero. If ΔE is positive, the particle is returned to its initial position with a probability of $1-\exp(-\Delta E)$. Numerical errors up to second order are eliminated by using a 'leapfrog' method which uses the potential energy gradient at time τ to compute the velocity increment between time $\tau-\frac{1}{2}$ and $\tau+\frac{1}{2}$ and uses the velocity at time $\tau+\frac{1}{2}$ to compute the position increment between time τ and $\tau+1$. The Langevin method corresponds to precisely one step of HMC (*i.e.* $T=1$).

A host of clever MCMC samplers can be found in the literature. We refer to the excellent review [2] for more information.

3. The mode-hopping MCMC algorithm

We start with reviewing the closely related darting algorithm described in [1]. In darting-MCMC we place spherical jump regions of equal volume at the location of the modes of the target distribution. The algorithm is based on a simple local MCMC sampler which is interrupted with a certain probability to check if its current location is inside one of these spheres. If so, we initiate a jump to the corresponding location in another sphere, chosen uniformly at random, where the usual Metropolis acceptance rule applies. To maintain detailed balance we decide not to move if we are located outside any of the balls. It is not hard to check that this algorithm maintains detailed balance between any two points in sampling space.

In high-dimensional spaces this procedure may still lead to unacceptably high rejection rates because the modes will likely decay sharply in at least a few directions. Since these ridges of probability are likely to be uncorrelated across the modes, the proposed target location of the jump will have very low probability, resulting in almost certain rejection. In the following we will propose two important improvements over the darting method. Firstly, we allow the jump regions to have arbitrary shapes and volumes and secondly these regions may overlap. The first extension opens the possibility to align the jump regions precisely with the shape of the high probability regions of the target distribution. The second extension simplifies the design and placement of the jump regions since we don't have to worry about possible overlaps of the chosen regions.

First consider the case when the regions are non-overlapping but of different volumes. Like in the darting method we could consider a one-to-one mapping between points in the different regions, or we could choose to sample the target point uniformly inside the new region. Because the latter is somewhat simpler conceptually, we will use uniform sampling in this section. The deterministic case will be treated in the next section. Also, to simplify the discussion we will first consider the case where the underlying target distribution is uniform, *i.e.* has equal probability everywhere. Due to the difference in volumes, particles are more likely to be inside a large region than in small ones. Thus, there will be a larger flow of particles going from the bigger regions towards the smaller ones violating detailed balance. To correct for it we could reject a fraction of the proposed jumps from larger towards smaller regions. There is however a smarter solution that picks the target region proportional to its volume:

$$P_i = \frac{V_i}{\sum_j V_j} \tag{3}$$

If we view the jumps between the various regions as a (separate) Markov chain, this method samples directly from the equilibrium distribution while a rejection method would require a certain mixing time to reach equilibrium. Clearly, if the underlying distribution is not uniform, we need the Metropolis acceptance rule between the jump point and its image in the target region:

$$P_{accept} = \min\left[1, \frac{p(\mathbf{t})}{p(\mathbf{x})}\right] \tag{4}$$

where $\mathbf{t}$ is the target point and $\mathbf{x}$ is the exit point.

Now, let us see what happens if two regions happen to overlap. Again, we first consider sampling the target point uniformly in the new region, and consider a target distribution which is uniform. Consider two regions which partly overlap. Due to the fact that we use the probability P_i (3), each volume element $\mathbf{dx}$ inside the regions has equal probability of being chosen. However, points located in the intersection will be a target twice as often as points outside the intersection. To compensate, *i.e.* to maintain detailed balance, we need to reject half of the proposed jumps into the intersection. In general, we check the number of regions that contain the exit point, $n(\mathbf{x})$, and similarly for the target point, $n(\mathbf{t})$. The appropriate fraction of moves that is to be accepted in order to maintain detailed balance is $\min[1, n(\mathbf{x})/n(\mathbf{t})]$. Combining this with the Metropolis acceptance probability (4) we find

$$P_{accept} = \min\left[1, \frac{n(\mathbf{x})p(\mathbf{t})}{n(\mathbf{t})p(\mathbf{x})}\right] \tag{5}$$

Putting everything together, we define the mode-hopping MCMC sampler explained in Fig. 1.

3.1. Elliptical regions with deterministic moves

In the previous section we have uniformly sampled the proposed new location of the particle inside the target region. This is a very flexible method for which it is easy to prove detailed balance. However, a deterministic transformation can be tuned to map between points of roughly equal probability which is expected to improve the acceptance rate. Consider for instance the case that the energy surfaces near the regions is exactly quadratic and have the same height (*i.e.* their centers have equal probability). We can now define a transformation between ellipses that maps between points of equal probability resulting in a vanishing rejection rate. This is obviously not the case when we use uniform sampling.

[1] One can use more general biased proposal distributions, but the one defined in (2) was chosen because of its vanishing rejection rate in the limit $\Delta\rightarrow 0$.

```
Generalized Darting MCMC Sampler

Repeat until convergence

1. Draw a sample $u_1 \sim U$ [0, 1].

2. if $u_1 > P_{check}$:
   perform one step of a local MCMC sampler.

3. if $u_1 < P_{check}$
   (a) Identify the number of regions $n$ (x) that
       contain the current sample.
   (b) if $n$ (x) = 0
       do nothing.
   (c) if $n$ (x) > 0
       i. Sample a new region according to $P_i$
          (3).
       ii. Propose a location inside the new
           region (either deterministically or
           uniformly at random).
       iii. Identify the number of regions $n$ (t)
            that contain the proposed sample.
       iv. Draw a sample $u_2 \sim U$ [0, 1].
       v. if $u_2 > P_{accept}$ (5)
          reject move.
       vi. if $u_2 < P_{accept}$ (5)
           accept move
```

Fig. 1. The steps of our generalized darting sampler.

We first consider the case of non-overlapping elliptical regions. Ellipses seem a natural choice, but the algorithm presented here is by no means restricted to it. For instance, the method is readily generalized to the use of rectangles as basic shapes. We will parameterize an ellipse by a mean $\boldsymbol{\mu}$, a covariance $\boldsymbol{\Sigma}$ and a scale α, *i.e.* the ellipse is defined to be the equiprobability contour that is α standard deviations away from the mean. We will also need the eigenvalue decomposition of the covariance, $\boldsymbol{\Sigma} = \mathbf{USU}^{\top}$, where $\mathbf{S}$ is a diagonal matrix containing the eigenvalues denoted by $\{\sigma_i\}$. A deterministic transformation between two ellipses 1→2 is given by

$$\mathbf{x}_2 = \boldsymbol{\mu}_2 - \mathbf{U}_2\mathbf{S}_2^{1/2}\mathbf{S}_1^{-1/2}\mathbf{U}_1^{\top}(\mathbf{x}_1 - \boldsymbol{\mu}_1) \tag{6}$$

We note that this transformation would not leave a point invariant if we chose the second ellipse to be equal the first one, but mirrors it in the origin. Even though the transformation above is one-to-one, it does change the volume element $\mathbf{dx}$, implying that we need to take the Jacobian of the transformation into consideration. The intuitive reason for this is the same as in the previous section: more particles will be located in the larger ellipses resulting in more jumps to smaller ellipses than back, violating detailed balance. To compensate we sample the target ellipse again proportional to its volume, *i.e.* using (3), where

$$V_{ellipse} = \frac{\pi^{d/2}\alpha^d \prod_{i=1}^{d} \sigma_i}{\Gamma(1+d/2)} \tag{7}$$

where $\Gamma(x)$ is the gamma-function with $\Gamma(x+1) = x\Gamma(x)$, $\Gamma(1) = 1$ and $\Gamma(\frac{1}{2}) = \sqrt{\pi}$.

