Appearing in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2003, Madison.*

Kinematic Jump Processes For Monocular 3D Human Tracking

Cristian Sminchisescu* **Bill Triggs**
INRIA Rhône-Alpes, GRAVIR, 655 avenue de l'Europe, 38330 Montbonnot, France
{*Cristian.Sminchisescu,Bill.Triggs*}*@inrialpes.fr, www.inrialpes.fr/movi/people/*{*Sminchisescu,Triggs*}

Abstract

A major difficulty for 3D human body tracking from monocular image sequences is the near non-observability of kinematic degrees of freedom that generate motion in depth. For known link (body segment) lengths, the strict non-observabilities reduce to twofold 'forwards/backwards flipping' ambiguities for each link. These imply $2^{\#\,links}$ *formal inverse kinematics solutions for the full model, and hence linked groups of* $\mathcal{O}(2^{\#\,links})$ *local minima in the model-image matching cost function. Choosing the wrong minimum leads to rapid mistracking, so for reliable tracking, rapid methods of investigating alternative minima within a group are needed. Previous approaches to this have used generic search methods that do not exploit the specific problem structure. Here, we complement these by using simple kinematic reasoning to enumerate the tree of possible forwards/backwards flips, thus greatly speeding the search within each linked group of minima. Our methods can be used either deterministically, or within stochastic 'jump-diffusion' style search processes. We give experimental results on some challenging monocular human tracking sequences, showing how the new kinematic-flipping based sampling method improves and complements existing ones.*

Keywords: *Monocular 3D human body tracking, kinematic ambiguity, Covariance Scaled Sampling, inverse kinematics, particle filtering, constrained optimization, high-dimensional search.*

1 Introduction

A major difficulty for 3D human body tracking from monocular image sequences is the quasi-unobservability of kinematic degrees of freedom that generate motion in depth. For unknown limb (link) lengths this leads to continuous nonrigid 'affine folding' ambiguities, but once lengths are known these reduce to twofold 'forwards/backwards flipping' ambiguities for each link. The full model thus has $2^{\#\,\text{links}}$ formal inverse kinematics solutions. Even with strong joint limits and no image correspondence ambiguities, the model-image matching cost function typically still has $\mathcal{O}(2^{\#\,\text{links}})$ local minima, so optimizing it is a difficult global search problem. But also a necessary one, as following the wrong local minimum rapidly leads to mistracking.

Several generic global search methods have already been applied to this problem [4, 11, 14, 16], but they tend to be somewhat inefficient as they make little use of the specific problem structure. Here, we develop a new method that speeds the search for local minima by using simple kinematic principles to construct 'interpretation trees' generating the possible 3D body configurations associated with a given set of projected joint centres (§3). We give simple closed-form inverse kinematics solutions for constructing these trees for human limbs, and show how the method can be used to produce an efficient deterministic 'kinematic jump' sampler for the different configurations. We use this sampler to construct a novel mixture density propagation based tracking algorithm (§4) that combines local covariance based diffusion, adaptive kinematic chain selection based on local uncertainties, quasi-global kinematic jumps and local continuous constrained optimization. We present quantitative results showing the effectiveness of the new samplers compared to existing methods (§5.1), and conclude with some challenging monocular experiments showing the final tracker's ability to follow rapid, complex human motions in clutter.

1.1 Related Research

There is a large literature on human motion tracking but relatively little work on developing search methods that exploit both the local and global structure present in the 3D monocular articulated problem. Sidenbladh *et al* use particle filtering with importance sampling based on either a learned walking model or a database of motion snippets, to focus search in the neighborhood of known trajectory pathways [11, 12]. Deutscher *et al* propose an annealing framework in a multi-camera setting [3]. During annealing, the search for parameters is driven by noise proportional with their individual variances [4]. Considered as an improved (implicit) search space decomposition mechanism, an early method of this type was proposed by Gavrila & Davis [6] to efficiently sample partial kinematic chains. Adaptively identifying and sampling parameters with high variance is useful, but kinematic parameters usually have quite strong interactions that make simple axis-aligned sampling questionable. It is important to realize that the principal axes of the covariance change drastically depending on the viewing direction, and that even if these are computed and used for sampling (as in [14]), they are only local measures that capture little information about the global

*Current affiliation: University of Toronto, Department of Computer Science, Artificial Intelligence Laboratory, 6 King's College Road, Toronto, Canada, M5S 3G4; *crismin@cs.toronto.edu, www.cs.toronto.edu/~crismin.*

minimum structure.

Sminchisescu & Triggs [14] argue that an effective random sampler must combine all three of cost-surface-aware covariance scaling, a sampling distribution with widened tails for deeper search, and local optimization (because deep samples usually have very high costs, and hence will not be resampled even if they lead to other minima). More recently, they have also constructed deterministic optimization methods [15] and cost-function-modifying MCMC samplers [16], for finding 'transition states' (saddle points) leading to nearby minima.