We will now discuss how this algorithm can be generalized in case the ellipses overlap. Consider again two ellipses which partly overlap and a uniform target density. Consider a point that is located inside both ellipses, *i.e.* in the overlap (point 1). To apply the deterministic mapping, we first need to choose one of the two ellipses as a basis for the transformation. Unfortunately, an arbitrary rule such as the ellipse on top of the stack, or the one with the largest volume will result in a violation of detailed balance. Thus, we propose to pick the ellipse at random with equal probability. Now consider the image point under the mapping (point 2), choosing either the same ellipse (resulting in mirroring the point at the origin) or choosing the other ellipse. Assume point 2 is not located in the overlap. The probability of moving from 1→2 is $\frac{1}{4}$; a factor $\frac{1}{2}$ coming from the fact that we first choose with equal probability which ellipse will be used to define the transformation, and another factor $\frac{1}{2}$ because we sample the target ellipse using (3). However, in the other direction 2→1 the probability is $\frac{1}{2}$. Note that unlike the case of uniformly sampling a target point (see previous section) the probability of going from 2→1 is not doubled.[2] Thus, to rescue detailed balance we need to accept only half of the proposed moves from 2→1, or more generally $\min[1, n(\mathbf{x})/n(\mathbf{t})]$ with $n(\cdot)$ the number of ellipses containing a point. Combining this with the usual Metropolis acceptance rule applicable to general target densities, we arrive precisely at the rule in (5).

To summarize, the deterministic algorithm has precisely the same structure as algorithm in Fig. 1, where in the transformation (6) ellipse 1 is chosen uniformly at random from all ellipses containing point 1 and ellipse 2 is chosen using (3) with V_i given by (7).

3.2. Mode-hopping in discrete state spaces

Many practical problems are best described by probability distributions with discrete state spaces. It is therefore of importance to discuss this case as well. Fortunately, the extension is rather straightforward, the main difference being that 'volumes' are to be replaced by 'number of states within a certain distance'.

In this section we consider the Manhattan distance, but the algorithm is by no means restricted to that choice. Consider a discrete state s in some D-dimensional space, where every dimension can take one of V values, e.g. $\mathbf{s} = [0,3,6,1]$ for $D=4$ and $V=6$. The Manhattan distance between two states $\mathbf{s}_1$ and $\mathbf{s}_2$ is the total number of changes we need to make to transform one state into the other, or:

$$\mathcal{D}(\mathbf{s}_1, \mathbf{s}_2) = \sum_{i=1}^{D} |\mathbf{s}_1^i - \mathbf{s}_2^i| \tag{8}$$

First consider the situation where no points are contained in two distinct regions and the regions have the same shape. Again, we have a choice of using a deterministic transformation, mapping states one-to-one to each other. For instance, if regions are defined to be the collection of all states that are at most a distance $\mathbf{d}$ away from a reference state $\mathbf{r}$, than we can use the offset $\mathbf{s} - \mathbf{r}$ to define the mapping: $\mathbf{s}_2 \rightarrow \mathbf{r}_2 + (\mathbf{s}_1 - \mathbf{r}_1)$. Since all regions have the same number of states, we can simply pick a target region uniformly at random.

The situation is slightly more complicated if we allow for regions with different numbers of states. It is clear that one-to-one mappings are now no longer possible. If one insists on a deterministic mapping many-to-one mappings are possible, but intricate acceptance rules will need to be designed to retain detailed balance. We will therefore proceed by using a random method, where the state in the target region is picked uniformly at random from all possible states in that region. In analogy with

[2] The reason is that for every target ellipse the image of the point under the mapping (6) is different. However, there are circumstances, e.g. when one ellipse is completely encircled by a larger one, that isolated points have the same image for two distinct target ellipses, resulting in violation of detailed balance. Since in the continuous case this set has measure zero, we will ignore it.

the continuous case, in order to maintain detailed balance, we need to pick the target region according to the distribution:

$$P_i = \kappa_i \Big/ \sum_j \kappa_j. \tag{9}$$

where κ_i is the number of states contained in region i.

It is also easy to generalize this to overlapping regions. The same reasoning as in Section 3 leads to the conclusion that a fraction of samples $\min[1, n(\mathbf{x})/n(\mathbf{t})]$ should be accepted where $n(\cdot)$ is the number of regions that contains a point. Finally, combining this with general target densities leads to the acceptance rule (5). The resulting MCMC algorithm is now very similar to the one in Fig. 1, but with a different distance measure and a probability of picking a target region given by (9).

3.3. A further generalization

In the previous sections we have used distance measures to define regions between which the samples could 'jump'. This is geometrically appealing, but unnecessary for the algorithm to function properly. More generally, we can use a set of conditions that must be satisfied in order to be able to jump between these generalized regions. In order to maintain detailed balance we should however be able to determine the total number of states which satisfy each set of conditions. The probability (9) can then be used to pick a target region and the acceptance rule (5) can be used to accept or reject a randomly picked point from that region. Overlaps are also allowed in this case.

4. Learning random fields

The proposed mode-hopping algorithm can only be successful if we have advance information[3] about the expected location of regions of high probability. In the following two sections we discuss examples where this is indeed the case.

In the first example we consider a situation where we want to train a random field (RF) model from data. The general form of a RF is given by

$$p(\mathbf{x}|\mathbf{O},\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta},\mathbf{O})} e^{-E(\mathbf{x};\mathbf{O},\boldsymbol{\theta})} \tag{10}$$

where $\boldsymbol{\theta}$ is a set of parameters that we try to infer given the data, $\mathbf{O} = \{\mathbf{o}_1, \ldots, \mathbf{o}_n\}$ are image observations, $E(\mathbf{x};\mathbf{o},\boldsymbol{\theta})$ is the energy and $Z(\boldsymbol{\theta})$ the normalizing constant or partition function:

$$Z(\boldsymbol{\theta},\mathbf{O}) = \int \mathbf{d}\boldsymbol{\theta}\, e^{-E(\mathbf{x};\mathbf{O},\boldsymbol{\theta})} \tag{11}$$

We use the maximum likelihood criterium to define a cost function for finding the optimal setting of these parameters:

$$\mathcal{F} = -\frac{1}{N}\sum_{n=1}^{N} \log p(\mathbf{x}_n|\mathbf{o}_n,\boldsymbol{\theta}) \tag{12}$$

$$\mathcal{F} = \langle E(\mathbf{x};\mathbf{o}_n,\boldsymbol{\theta})\rangle_{data} + \log Z(\boldsymbol{\theta},\mathbf{O}) \tag{13}$$

To minimize this 'free energy' (negative log-likelihood) we need to compute its gradients:

$$\frac{d\mathcal{F}}{\mathbf{d}\boldsymbol{\theta}} = \left\langle \frac{dE(\mathbf{x};\mathbf{O},\boldsymbol{\theta})}{\mathbf{d}\boldsymbol{\theta}} \right\rangle_{data} - \left\langle \frac{dE(\mathbf{x};\mathbf{O},\boldsymbol{\theta})}{\mathbf{d}\boldsymbol{\theta}} \right\rangle_{model} \tag{14}$$

where the second term is equal to the negative derivative of the log-partition function w.r.t. $\boldsymbol{\theta}$. Note that the only difference between the two terms in (14) is the distribution which is used to average the energy derivative. In the first term we use the empirical distribution, *i.e.* we simply average over the available data-set. In the second term however we average over the model distribution as defined by the current setting of the parameters. Computing this second average analytically is typically too complicated, so approximations are needed instead. An unbiased estimate can be obtained by replacing the integral by a sample average, where the sample is to be drawn from the model $p(\mathbf{x}|\mathbf{O},\boldsymbol{\theta})$. In many cases MCMC is the only method available that can generate this sample set.