Skeletal reconstruction methods recover an interpretation tree of possible 3D joint positions, based on user-specified image joint positions [8, 17]. Lee & Chen [8] attempt to prune their perspective interpretation tree using physical reasoning, while Taylor [17] relies on additional user input to specify plausible relative joint-centre depths for his affine one. Although these methods do incorporate the forward-backward flipping ambiguity, they can not reconstruct skeletal joint angles, and this makes them inappropriate for tracking applications.

Our approach can be seen as a marriage of locally optimized covariance based random sampling with a domain-specific deterministic sampler based on skeletal reconstruction using inverse kinematics. The local covariance information obtained during optimization also provides a useful heuristic for which kinematic parameters to sample.

2 Modeling and Estimation

Representation The 3D body model used in our human tracking experiments consists of a kinematic 'skeleton' of articulated joints controlled by angular joint parameters, covered by a 'flesh' built from superquadric ellipsoids with additional global deformations [1]. A typical model has 30–35 joint parameters; 8 internal proportions encoding the positions of the hip, clavicle and skull tip joints; and 9 deformable shape parameters for each body part. The complete model is encoded in a single large parameter vector $\mathbf{x}$. During tracking and static pose estimation we usually estimate only joint parameters, but during initialization some length ratios are also estimated. In use, the superquadric surfaces are discretized into 2D meshes and the mesh nodes are mapped to 3D points using the kinematic body chain then projected to predicted image points $\mathbf{r}_i(\mathbf{x})$ using perspective image projection.

Observation Likelihood: During tracking robust model-to-image matching cost metrics are evaluated for each predicted image feature $\mathbf{r}_i$, and the results are summed over all observations to produce the image contribution to the parameter space cost function. Cost gradient and Hessian contributions $\mathbf{g}_i, \mathbf{H}_i$ are also computed and assembled. We use a robust combination of extracted-feature-based metrics and intensity-based matching ones (registering the model reprojected texture at previous tracking step with the current image) and robustified normalized edge energy. The feature-based terms associate the predictions $\mathbf{r}_i$ with nearby image features $\bar{\mathbf{r}}_i$, the cost being a robust function of the prediction errors $\mathbf{\Delta r}_i(\mathbf{x}) = \bar{\mathbf{r}}_i - \mathbf{r}_i(\mathbf{x})$. We also give results for a simpler likelihood designed for model initialization, based on squared distances between reprojected model joints and their specified image positions.

Priors and Constraints: Our model [14] incorporates both hard constraints (for joint angle limits) and soft priors (penalties for anthropometric model proportions, collision avoidance between body parts, and stabilization of useful but hard-to-estimate model parameters such as internal d.o.f. of the clavicle complex). The priors provide additional cost, gradient and Hessian contributions for the optimization.

Estimation: We apply Bayes rule and maximize the total posterior probability to give locally MAP parameter estimates:

$$\log p(\mathbf{x}|\bar{\mathbf{r}}) \propto \log p(\mathbf{x}) + \log p(\bar{\mathbf{r}}|\mathbf{x}) = \log p(\mathbf{x}) - \int e(\bar{\mathbf{r}}_i|\mathbf{x})\, di \tag{1}$$

Here, $p(\mathbf{x})$ is the prior on the model parameters, $e(\bar{\mathbf{r}}_i|\mathbf{x})$ is the cost density associated with observation i, and the integral is over all observations (assumed independent). Equation (1) gives the model likelihood in a single image, under the model priors but without initial state or temporal priors. During tracking, the temporal prior at time t is determined by the previous posterior $p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1})$ and the system dynamics $p(\mathbf{x}_t|\mathbf{x}_{t-1})$, where we have collected the observations at time t into vector $\mathbf{r}_t$ and defined $\mathbf{R}_t = \{\mathbf{r}_1, \ldots, \mathbf{r}_t\}$. The posterior at t becomes

$$p(\mathbf{x}_t|\mathbf{R}_t) \propto p(\bar{\mathbf{r}}_t|\mathbf{x}_t)\, p(\mathbf{x}_t) \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1})$$

Together $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1})$ form the time t prior $p(\mathbf{x}_t|\mathbf{R}_{t-1})$ for the image correspondence search (1).

3 Kinematic Jump Processes

Each configuration of the skeletal kinematic tree has an associated *interpretation tree* — the tree of all fully- or partially-assigned 3D skeletal configurations that can be obtained from the given one by forwards/backwards flips. The tree contains only, and generically all, configurations that are image-consistent in the sense that their joint centres have the same image projections as the given one. (Some of these may still be inconsistent with other constraints: joint limits, body self-intersection, occlusion...). The interpretation tree is constructed by traversing the kinematic tree from the root to the leaves. For each link, we construct the 3D sphere centred on the currently hypothesized position of the link's root, with radius equal to link length. This sphere is pierced by the camera ray of sight through the observed image position of the link's endpoint to give (in general) two possible 3D positions of the endpoint that are consistent with the image observation

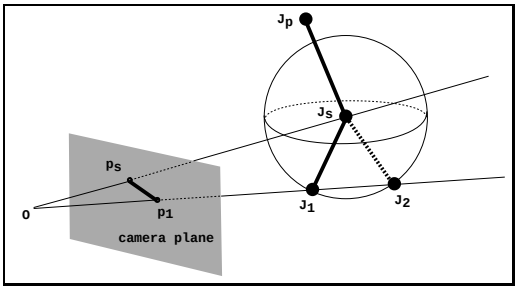

Figure 1: Forwards/backwards ambiguity for a kinematic link under monocular perspective projection. Given a standard joint configuration $...J_pJ_sJ_1$, one can build an alternative 'flipped' configuration $...J_pJ_sJ_2$ with the same joint-centre image projections. J_2 is found by intersecting the sphere centered at J_s with radius $|J_sJ_1|$ with the camera line of sight through the projection of J_1, OJ_1.

and the hypothesized parent position (see fig. 1). Joint angles are then recovered for each position using simple inverse kinematics (see below). If the ray misses the sphere, the parent hypothesis was inconsistent with the image data and the branch can be pruned.

More precisely, the above tree structure applies to non-branching kinematic chains such as limbs. When there is kinematic branching — *e.g.* for the four limbs attached to the trunk — each branch can be sampled independently, so the set of possible interpretations has a natural factored 'product of trees' structure. In such cases we build independent trees for each limb and take their product, *e.g.* each full-body configuration contains independently-sampled configurations for each of the four limbs.

Compared to current generic configuration space sampling methods, forwards/backwards flipping generates high-quality hypotheses very rapidly, and also provides unusually thorough coverage, at least within each kinematically-induced equivalence class of minima. Its quality stems from the fact that the hypotheses generated all have approximately-correct image projections (in particular, correct joint-centre projections). Its rapidity stems from the existence of simple closed form solutions for the inverse kinematics in this particular case (*i.e.* flexible kinematics constrained by observed joint-centre projections), and the fact that the accurate hypotheses generated do not need further 'polishing' by expensive non-linear optimization.

One could also generate 'flips' using classical closed-form or iterative techniques for solving the full inverse kinematics of the articulated skeleton, *e.g.* [10, 18]. However these methods are not well-adapted to this application in the sense that they solve a much more complicated problem (full redundant kinematics from a given end-effector pose) while ignoring much of the available image information (constrained projections of intermediate joint centres).

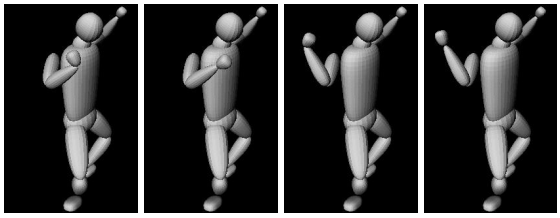

Figure 2: The 'flipping' ambiguities of the forearm and hand under monocular perspective. (The left-most configuration violates a wrist joint-angle limit and will be pruned away).

3.1 Direct Inverse Kinematics

As described above, flipping applies only to kinematic chains with fully spherical joints. Single d.o.f. joints such as hinges are usually too rigid to have a flipping ambiguity, as two d.o.f. are needed to move the link end to an arbitrary new position on the sphere. However, for human kinematics, flipping ambiguities apply even to hinge joints such as the elbow: although physically a hinge, the elbow effectively has spherical mobility once axial rotations of the upper arm about the shoulder are included. Here we give the inverse kinematics of this three link case as an example. We work in a reference coordinate system and know the 3D positions $\mathbf{P}_i$ of joints J_i, $i = 1..4$, as well as the rotational displacement $\mathbf{R}$ of J_1 with respect to the reference frame. The kinematic chain is represented in terms of Euler angles and pure translations along negative z axes. We use $\widehat{\mathbf{R}}$ to denote the z column of the rotation matrix $\mathbf{R}$. Suppose $\mathbf{p}_i = \frac{\mathbf{P}_i - \mathbf{P}_{i+1}}{\|\mathbf{P}_i - \mathbf{P}_{i+1}\|}$, with $i = 1..3$ unit vectors specifying the (known) z axes at each individual joint, after applying the rotation in that joint. There are 3 d.o.f. in J_1, 1 d.o.f. in J_2 and 2 d.o.f. in J_3 – these are represented by rotation matrices $\mathbf{R}^{1,2,3}_{x,y,z}$ as in fig. 3. To solve for rotations, we descend the kinematic chain and factor rotation angles $(x, y, z)_{1,2,3}$ by applying the constraints derived from the known positions of $\mathbf{P}_i$. The key observation is that, at any joint J_i, given the known previous rotational displacement, we have to factor out a rotation that aligns the z-axis with $\mathbf{p}_i$. For instance, at J_1, $\widehat{\mathbf{R}^1_x\mathbf{R}^1_y\mathbf{R}^1_z} = \mathbf{R}^T\mathbf{p}_i$ and we extract x_1, y_1 from:

$$\begin{pmatrix} -\sin(y_1) \\ \sin(x_1)\cos(y_1) \\ \cos(x_1)\cos(y_1) \end{pmatrix} = \mathbf{R}^T\mathbf{p}_1$$