Imagine the target distribution $p(\mathbf{x}|\mathbf{O},\boldsymbol{\theta})$ has many modes and the Markov chain is initialized in one of them. Due to the energy barriers, we do not expect the chain to mix very well between the modes which results in very poor estimates of the second term in (14). Under the assumption that the modes of the distribution are located close to clusters of data-points, a viable strategy is to start the Markov chains at various different data-points in order to have some representative samples in each mode. This strategy is used in *contrastive divergence learning* [8] where a Markov chain is initiated at each data-point and run for only a few steps (*i.e.* not to equilibrium) to generate distorted reconstructions of the data-point. Those reconstructions are subsequently used in the second term of (14) to compute an estimate of the energy derivative.[4]

Even though we have arranged to generate samples in most relevant modes of the distribution, the fact that the samples do not properly mix between the modes results in poor estimates of the relative probability masses of the modes under the distribution defined by the *model*. The mode-hopping extension of the MCMC sampler proposed in this paper can help resolve this problem by defining regions around data clusters between which the sampler jumps. In one limit one could imagine defining a small spherical region around every data point, or an appropriate subset of all data points. An alternative possibility is to run a clustering algorithm as a preprocessing step and to define regions corresponding to each cluster. For example, a mixture of Gaussian model could be trained using the Expectation Maximization algorithm with regions corresponding to the equiprobability contours α standard deviations away from the mean. Since these regions are elliptical, the deterministic mode-hopping algorithm described in Section 3.1 may be used.

5. An application to human articulated pose estimation using a single image

We explore the potential of the generalized darting method for *monocular*3D human pose estimation. This problem has applications for human–computer interaction and for actor reconstruction from movie footage—in this case only one camera viewpoint, the one presented in the movie, is usually available.

We run experiments based on correspondences between the articulated joints of a subject in the image and the joints of a 3D articulated model (2D–3D correspondences). We also report experiments for learning the model parameters in a maximum likelihood framework (Section 4), using a more sophisticated edge-based observation model. Monocular human pose estimation is well adapted to illustrate the algorithm because the resulting 3D pose

[3] Adapting the Markov chain to include new regions on-line would violate the Markov assumption and is therefore not guaranteed to converge to the desired probability distribution.

[4] Contrastive divergence was successfully used for learning random fields, under the assumption that training data from the model state distribution is available. The method combines incomplete MCMC simulations initiated at training data points into an efficient, but biased estimate [9]. In contrast, we do not assume the availability of training data (this may not exist in many problems, or may be hard to obtain when the model state is not directly observable), and we pursue an asymptotically unbiased sampling procedure obtained, as typical, within a single MCMC simulation.

posterior is both high-dimensional ($\approx$35 human joint angle state variables) and highly multimodal. In any single monocular image, under point-wise 3D human joints and their image projections, each limb of the human is subject to a 'reflective' kinematic flip ambiguity. Two 3D human body configurations with symmetrical slant in depth w.r.t. the camera (see Fig. 4) produce identical point-wise image perspective projections. The number of possible solutions multiples over the number of links of the human body. For example, a 3D human model with 10 links (torso, head, left/right forearm, upperarm, thigh and calf) may have $2^{\#links}$ local optima, although this is usually an overestimate. Some solutions may not be physically plausible and may violate joint angle limits or body non-self-intersection constraints. The question this work addresses is not how to find the optima but how to efficiently sample from the 3D human pose *equilibrium distribution* once these are known.

Related research: Many powerful approaches to image-based Bayesian inference use non-parametric, sampled-based representations for their target distributions and employ factored sampling schemes [10] for computation. These allow working with non-Gaussian observation models and have guaranteed asymptotic correctness. Practically however, sample-based representations are computationally expensive, a factor often limiting their practical usage to models with 6–8 dimensions. The number of particles required for good accuracy grows exponentially when there are loosely coupled sub-systems, such as the limbs of a body, that each create local minima separated by high energy barriers. To alleviate these problems, partitioned samplers [11] or layered, annealing-based methods have been proposed [5,12]. However these methods tend not to be well adapted for our problem, where there is often not a large margin between the desired global optimum and other competing ones. Instead, we observe several competing local optima with comparable energy, see *e.g.* Figs. 4 and 7. These represent plausible 3D human pose interpretations given our overly simple human body model, the sparse set of image observations we consider and the intrinsic forward–backward depth ambiguities in the pose of 3D articulated chains from monocular images. It is likely that the margin and volume ratio between the correct pose optimum and the incorrect ones may be increased with better modeling and with learning (as we show in one of the experiments). It is likely, however, that at the early stages of learning the 3D human pose posterior will be highly multimodal and have high entropy. Effective ML learning requires efficient methods for sampling *cf.* (14). The proposed generalized darting is one possible way of doing this.

Another approach is to approximate the pose posterior using a mixture model. The energy minima can be located using non-linear continuous optimization and used to build an importance sampler based on a mixture of simple densities (*e.g.* Gaussians). However, the sampler may be deficient in approximating the tails of the modal distributions and may have low correction weights. An alternative is to use gradient-based hybrid MCMC schemes [4,2,13,14]. These can significantly improve acceptance rates for high-dimensional models, but broken ergodicity caused by trapping in local optima usually persists [14].

In this paper we assume that there is considerable prior knowledge about the shape of the energy surface (*e.g.* prior search to locate optima) and focus on using this information to accelerate fair sampling. Several algorithms for locating multiple pose optima can be used. Sminchisescu and Triggs [15] use problem dependent constraints about forward/backward pose ambiguities in order to explicitly enumerate an entire equivalence class of kinematic minima starting at any given one in the class. Heuristic sampling methods based on these have been also studied in [16]. For more general second-order differentiable energy functions without kinematic symmetries [6,14] present methods to systematically find new minima by locating first-order saddle points (transition states) surrounding the basin of attraction of an initially chosen one, found *e.g.* by gradient descent. The search progresses in a local to global manner, in order to eventually locate a sufficiently large and representative minimum set.

In this paper we propose an algorithm that uses knowledge of local optima and gradient-based sampling moves in order to greatly improve both local and non-local acceptance rates—both mixing within a mode and mixing between different modes. The algorithm is not only general but also motivated by the structure of the studied visual inference problem. The modes of the 3D human pose probability distribution are the configurations of a 3D human body model that align well with image features, *e.g.* 2D projected model edges align with the edges of the filmed human. The use of monocular image sequences makes the depth information difficult to infer because depth is lost in perspective projection. The combinations of model variables that produce motion in depth (towards or away from the camera) have the highest uncertainty. The variables that control the motions parallel with the image plane are better constrained by the image evidence. Hence the core of the maxima have highly anisotropic ellipsoidal structure—see Fig. 5a for the spectral covariance structure of a local minimum with ratio of most uncertain/most certain direction $\approx 10^3$. The combinations of variables that lead to motion in depth change with 3D human pose and with camera viewing direction. Hence the different optima do not share the same principal directions, nor the same scaling. In contrast to classical darting, where the sampler uses spherical, mode-centered regions, generalized darting uses oriented ellipsoids computed from the local uncertainty structure of the distribution, for more global and faster sampling estimates with increased acceptance rates.

5.1. Domain modeling

This section describes the humanoid visual models used in our sampling experiments. For more details see [17].

Representation: A typical human body model is constructed from a 'skeleton' that has 30–35 rotational joints controlled by angular joint state variables $\mathbf{x}$, which includes a global 6D translation of the body center. It also has 'body flesh' built from 3-D ellipsoids with deformation parameters $\boldsymbol{\theta}$, here 36 variables for the head, torso, arms and legs. The surface model improves the image representation for the 3D human pose estimates based on image features like edges.[5] In one of the experiments we not only estimate the model state, but also learn its parameters using maximum likelihood, *cf.* Section 4.

Joint positions $\mathbf{u}_i$ in local coordinate systems for each body limb are transformed into points $\mathbf{p}_i(\mathbf{x},\mathbf{u}_i)$ in a global 3D coordinate system, then into predicted image points $\mathbf{r}_i(\mathbf{x},\mathbf{u}_i)$ using composite non-linear transformations $\mathbf{r}_i(\mathbf{x},\mathbf{u}_i)=\mathbf{P}(\mathbf{p}_i(\mathbf{x},\mathbf{u}_i))=\mathbf{P}(\mathbf{K}(\mathbf{x},\mathbf{u}_i))$, where $\mathbf{K}$ represents a chain of rigid transformations that map different body links through the kinematic chain to their global 3D position (see Fig. 2), and $\mathbf{P}$ represents perspective image projection.