In general this gives 4 solutions for x_1, y_1, but usually 2 do not satisfy all 3 equalities and are removed. z_1 is then recovered together with x_2 by solving $\widehat{\mathbf{R}^1_z\mathbf{R}^2_x} = (\mathbf{R}\mathbf{R}^1_x\mathbf{R}^1_y)^T\mathbf{p}_2$ for the next joint J_2:

$$\begin{pmatrix} \sin(z_1)\sin(x_2) \\ \cos(z_1)\sin(x_2) \\ \cos(x_2) \end{pmatrix} = (\mathbf{R}\mathbf{R}^1_x\mathbf{R}^1_y)^T\mathbf{p}_2$$

Again there are 4 possible solutions but 2 can be pruned. Finally, x_3, y_3 are obtained in the same way as x_1, y_1, given the

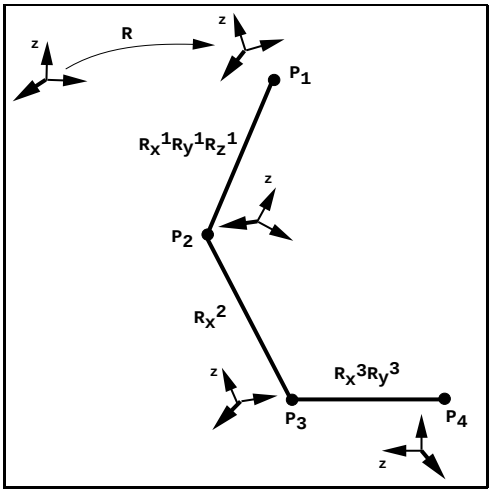

Figure 3: A three-joint link modeling anthropometric limbs. It has one spherical joint J_1, one hinge joint J_2 and a 2 d.o.f. end effector J_3. The representation is built in terms of Euler angles (with associated rotation matrices $\mathbf{R}^{1,2,3}_{x,y,z}$ with angles as sub-scripts and the joint rotation centers as superscripts) and pure translations to the next joint along the negative z axis. The inverse kinematics solution factors rotation angles using knowledge of successive z axes (computed from $\mathbf{P}_i - \mathbf{P}_{i+1}$) for limbs.

known x_1, y_1, z_1, x_2 values. As as special case, note that $\mathbf{R}^1_z$ remains unconstrained when $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{P}_3$ are collinear. In this case, z_1 is either fixed to some default value or (for tracking) sampled within its range of variation.

3.2 Iterative Inverse Kinematics

In some situations, the simple closed form inverse kinematics given above does not suffice. This might happen for more general kinematic structures — for example the looped kinematic chains formed when the hands are joined or placed on the hips — or when the exact inverse kinematics either fails (a camera ray does not intersect its sphere) or is expected to be inaccurate for some reason (a joint limit or body non-self-intersection constraint is violated). In such cases, we can fall back on a more general approach that directly minimizes the sum of squared differences between the current and desired joint configurations, using nonlinear optimization in joint space. Our minimizer uses analytical gradients and Hessians in a second-order damped Newton trust-region framework, with both hard joint-angle limits and soft non-self-intersection and image correspondence constraints [14]. In practice, this method locates new flipped local minima fairly successfully, but is significantly more expensive than kinematics-based flipping as $\mathcal{O}(1)$ full local optimization runs are needed for each new minimum found. However this is still significantly more efficient than the random samplers we have tested — see §5.

4 The Algorithm

In normal use, we embed our kinematic jump sampler within a cost-sensitive mixture density propagation framework [14]. The jump sampler ensures rapid, consistent diffusion of samples across the kinematic minima associated with any given set of image joint positions, while the random sampler provides robustness against incorrect image correspondences. Here, we use a Covariance Scaled Sampling [14] tracker. This probabilistic method represents the posterior distribution of hypotheses in joint space as a mixture of long-tailed Gaussian-like distributions $m_i \in \mathcal{M}$, whose weights, centres and scale matrices ('covariances') $m_i = (c_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ are obtained as follows. Random samples are generated, and each is optimized (by nonlinear local optimization, respecting any joint constraints, *etc.*) to maximize the local posterior likelihood encoded by an image- and prior-knowledge based cost function. The optimized likelihood value and position give the weight and centre of a new component, and the inverse Hessian of the log-likelihood gives a scale matrix that is well adapted to the contours of the cost function, even for very ill-conditioned problems like monocular human tracking. However, when sampling, particles are deliberately scattered more widely than a Gaussian of this scale matrix (covariance) would predict, in order to probe more deeply for alternative minima.

Fig. 4 gives the general form of the algorithm, and fig. 5 describes the novel **KinematicDiffusionJumpSampling** routine that lies at its core. On entry, the user specifies a set $\mathcal{C}$ of kinematic sub-chains that may be sampled (this can be quite large, as the routine adaptively decides which to sample). At each time step, covariance scaled samples are generated from the prior. For each such sample an interpretation tree is created on-line by the **BuildInterpretationTree** routine, with kinematic solutions obtained using **InverseKinematics**. The chain to be sampled is chosen adaptively using a voting process based on the local covariance structure of that region of the parameter space, **SelectSamplingChain** in fig. 5. Local covariance scaled resampling is performed before the jump because we do not (yet) have the covariance information needed to perform it afterwards. Each element of the sampleable sub-chain set $\mathcal{C}$ is simply a list of parameter names to sample. For instance, for a sub-chain rooted at the left shoulder, this might include the rotational parameters $(x_s, y_s, z_s, x_e, x_h, y_h)$ where the (s, e, h) stand for (shoulder, elbow, hand) and x, y, z for the rotation axes.

The proposed sampling strategy provides a balance between local and global search effort since samples are generated around the prior modes, as well as around new peaks that are potentially emerging and have not yet been explored. Re-weighting based on closest prior modes as in fig. 4, step 5, ensures the tracker is not distracted by remote multi-modality when tracking the correct minima.

Kinematic Jump + CSS Diffusion Based Tracker

Input: The set $\mathcal{C}$ of permissible kinematic chain partitions to use for sampling, and the previous posterior $p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1}) = \sum_{i=1}^{K} \pi_i^{t-1} \mathcal{N}(\mu_i^{t-1}, \Sigma_i^{t-1})$.

---

1. Build the covariance scaled proposal density $p^*(t-1) = \sum_{i=1}^{K} \pi_i^{t-1} \mathcal{N}(\mu_i^{t-1}, s\Sigma_i^{t-1})$. ($s \sim$ 4-6).

---

2. Generate a set of samples S using **KinematicDiffusionJumpSampling** on $p^*(t-1)$ and $\mathcal{C}$.

---

3. Optimize each sample $s_j \in S$ w.r.t. the time t observation likelihood (1), using local constrained optimization to get MAP estimates μ_j^t with covariances $\Sigma_j^t = \mathbf{H}(\mu_j^t)^{-1}$.

---

4. Construct the unpruned posterior $p_t^u(\mathbf{x}_t|\mathbf{R}_t) = \sum_{j=1}^{N} \pi_j^t \mathcal{N}(\mu_j^t, \Sigma_j^t)$, where $\pi_j^t = \frac{p(\mu_j^t|\bar{\mathbf{r}}_t)}{\sum_{j=1}^{N} p(\mu_j^t|\bar{\mathbf{r}}_t)}$, and prune it to keep the K components with highest probability: $p_t^p(\mathbf{x}_t|\mathbf{R}_t) = \sum_{k=1}^{K} \pi_k^t \mathcal{N}(\mu_k^t, \Sigma_k^t)$, with $\pi_k^t = \frac{p(\mu_k^t|\bar{\mathbf{r}}_t)}{\sum_{j=1}^{K} p(\mu_j^t|\bar{\mathbf{r}}_t)}$.

---

5. For each mixture component $j = 1..K$ in p_t^p, find the closest prior component i in $p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1})$ in Bhattacharyya distance $\mathcal{B}_{ij}(\mu_i^{t-1}, \Sigma_i^{t-1}, \mu_j^t, \Sigma_j^t)$. Scale $\pi_j^t = \pi_j^t * \pi_i^{t-1}$ and discard component i of $p(\mathbf{x}_{t-1}|\mathbf{R}_{t-1})$.

---

6. Compute the final posterior mixture $p(\mathbf{x}_t|\mathbf{R}_t) = \sum_{k=1}^{K} \pi_k^t \mathcal{N}(\mu_k^t, \Sigma_k^t)$, with $\pi_k^t = \frac{\pi_k^t}{\sum_{j=1}^{K} \pi_j^t}$.

Figure 4: The steps of our mixture density propagation algorithm.

5 Experiments

This section gives experiments showing the performance of our new Kinematic Jump Sampling (KJS) method relative to two established random sampling methods, cost-surface-sensitive Covariance Scaled Sampling (CSS) [14] and the traditional cost-insensitive Spherical Sampling (SS) method used implicitly, *e.g.* in CONDENSATION [7][1].

5.1 Quantitative Evaluation

Our first set of experiments studies the quantitative behavior of the different sampling methods, particularly their efficiency at locating minima or low-cost regions of parameter space. We study performance for different kinematic partitions of the joint space under deterministic Kinematic Jump Sampling (KJS), and also give results for the random Covariance Scaled (CSS) and Spherical (SS) samplers, showing how different core shapes (spherical vs. local covariance-based) and tail widths (scaled-Gaussian versus Cauchy) affect their efficiency. The study was based on the simple, but still highly multi-modal, model-joint to known-image-joint likelihood function that we use to initialize our 34 d.o.f. articulated model[2]. The model started at approximately its true 3D con-

$S =$ **KinematicDiffusionJumpSampling**$(\{p^*\}, \mathcal{C})$

Generates a set of samples S based on Covariance Scaled Sampling diffusion and kinematic jump processes.