For model state estimation, we compute the negative log likelihood of each known image joint position, $\mathbf{o}_i$, under a Gaussian centered at its projected (hypothesized) image location, $\mathbf{r}_i$. The costs are summed over all the human body joints to produce the state-space energy function. This is a function $e(\mathbf{o}_i|\mathbf{x},\boldsymbol{\theta})$ of the prediction error $\Delta\mathbf{o}_i(\mathbf{x})=\mathbf{o}_i-\mathbf{r}_i(\mathbf{x})$ between the model and the given image data. The cost gradient $\mathbf{g}_i(\mathbf{x})$ and

[5] The energy function is a sum of residuals between projected model contours and observed image edges, integrated over all elements on the model occluding contour.

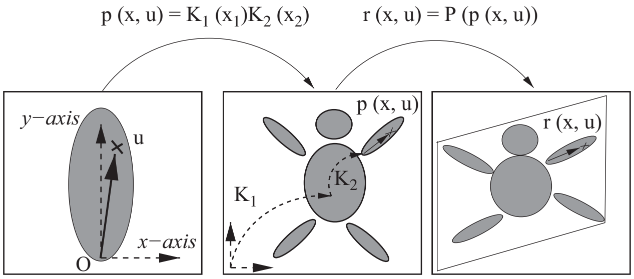

Fig. 2. A simple model of a kinematic chain consisting of ellipsoidal parts. First, the feature $\mathbf{u}_i$, defined in its local coordinate frame, is mapped to a 3D position, $\mathbf{p}_i(\mathbf{x},\mathbf{u}_i)$, in the body model through a chain of transformations, $\mathbf{K}_i(\mathbf{x}_i)$, between local coordinate systems. $\mathbf{x}_i$ are state variables that encode transformations (here rotation angles) between these reference frames, state variables are collectively stored in a vector $\mathbf{x}$. Finally, the 3D surface point given by $\mathbf{p}_i(\mathbf{x},\mathbf{u}_i)$ is mapped into the image using perspective image projection: $\mathbf{r}_i(\mathbf{x},\mathbf{u}_i)=\mathbf{P}(\mathbf{p}_i(\mathbf{x},\mathbf{u}_i))$, where $\mathbf{P}$ is the viewing camera projection matrix (this includes the global orientation of the camera and its intrinsic parameters, *e.g.* focal length, pixel dimensions, *etc.*).

Hessian $\mathbf{H}_i(\mathbf{x})$ are also computed, being used for second continuous optimization and hybrid Monte Carlo (HMC) step calculations.

Energy function: The model state estimates are obtained by optimizing a maximum *a posteriori* criterion, the total posterior probability according to Bayes rule:

$$p(\mathbf{x}|\mathbf{O},\boldsymbol{\theta}) \propto p(\mathbf{O}|\mathbf{x},\boldsymbol{\theta})p(\mathbf{x}) \tag{15}$$

$$p(\mathbf{x}|\mathbf{O},\boldsymbol{\theta}) = \exp\left(-\sum_i e(\mathbf{o}_i,\mathbf{x},\boldsymbol{\theta})\right)p(\mathbf{x}) \tag{16}$$

where $e(\mathbf{o}_i,\mathbf{x})$ is the cost density associated with observation i, the sum is over all observations $\mathbf{O}=\{\mathbf{o}_1,\ldots,\mathbf{o}_n\}$, and $p(\mathbf{x})$ is the model state prior. The energy function is defined as the negative log-likelihood of the posterior:

$$E(\mathbf{x},\mathbf{O},\boldsymbol{\theta}) = -\log p(\mathbf{O}|\mathbf{x},\boldsymbol{\theta}) - \log p(\mathbf{x}) \tag{17}$$

$$E(\mathbf{x},\mathbf{O},\boldsymbol{\theta}) = \sum_i e(\mathbf{o}_i,\mathbf{x},\boldsymbol{\theta}) + E_p(\mathbf{x}) \tag{18}$$

For experiments, we have used the classical Langevin sampler (see Section 2), in combination with long-range jumps using the spherical darting and the generalized darting methods. We also experiment using an independence sampler based on a mixture of normal distributions centered at the local pose optima.

Observation likelihood: We used a simple product of Gaussian likelihoods for each model skeletal joint (assumed independent) with cost $e(\mathbf{o}_i,\mathbf{x},\boldsymbol{\theta}) = \Delta\mathbf{o}_i^2/2\sigma^2$. The negative log-likelihood for the observations is the sum of squared model joint re-projection errors. For our study, it provides an interesting and difficult to handle degree of multimodality owing to the kinematic complexity of the human model and the lack of observability of 1/3 of the model state variables (the depth related ones) in any single monocular image. For learning experiments, we use an observation model based on edge residuals. These are collected at model occluding contours predicted in the image. At each 3D model configuration, for each element on a image-predicted model contour, a line search along the normal direction is used to locate an image edge that matches it. The distance between the location of the model contour and the image edge is used to construct a quadratic function of the residual, similar to the one based on skeletal joint residuals. This is summed over all contour predictions and all the human body parts.

Prior distributions: The priors we use are stabilizers to avoid singular distributions for hard to estimate state variables (*e.g.* in the clavicle and shoulder complex), terms for collision avoidance between body parts, and joint angle limits.[6] Gradients and Hessians for the priors are evaluated and added to the ones of the observation *cf.* (15) and (17). The overall processing pipeline of a full system is given in Fig. 3. Experimental details about each component are given in the experimental section.

5.2. Experiments

Sampling experiments: We have selected four local minima corresponding to the left forearm and left calf in a monocular side view of the body (see Fig. 4). The local minima have relative volumes of (0.16, 0.38, 0.10, 0.36) and energy levels (4.41, 6.31, 7.20, 8.29).

For local optimization we use a second-order damped Newton trust region method [18] where gradient and Hessians of the energy functions are computed analytically and assembled using the chain rule with gradient back-propagation on individual kinematic chains, for efficiency. For generalized darting, we estimate local covariances as inverse Hessians at each local minimum. For MCMC simulations, we enforce joint limit constraints using reflective boundary conditions, *i.e.* by reversing the sign of the normal momentum when it hits a joint limit. We found this gave an improved sampling acceptance rate compared to simply projecting the proposed configuration back on the constraint surface, as the latter leads to cascades of rejected moves until the momentum direction gradually swings around.

We ran the simulation for $\Delta\tau=0.1$, using the Langevin sampler (Fig. 6a), the darting method with spherical covariances (Fig. 6b) and the generalized darting method with deterministic moves (Fig. 6c). In Fig. 6 we show a fragment of a larger simulation that uses a small jump probability $P=0.03$, in order to diminish the frequency of jumps for illustrative purposes. It is easily noticeable that the classical sampler is trapped in the starting mode, and wastes all of its samples exploring it repeatedly. The spherical darting method explores only two minima based on one successful long-range jump during 600 iterations. The darting method (right) explores more minima by combining local moves with non-local jumps that are accepted more frequently. Different minima are visited using seven jumps. This could be visually observed in Fig. 6. After each jump, the sampler equilibrates at a different energy level associated to the new local minimum.

We have also performed a large simulation (10^5 steps) with $\Delta\tau=0.1$ and probability $P=0.25$ for the darting moves. The first 200 samples were discarded in order to let the chain reach equilibrium. The covariance volume scaling factor α was set to unity. For classical darting, we place spheres of unit radius around each minimum. With these parameters, the sampler mixes fast within each minimum, but still has good acceptance rates of 94% for local moves. The acceptance rate for long-range jumps in the spherical case is $a_s = 1292/24{,}863 = 0.052$ whereas for the generalized darting case is $a_g = 9642/25{,}850 = 0.388$, which is an important improvement. According to our tests, the results are stable to changes in the volume factor α by roughly 10%. We have also experimented with an independence sampler based on a mixture of normal distributions, centered at different minima, with covariances estimated as inverse Hessians and mixing proportions given by the relative covariance volumes, as for generalized darting. The overall acceptance ratio is $3492/10^5 = 0.3492$,

[6] The priors are useful in order to downgrade the probability of non-admissible model configurations, particularly for configurations where the arms project inside the body contour, *e.g.* side views (Fig. 4). Without physical priors, 3D structural modeling errors or image correspondence errors could lead to state estimates where body parts penetrate each other.

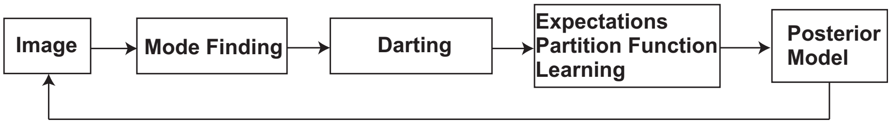

Fig. 3. The pipeline in which the darting method is used for model learning and the calculation of expectations for state estimation. Given an image, we compute the maxima of the current observation model using eigenvector tracking/hypersurface sweeping (any other efficient mode finding methods to locate multiple maxima can be used). These maxima are used in order to initialize a darting-based MCMC calculation which provides samples from the equilibrium distribution. The samples are use both for inference and for model learning (partition function approximations), and the system can operate in a loop for video processing and tracking. See illustrations for both types of calculations in the experimental section.

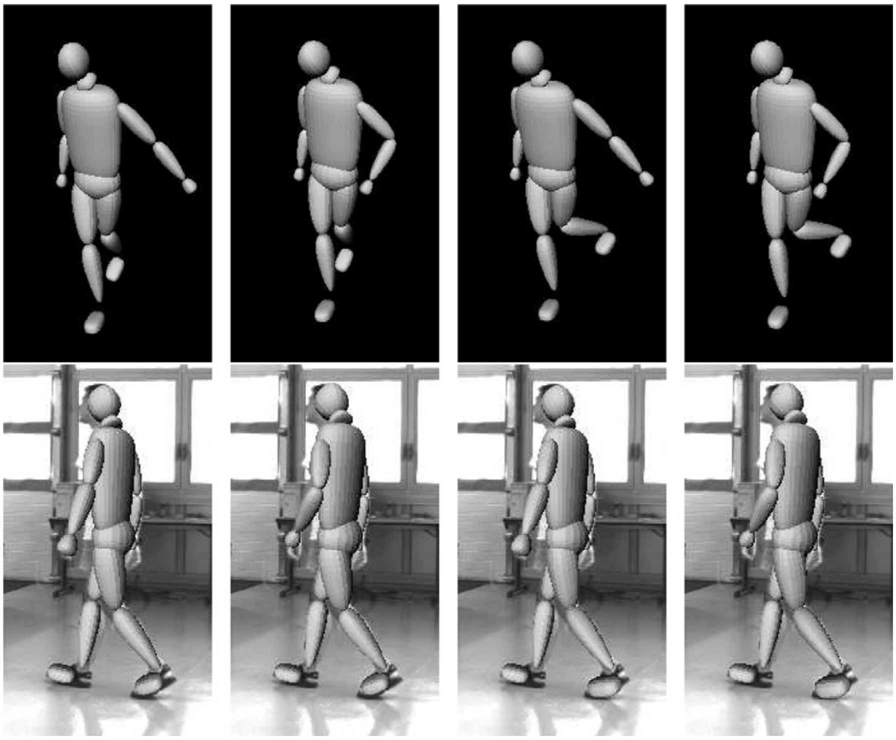

Fig. 4. Human pose estimation based on a single image of a walking person photographed sideways. In any single monocular image, under point-wise 3D human joints and their image projections, each limb of the human is subject to a ‘reflective’ kinematic flip ambiguity. Two 3D human body configurations with symmetrical slant in depth w.r.t. the camera produce identical point-wise image perspective projections. The bottom row shows four copies of the same image, with the projection of four different poses of the model superimposed on the image. The four poses are shown from a different viewpoint in the top row. The different poses correspond to four local minima in the energy function given by (17), defined over 35 state variables (the human joint angles). Notice how the four human body configurations indeed overlap and align well with the human subject in the image.

Table 1
Comparative results of different algorithms for models with different state dimensions.

Dimension	Gen. dart.	Sph. dart.	Indep. sampler
4	0.94	0.85	0.62
12	0.88	0.75	0.56
35	0.8	0.7	0.34

which is comparable to the acceptance ratio of a generalized darter for long-range jumps. However, notice that the overall number of accepted moves (including both global inter-minimum moves and local ones) for generalized and spherical darting is about 0.8 and 0.7 respectively, both significantly higher than the independence sampler. Clearly, different samplers offer different computational trade-offs. Here we aim for a balance between good mixing and reasonably low rejection rates. Results on different body models having 4, 12 and 35 state dimensions respectively (corresponding to the left arm, the left body side and the full-body) are given in Table 1. The acceptance rates are consistent among different methods, with a modest decrease, as dimensionality increases.

Ergodicity study: We study the performance of a generalized darter and of a hybrid MCMC sampler in a different experiment based on three runs of 20,000 simulation steps each. We compute the ergodic measure [1], an indicator for the rate of self-averaging in equilibrium calculations. Although self-averaging is a necessary but not sufficient condition for the ergodic hypothesis to be satisfied, it gives intuition about the rate of state-space sampling. We have selected the state-space configuration as the quantity to average (alternatively an ergodic measure based on some other property, *e.g.* the energy could be used). This measure is an average over pair-wise differences between average state-space positions, for trajectories initiated in different minima during a simulation. More specifically, the average state-space position after S moves from a trajectory *initiated* at minimum a, containing configurations $\{\mathbf{x}_i^a, i=1..S\}$ obtained[7] during sampling run k is

[7] The trajectory may well include configurations inside minima basins other than a, but in a slight abuse of notation we will identify *both* the starting minima *and* the trajectory itself with the same letter.

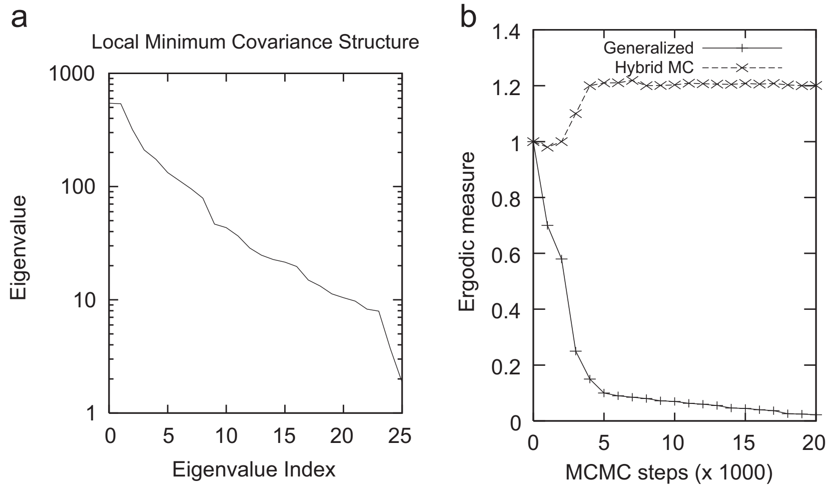

Fig. 5. (a, left) Top 25 eigenvalues of the covariance matrix (corresponding to a 35 state variable model) for a local minimum shows the typical ill-conditioning of the monocular human pose estimation. (b, right) The ergodic measure compared for a classical Langevin gradient-based sampling scheme and the generalized darting method. The classical sampler does not mix well—the long-term energy difference between trajectories reflects the memory of the minima where they were initiated. The generalized method mixes much better and explores various minima so the average state-space difference over long-term trajectories tends to zero (see text).