1. Use **SelectSamplingChain**$(\Sigma_i, \mathcal{C})$ to select a kinematic chain $C_i \in \mathcal{C}$ to sample for each mixture component p_i^*.
2. Generate N random samples as follows:
2.1. Choose a mixture component p_i^* with probability π_i.
2.2. CSS sample from p_i^* to obtain $\mathbf{s}_j$.
2.3. T_j = **BuildInterpretationTree**$(\mathbf{s}_j, C_i)$.
2.4. For each path (list of 3D joint positions) P in T_j, use **InverseKinematics**(P) to find joint angles c_P, and add c_P to the list of samples, $S = S \cup c_P$.

SelectSamplingChain$(\Sigma, \mathcal{C})$

Heuristic to select a chain $C \in \mathcal{C}$ to sample for a component with covariance Σ. $C = \cup_{i=1}^{M} C_i$. The function $\mathrm{Idx}(C_i)$ will provide the index of parameter C_i in the N-d skeleton joint state.

1. Diagonalize Σ to obtain $(\mathbf{v}_j, \sigma_j)_{j=1..N}$.
2. For each chain $C \in \mathcal{C}$, find
$\mathrm{vote}_k = \sum_{i=1}^{M} \sum_{j=1}^{N} \sigma_j \mathbf{v}_j[\mathrm{Idx}(C_i)]$
Intuitively this counts the cumulated uncertainty of C along the local covariance principal directions $\mathbf{v}_j$, weighted by their corresponding standard deviations σ_j.
3. Return the chain C with the highest vote. (Alternatively, the best k chains could be returned, or a vote-weighted random one).

BuildInterpretationTree$(\mathbf{s}, C)$

Builds the interpretation tree for $\mathbf{s}$ based on flipping the variables in chain C (§3).

InverseKinematics(P)

Uses either closed-form (§3.1) or iterative (§3.2) inverse kinematics to find the joint-space configuration associated with a list of 3D joint positions P.

Figure 5: The components of our CSS diffusion plus kinematic jump sampling algorithm.

[1] CONDENSATION samples 'spherically' in the sense that the source of randomness is Gaussian dynamical noise with a fixed prespecified covariance. We could choose coordinates in which this distribution was spherically symmetric. Whereas in CSS, the 'noise' adapts to the local cost surface at each time step.

[2] Our full initialization procedure also estimates some body dimensions, but here these are held fixed.

METHOD	SCALE	NUMBER OF MINIMA	MEDIAN PARAMETER DISTANCE		MEDIAN STANDARD DEVIATION		MEDIAN COST	
			NO OPT	OPT	NO OPT	OPT	NO OPT	OPT
KJS1	-	1024	2.9345	2.8378	92.8345	93.9628	0.0998	0.0212
KJS2	-	1466	3.2568	2.2986	83.4798	82.5709	0.1045	0.0203
CSS	1	8	1.1481	2.5524	10.9351	47.6042	116.9512	8.4968
CSS	4	59	3.2123	2.9474	35.2918	55.3163	1995.1232	6.9810
CSS	8	180	4.9694	3.3466	75.1119	109.8131	16200.8134	7.0986
CSS	16	667	6.4242	6.7209	177.1111	465.8892	45444.1223	8.6958
CSS	1/HT	580	5.0536	6.9362	106.6311	517.3872	15247.7134	8.7242
SS	1	0	0.1993	-	24.5274	-	273.5091	-
SS	4	11	0.7673	2.0492	96.1519	39.0745	4291.1211	6.2801
SS	8	42	1.4726	2.5488	188.1571	56.8268	16856.1211	6.9648
SS	16	135	2.7195	2.8494	367.7461	87.8533	63591.4211	8.6958
SS	1/HT	232	2.1861	6.5474	178.6471	535.9991	18173.1121	17.8807

Table 1: Quantitative results on sample distribution for KJS, as well as CSS and SS with different types of tails (scaled-Gaussian vs. HT, with and without optimization NO OPT vs. OPT). KJS finds 1024 minima in 1024 samples for the first trial and 1466 minima in 1536 samples for the second round. The CSS/SS experiments used 2000 samples. Note that KJS finds many more minima than SS and CSS, and that its samples are already very close to the final minima in cost, whereas SS and CSS samples require a substantial amount of optimization to become plausible hypotheses. Also note that CSS has significantly better performance than SS, both in terms of numbers of minima found and median costs of raw samples.

figuration.