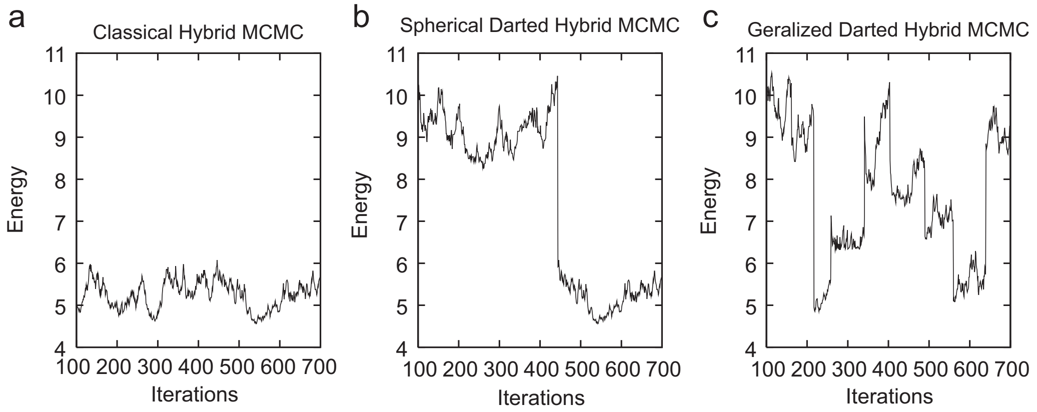

Fig. 6. Classical hybrid Monte Carlo (left) gets trapped in the starting minimum. Spherical darting explores the minima more thoroughly and one can see that only two minima are visited during 600 iterations (only one successful jump). Finally the generalized darting method (right) explores the different minima by combining local moves with non-local jumps that are accepted more frequently. Eight local minima are visited via seven jumps (note that after each jump the sampler explores its new local minimum for a while before the next jump).

given by

$$d_k^a(S) = \frac{1}{S}\sum_{i=1}^{S} \|\mathbf{x}_i^a\| \qquad (19)$$

and the ergodic measure is defined as the average between two trajectories initiated at different minima a and b in R runs[8]:

$$e(a,b,S,R) = \frac{1}{R}\sum_{k=1}^{R} [d_k^a(S) - d_k^b(S)]^2 \qquad (20)$$

For good mixing over large trajectories we expect the ergodic measure to converge to 0. In Fig. 5, we plot the ergodic measure corresponding to a classical Langevin simulation with no jumps against one using the generalized scheme for $S=20{,}000$ over $R=3$ runs. The mixing of the classical hybrid MCMC sampler is not satisfactory, perhaps reflecting the average state-space difference between the two local minima where the sampler is trapped, and which are explored repeatedly. In contrast, the long-range state self-averaging effect is clearly observed for generalized darting.

Learning model parameters: We run a parameter learning experiment using the same image in Fig. 4, the same state priors but using a more complex image *observation likelihood* based on *contour/edge measurements* (see Section 5.1). We estimate some of the model parameters, here the body proportions (36 parameters representing the superquadrics of the head, torso, upperarm, forearm, thigh and calf, with the symmetrical values on the left and right side of the body mirrored) and the variance used for the quadratic edge residual cost. We learn using Maximum Likelihood in a gradient-based framework as given in (14), and use generalized darting in order to approximate the partition function. This procedure is supervised, *i.e.* we need to specify the 3D state ground truth. Since this information was not available for our real scene, we selected, by visual inspection, one of the four pose configurations, considered to be the most plausible, as the ground truth (the second column in Fig. 4). The probability of the four configurations *before* and *after learning* is shown in Fig. 7. Learning takes seven iterations to converge on average. Notice how the process substantially improves the margin between the correct solution and the undesirable ones, in particular how the selected ground-truth emerged as most probable *after* learning despite not being the most probable *before*. 3D pose estimates based on the learned model identify the correct solution with significantly

[8] Note that there are two *different* simulations for each run k, one for a and another for b. Also notice that the subscript does not index the vector $\mathbf{x}$ but indicates different state vectors.

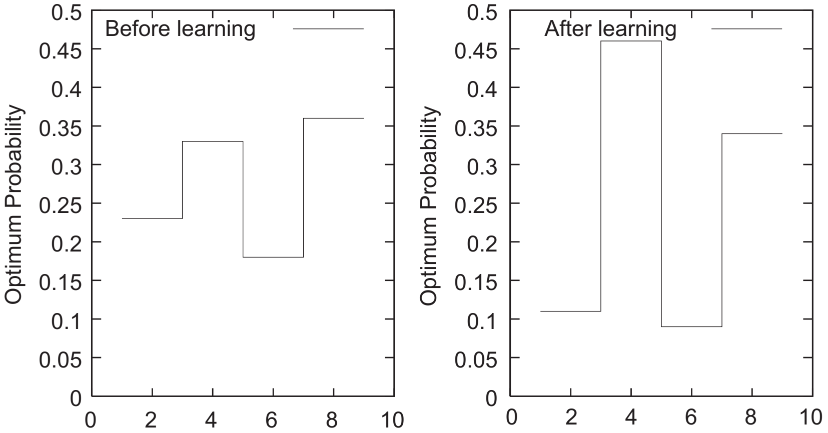

Fig. 7. Learning the body proportions and the variance of the observation likelihood based on matching contours improves the relative probability of the correct solution w.r.t. to undesirable ones. The plots show the probability levels of four different local optima (the numbering on the horizontal axis is irrelevant). The order of the probability peaks corresponds to the one shown in Fig. 4, with the second configuration visually selected as ground truth. Learning significantly increases the probability of the desired solution and downgrades competing, incorrect ones. 3D pose estimates based on the learned model identify the correct solution with higher probability.

higher probability, on the average. Learning does not make all the incorrect solutions extremely implausible due to several reasons. First, there is 3D structure sharing between the incorrect and the 'ground-truth' (*e.g.* solutions 2 and 4 share the upper body subcomponent of the state). Another factor may be the weak evidence provided by the contour features used for the observation likelihood. One can, *e.g.* use better lighting or surface reflection models in order to provide additional constraints to diminish uncertainty. Finally, since we are only able to find a local optimum for the parameters, it is possible that other good ones exist. However, we have not empirically identified better ones, even after multiple restarts from different initial starting points.

6. Discussion

In this paper we have discussed a new Markov chain Monte Carlo sampler that is able to effectively jump between modes in the target distribution while maintaining detailed balance. Our method is a generalization of 'darting MCMC' where the basic jump regions may have an arbitrary irregular shape and moreover are allowed to overlap. Generalizations to discrete and more general domains are also discussed.

An alternative view of some of the generalized darting samplers proposed in this paper is that of a mixture between an independence sampler and a Hybrid Monte Carlo sampler. In this view, we randomly alternate HMC sampling with proposing samples uniformly from the collection of regions $\{V_i\}$. The proposal distribution is not conditional on the previous sample, hence the name 'independence sampler'. However, to maintain detailed balance we cannot accept a proposal if the previous sample was located outside this collection of regions. Hence, instead of performing this check before proposing a new sample (as in darting MCMC), the check is implicitly performed after proposing a new sample by incorporating it in the acceptance rule.

Apart from the darting method [1], other MCMC schemes that mix between distant modes can be found in the literature.[9] In 'simulated tempering' [19], a new temperature random variable is introduced that extends the sample space in such a way that at high temperatures the energy function is much smoother. The temperature itself is also sampled via random walk. At high temperatures the Markov Chain mixes much faster between distant regions, while the samples acquired at $T=1$ are the desired samples from the target distribution. Neal extended this idea to his 'tempered transitions' method [20] that uses deterministic moves between low and high temperature regimes. In [21] the 'normal kernel coupler' is proposed to sample from multimodal distributions. The idea is to simulate N Markov chains in parallel with target distribution $p(\mathbf{x}) = \prod_i p_i(\mathbf{x})$, but to use proposal distributions based on a kernel estimate of all N particles. Finally [22] presents a generalized Metropolis–Hastings MCMC sampler is proposed that has the potential of incorporating deterministic optimization algorithms to locate local maxima in the target distribution. The inclusion of deterministic long range moves in a MCMC sampler for the purpose of computing the free energy of a physical system can also be found in the physics and chemistry literature [23,1,24–26].