Table 1 summarizes the results, giving the number of minima found by each method, and also their median costs (likelihoods relative to the true configuration) and their distances from the starting configuration in both spherical parameter space units and covariance-scaled standard deviations. It gives statistics both for raw samples, and for samples after local continuous optimization subject to joint and body non-self-intersection constraints. Fig. 6 shows some histograms of numbers of samples and minima found versus parameter space and Mahalanobis distance.

Spherical and Covariance Scaled Sampling: CSS and SS were run with both Gaussian and heavy tailed (HT Cauchy) distributions, using 2000 samples per run. For a fairer comparison we kept the volume of the distribution cores constant: the volume of the unit covariance CSS ellipsoid is always equal to the volume of the corresponding sphere, *i.e.* the sphere's radius is taken to be $R = \sqrt[n]{\lambda_1 ... \lambda_n}$, where λ_i are the eigenvalues of the covariance ellipsoid. We ran the methods for Gaussian distributions with scaling 4,8,16 and Cauchy distributions with scaling 1. Samples that violated physical constraints were projected back onto the feasible constraint surface. This often leads to highly non-Gaussian features such as multi-peaked histograms, even though the raw sampling distribution is Gaussian.

In the results, note the significantly greater number of local minima found by CSS than by SS, and also that CSS samples on average have much lower cost than SS ones. One can also see the large cost difference between unoptimized (NO OPT) and optimized (OPT) samples. Although the table seems to show that SS generates slightly lower-cost optimized minima than CSS, this is illusory. SS is simply too myopic to find more than a few close-lying (and hence low cost) minima, whereas CSS reliably finds both these and also many more distant ones, some of which naturally have somewhat higher cost.

Kinematic Jump Sampling: We ran KJS for several different partitions of the skeleton into sampleable subchains. Experiment KJS1 sampled the left and right shoulder joints and the left calf, for a frontal view similar to the one in fig. 2. Each of the 1024 configurations generated lead to a distinct local minimum after optimization. The second experiment KJS2 sampled the left and right calf joints and the right shoulder joint for a total of 1536 samples leading to 1466 minima after optimization. In both cases the parameter space minima were hit quite accurately, so optimization is largely superfluous. The KJS samples also have far lower costs than raw SS or CSS samples. Thus, KJS sampling is also likely to be effective when used with optimization-free discrete density propagation methods such as CONDENSATION.

5.2 Tracking

Finally, we illustrate the full KJS + CSS method on a 4 s sequence involving full-body tracking of a subject performing agile and rapid dancing moves. This sequence contains both self-occlusion and significant relative motion in depth. It was shot at 25 frames (50 fields) per second against a cluttered,

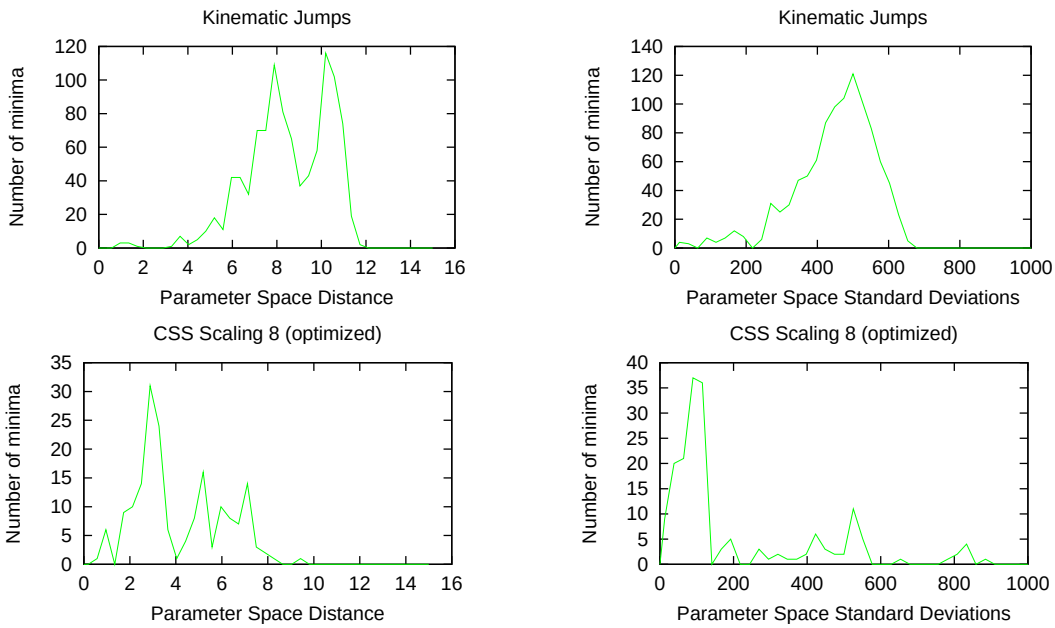

Figure 6: *Top:* Distribution of optimized parameter space distance and standard deviation for the KJS1 experiment. The samples are from the product of the interpretation trees for the left and right shoulder joints and the left calf, for a frontal view similar to fig. 2. *Bottom:* Analogous distributions for Covariance Scaled Sampling (CSS) with scaling factor 8.