The main advantage of the proposed generalized darting method is that one can tune the shape of the jump regions to match the shape of the high probability regions of the target distributions. This should help to achieve an improved acceptance probability of attempted jumps between regions. Note however that we do not claim that our method is superior to all earlier schemes under all circumstances. In fact, we have only compared our method with the classical darting method and shown improved acceptance rates. No doubt, the various methods described above will have different properties for different target distributions, or in the presence of different amounts of prior knowledge about the target distribution. We have made no further attempts to explore these issues in this paper.

In the absence of sufficient prior knowledge of the position and shape of the modes of the target distribution, the darting framework may suffer from unacceptably high rejection rates for long-range jumps. This problem will almost certainly be aggravated in high dimensions. The possibility to change the location and shape of the regions adaptively would be advantageous but difficult without violating the Markovian property of the chain.[10] Interesting ways around this obstacle do exist in the literature [27–30] but further research will be required to find out if they can be applied to the proposed generalized darting method.

[9] We do not claim that the listed methods are an exhaustive overview of the literature.

[10] The use of more sophisticated region shapes would also require an accurate volume computation method and an efficient inside/outside calculation during the MCMC simulation.

Acknowledgments

This work was supported by CNCSIS-UEFISCSU, project number PN II-RU-RC-2/2009 and by the European Commission under MCEXT-024581. We thank Geoff Hinton for feedback and earlier contribution to this work [33]. We also thank Radford Neal and Yee-Whye Teh for general comments.

Appendix A. Proof of detailed balance

The generalized darting Monte Carlo sampler can be viewed as a hybrid Monte Carlo sampler that is interrupted with a certain probability to attempt a long-range jump. Since hybrid Monte Carlo sampling is ergodic, a 'mixture' of hybrid Monte Carlo and any other (possibly non-ergodic) sampler is automatically ergodic as well.

To prove detailed balance between any pair of points in the sample space, we consider the following three possibilities:

1. Both points are located outside any of the jump-regions.
2. One of the two points is located inside one or more jump-regions while the other one is located outside any of the regions.
3. Both points are located in one or more of the regions.

1: When both points are located outside any of the jump-regions detailed balance follows because of the Markov chain for the local moves is assumed to respect detailed balance. With probability P_{check} this Markov chain is interrupted to check if the particle is located inside a jump-region. But since both points under consideration are assumed to be located outside any jump-region this interruption will be symmetric and does not destroy detailed balance.

2: The particle located outside any jump-region follows its local dynamics (i.e. it is not interrupted) with probability $1-P_{check}$. The particle inside one or more regions will also follow its local dynamics (i.e. it will not attempt a jump) with probability $1-P_{check}$. With probability P_{check} the sampler decides to perform a check. But in that case the particle outside any region will not move while the particle inside one or more regions will attempt a jump and will therefore never end up outside the set of all regions. Thus detailed balance again holds.

3: We will prove the case of two points in possibly overlapping regions, where the jump points are sampled uniformly at random inside a target region. The prove for the deterministic case goes along similar lines (see Section 3.1).

With probability $1-P_{check}$ we follow the local dynamics of the Markov chain which fulfills detailed balance by assumption. With probability P_{check} we initiate a jump to some other point in some other region. Define A to be the set of regions that contain point 1 and B the set of regions that contain point 2. We now have

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) \tag{21}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = p(\mathbf{x}_1)P_{check}\sum_{i \subset B}\frac{P_i}{V_i}P_{accept:1\to 2} \tag{22}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = p(\mathbf{x}_1)P_{check}\frac{n(\mathbf{x}_2)}{\sum_j V_j}\min\left[1,\frac{p(\mathbf{x}_2)n(\mathbf{x}_1)}{p(\mathbf{x}_1)n(\mathbf{x}_2)}\right] \tag{23}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = P_{check}\frac{1}{\sum_j V_j}\min[p(\mathbf{x}_1)n(\mathbf{x}_2),p(\mathbf{x}_2)n(\mathbf{x}_1)] \tag{24}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = p(\mathbf{x}_2)P_{check}\frac{n(\mathbf{x}_1)}{\sum_j V_j}\min\left[1,\frac{p(\mathbf{x}_1)n(\mathbf{x}_2)}{p(\mathbf{x}_2)n(\mathbf{x}_1)}\right] \tag{25}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = p(\mathbf{x}_2)P_{check}\sum_{i \subset A}\frac{P_i}{V_i}P_{accept:2\to 1} \tag{26}$$

$$p(\mathbf{x}_1)P(\mathbf{x}_1 \to \mathbf{x}_2) = p(\mathbf{x}_2)P(\mathbf{x}_2 \to \mathbf{x}_1) \tag{27}$$

where P_i (Eq. (3)) is the probability of jumping to region i and the factor $1/V_i$ is included because the target point is sampled uniformly at random inside this region. Thus, once again establish detailed balance.

Appendix B. Auxiliary variable formulation

The algorithm we presented in Section 3 can be formulated, more generally as an auxiliary variable method. Here the index of the regions (wormholes) plays the role of the auxiliary variable and we sample from the joint distribution over state space and region indexes using a mixture of Metropolis–Hastings transitions.

Consider $x \in \mathcal{X}$ the space and p the target distribution, and a covering with regions R_i (with volumes $V_i = |R_i|$) of $\mathcal{X}$ such that

$$\mathcal{X} = \bigcup_{i=1}^{N} R_i \tag{28}$$

Define, for any $\mathbf{x} \in \mathcal{X}$, $n(\mathbf{x}) = \sum_{i=1}^{N}\mathbb{I}(\mathbf{x} \in R_i)$, where $\mathbb{I}(\mathbf{x} \in A)$ is the indicator function for the set A. We use the following identity:

$$p(\mathbf{dx}) = p(\mathbf{dx})\sum_{i=1}^{N}\frac{\mathbb{I}(\mathbf{x} \in R_i)}{n(\mathbf{x})} \tag{29}$$

$$p(\mathbf{dx}) = \sum_{i=1}^{N}\frac{p(\mathbf{dx})/n(\mathbf{x})\mathbb{I}(\mathbf{x} \in R_i)}{\int_{R_i}p(\mathbf{dx})/n(\mathbf{x})}\int_{R_i}p(\mathbf{dx})/n(\mathbf{x}) \tag{30}$$

We can now define a joint probability distribution over the space defined by the Cartesian product of the region index set and the state space $\{1,\ldots,N\} \times \mathcal{X}$:

$$p(i,\mathbf{dx}) \equiv \frac{p(\mathbf{dx})/n(\mathbf{x})\mathbb{I}(\mathbf{x} \in R_i)}{\int_{R_i}p(\mathbf{dx})/n(\mathbf{x})}\int_{R_i}p(\mathbf{dx})/n(\mathbf{x}) \tag{31}$$

where based on Bayes' rule:

$$p(\mathbf{dx}|i) = \frac{p(\mathbf{dx})/n(\mathbf{x})\mathbb{I}(\mathbf{x} \in R_i)}{\int_{R_i}p(\mathbf{dx})/n(\mathbf{x})} \tag{32}$$

$$p(i) = \int_{R_i}p(\mathbf{dx})/n(\mathbf{x}) \tag{33}$$

In this auxiliary variable setting, we will sample from the joint target distribution $p(i,\mathbf{x})$ using a proposal distribution $Q(i,\mathbf{dx})$. The transition probability for the algorithm, excluding local moves and assuming the ratio is well defined, is

$$P(i,\mathbf{x};j,\mathbf{dy} = \mathbf{t}-\mathbf{x}) \tag{34}$$

$$= \sum_{k=1}^{N}\left\{\min\left(1,\frac{p(j,\mathbf{dy})Q(j,\mathbf{dy};i,\mathbf{dx})}{p(i,\mathbf{dx})Q(i,\mathbf{dx};j,\mathbf{dy})}\right)Q(j,\mathbf{dy})\right. \tag{35}$$

$$\left.+\delta_{i,\mathbf{x}}(j,\mathbf{dy})r(i,\mathbf{x})\right\} \tag{36}$$

where in the above $r(i,\mathbf{x})$ is the probability of not moving from $(i,\mathbf{x})$, either because a proposed move is rejected or because no move is attempted [31]. Choices of Q are given in the next sections.

B.1. Uniform sampling inside regions

For this case, we use

$$Q(i) = \frac{V_i}{\sum_{k=1}^{N} V_k} \square \quad (i \in 1, \ldots, N) \tag{37}$$

$$Q(\mathbf{dx}|i) = 1/V_i\mathbf{dx} \square \quad (\mathbf{x} \in R_i) \tag{38}$$

and the acceptance probability simplifies to (5).

B.2. Deterministic moves between regions

The second choice of proposal Q corresponds to deterministic moves [32], given in Section 3.1:

$$Q(i,\mathbf{x};j,\mathbf{dy}) = \delta_{F(i,\mathbf{x})}(j,\mathbf{dy}) \frac{V_i}{\sum_{k=1}^{N} V_k} \square \quad (i \in \{1, \ldots, N\}) \tag{39}$$

for a deterministic mapping $F : \mathcal{X} \to \mathcal{X}$. This typically requires a change of variables, thus a Jacobian term. We sample using (3), hence the need to consider the relative volume ratios in the equation.

References

[1] I. Andricioaiei, J. Straub, A. Voter, Smart darting Monte Carlo, J. Chem. Phys. 114 (16) (2001).

[2] R. Neal, Probabilistic inference using Markov chain Monte Carlo, Technical Report CRG-TR-93-1, University of Toronto, 1993.

[3] G. Celeux, M. Hurn, C. Robert, Computational and inferential difficulties with mixture posterior distributions, J. Am. Stat. Assoc. 95 (2000) 957–979.

[4] S. Duane, A.D. Kennedy, B.J. Pendleton, D. Roweth, Hybrid Monte Carlo, Physics Letters B 195 (2) (1987) 216–222.

[5] R. Neal, Annealed importance sampling, Statistics and Computing 11 (2001) 125–139.

[6] C. Sminchisescu, B. Triggs, Building roadmaps of local minima of visual models, in: European Conference on Computer Vision, vol. 1, Copenhagen, 2002, pp. 566–582.

[7] C. Sminchisescu, M. Welling, Generalized darting Monte-Carlo, in: Artificial Intelligence and Statistics, Puerto-Rico, 2007.

[8] G. Hinton, Training products of experts by minimizing contrastive divergence, Neural Computation 14 (2002) 1771–1800.

[9] M.C. Perpinan, G. Hinton, On contrastive divergence learning, Artificial Intelligence and Statistics, vol. 1, 2005.

[10] M. Isard, A. Blake, CONDENSATION – Conditional density propagation for visual tracking, Int. J. Comput. Vision 28 (1) (1998) 5–28.

[11] J. MacCormick, M. Isard, Partitioned sampling, articulated objects, and interface-quality hand tracker, European Conference on Computer Vision, vol. 2, 2000, pp. 3–19.

[12] J. Deutscher, A. Blake, I. Reid, Articulated body motion capture by annealed particle filtering, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2000.

[13] K. Choo, D. Fleet, People tracking using hybrid Monte Carlo filtering, in: IEEE International Conference on Computer Vision, 2001.

[14] C. Sminchisescu, B. Triggs, Hyperdynamics importance sampling, in: European Conference on Computer Vision, vol. 1, Copenhagen, 2002, pp. 769–783.

[15] C. Sminchisescu, B. Triggs, Kinematic jump processes for monocular 3D human tracking, in: IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, Madison, 2003, pp. 69–76.

[16] M. Lee, I. Cohen, Proposal maps driven MCMC for estimating human body pose in static images, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2004.

[17] C. Sminchisescu, B. Triggs, Estimating articulated human motion with covariance scaled sampling, International Journal of Robotics Research 22 (6) (2003) 371–393.

[18] R. Fletcher, Practical Methods of Optimization, John Wiley, 1987.

[19] E. Marinari, G. Parisi, Simulated tampering: a new Monte Carlo scheme, Europhys. Lett. 19 (6) (1992).

[20] R. Neal, Sampling from multimodal distributions using tempered transitions, Statistics and Computing 6 (1996) 353–366.

[21] G. Warnes, The normal kernel coupler: an adaptive Markov chain Monte Carlo method for efficiently sampling from multi-modal distributions, PhD Thesis, University of Washington, 2000.

[22] H. Tjelmeland, B.K. Hegstad, Mode jumping proposals in MCMC, Technical Report, Norwegian University of Science and Technology, Trondheim, Norway, preprint Statistics No. 1/1999, 1999.

[23] A. Voter, A Monte Carlo method for determining free-energy differences and transition state theory rate constants, J. Chem. Phys. 82 (4) (1997).

[24] H. Senderowitz, F. Guarnieri, W. Still, A smart Monte Carlo technique for free energy simulations of multiconformal molecules. Direct calculation of the conformational population of organic molecules, J. Am. Chem. Soc. 117 (1995).

[25] M. Miller, W. Reinhardt, Efficient free energy calculations by variationally optimized metric scaling, J. Chem. Phys. 113 (17) (2000).

[26] C. Jarzynski, Targeted free energy perturbation, Technical Report LAUR-01-2157, Los Alamos National Laboratory, 2001.

[27] W.R. Gilks, G.O. Roberts, S.K. Sahu, Adaptive Markov chain Monte Carlo through regeneration, J. Am. Stat. Assoc. 93 (443) (1998) 1045–1054.

[28] C. Andrieu, E. Moulines, On the ergodicity properties of some adaptive MCMC algorithms, Technical Report, University of Bristol, 2002.

[29] H. Haario, E. Saksman, J. Tamminen, An adaptive Metropolis algorithm, Bernoulli 7 (2).

[30] Y. Atchade, J. Rosenthal, An adaptive Markov chain Monte Carlo algorithms, Technical Report, University of Montreal, 2003.

[31] P. Green, Reversible jump MCMC computation and Bayesian model determination, Biometrika 82 (1995) 711–732.

[32] L. Tierney, A note on Metropolis–Hastings kernel for general state spaces, Ann. Appl. Prob. 8 (1) (1998) 1–9.

[33] C. Sminchisescu, M. Welling, G. Hinton, A mode-hopping MCMC sampler, Technical Report CSRG-478, University of Toronto, September 2003.

Cristian Sminchisescu has obtained a doctorate in Computer Vision and Applied Mathematics from INRIA in 2002 and has been a research fellow at the University of Toronto during 2003–2004. His research interests are in the area of computer vision and machine learning with a focus on probabilistic methods, optimization and sampling algorithms, and latent variable models, with applications to human motion analysis and object recognition. He is Faculty member at the University of Bonn, and holds a professor equivalent senior scientist title at the Institute for Mathematics (IMAR), and a professor status appointment at the University of Toronto.

Max Welling is a professor at the University of California at Irvine where he is directing a research group in Statistical Computation, Information, Vision and Inference. He has obtained a Ph.D. in Physics in 1998 from the University of Utrecht and has been a research fellow working in Machine Learning at Gatsby Computational Neuroscience Unit, London, and the University of Toronto. Welling's research area can broadly be categorized as machine learning and machine vision with links to closely related areas such as pattern recognition, data mining, computational statistics, and large-scale data analysis.