unevenly illuminated background, without special clothing or markers. Fig. 7 shows some frames from the original sequence (first row), 2D tracking results showing the current-best model configuration reprojected into the original image (middle row), and the corresponding 3D model pose rendered from a downwards-looking synthetic camera (bottom row). The tracks were initialized by running a method similar to that in §5.1, then selecting an initial set of 8 hypotheses that gave plausible initial body poses. From then on, the full sequence was tracked automatically using an observation likelihood function based on edge and intensity measurements as explained in §2. The sampling procedure was based on CSS diffusion (with scaling 4-6) followed by kinematic jump sampling with closed-form inverse kinematics. The selection of which kinematic sub-chain to sample at a given mode and time was done automatically using the local-uncertainty based voting mechanism described in §5. In this experiment the list $\mathcal{C}$ of user supplied chains contained the short 3-link chains associated with the neck, and each shoulder and each hip. For tracking, one usually needs a search process that does not wander too far from the given prior modes, and these chains have the advantage of generating shallow interpretation trees representing relatively probable local jumps or ambiguities. Such behavior is important not only for efficient and reliabile tracking, but also for the coherence of the post-tracking smoothing process, if any. (No smoothing was done here). The above settings prove highly effective in the sequence analyzed here, as can be seen from the model reprojection both in the original image, and as seen from above.

6 Conclusions

We have presented a novel kinematic sampling framework for recovering 3D human body motion from monocular video sequences. The cost surface for monocular human tracking is structured and highly multi-modal. For any feasible set of image joint positions, there are exponentially many 3D body configurations projecting to it. All of these have similar image projections, and they tend to have similar image likelihoods as well. The different 3D configurations are linked by 'forwards/backwards flipping' moves, one for each kinematic link. Our method uses simple inverse kinematics to systematically generate the complete set of such configurations given any one of them, and hence to investigate the full set of associated cost minima. Our experiments show that kinematic sampling complements and substantially improves on conventional random sampling based trackers, and that it can be used very effectively in tandem with them. The combined system is able to track short sequences involving fast, complex dancing motions in cluttered backgrounds.

Ongoing work is studing whether adding further physical scene constraints can improve the pruning of inconsistent samples, and also investigating the possibility of applying jump-based strategies for non-kinematic ambiguities such as image matching (*e.g.* 'right limb but wrong edge' correspondence errors) and within other MCMC algorithms. We also plan to make a more quantitative evaluation of our voting heuristic, and we are interested in developing smoothing algorithms that are better adapted to long range inter-frame dynamic moves.

Figure 7: Jump kinematics in action! Tracking results for a 4 s agile dancing sequence. *First row:* original images. *Middle row:* 2D tracking results showing the model-image projection of the best candidate configuration at the given time step. *Bottom row:* the corresponding 3D model configuration rendered from above. Note the difficulty of the sequence, the good model image overlap, and the realistic quality of 3D reconstucted model poses.

Acknowledgement Work supported by EU project VIBES.

References

[1] A. Barr. Global and Local Deformations of Solid Primitives. *Computer Graphics*, 18:21–30, 1984.

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

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

[4] J. Deutscher, A. Davidson, and I. Reid. Articulated Partitioning of High Dimensional Search Spacs associated with Articulated Body Motion Capture. In *CVPR*, 2001.

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

[6] D. Gavrila and L. Davis. 3-D Model Based Tracking of Humans in Action:A Multiview Approach. In *CVPR*, pages 73–80, 1996.

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

[8] H. J. Lee and Z. Chen. Determination of 3D Human Body Postures from a Single View. *CVGIP*, 30:148–168, 1985.

[9] J. MacCormick and M. Isard. Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracker. In *ECCV*, volume 2, pages 3–19, 2000.

[10] C. Samson, M. Borgne, and B. Espiau. *Robot Control. The Task Function Approach.* Oxford Science Publications, 1991.

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

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

[13] C. Sminchisescu. Consistency and Coupling in Human Model Likelihoods. In *FGR*, pages 27–32, Washington D.C., 2002.

[14] C. Sminchisescu and B. Triggs. Covariance-Scaled Sampling for Monocular 3D Body Tracking. In *CVPR*, volume 1, pages 447–454, Hawaii, 2001.

[15] C. Sminchisescu and B. Triggs. Building Roadmaps of Local Minima of Visual Models. In *ECCV*, volume 1, pages 566–582, Copenhagen, 2002.

[16] C. Sminchisescu and B. Triggs. Hyperdynamics Importance Sampling. In *ECCV*, volume 1, pages 769–783, Copenhagen, 2002.

[17] C. J. Taylor. Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image. In *CVPR*, pages 677–684, 2000.

[18] D. Tolani, A. Goswami, and N. Badler. Real-Time Inverse Kinematics Techniques for Anthropometric Limbs. *Graphical Models*, 62:353–388, 2000.