Avoiding the "Streetlight Effect": Tracking by Exploring Likelihood Modes

David Demirdjian, Leonid Taycher, Gregory Shakhnarovich, Kristen Grauman, and Trevor Darrell
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA, 02139

Abstract

Classic methods for Bayesian inference effectively constrain search to lie within regions of significant probability of the temporal prior. This is efficient with an accurate dynamics model, but otherwise is prone to ignore significant peaks in the true posterior. A more accurate posterior estimate can be obtained by explicitly finding modes of the likelihood function and combining them with a weak temporal prior. In our approach modes are found using efficient example-based matching followed by local refinement to find peaks and estimate peak bandwidth. By reweighting these peaks according to the temporal prior we obtain an estimate of the full posterior model. We show comparative results on real and synthetic images in a high degree of freedom articulated tracking task.

1. Introduction

Online articulated human tracking is the task of inferring (for each frame) the pose that both explains the observed image well, and is consistent with previous pose estimates and our notion of human motion dynamics. The human pose space is known to be large, making brute-force search methods infeasible.

Since the peaks in the compatibility function between images and pose are sharp [19], and dynamics are highly uncertain (except for very structured cases such as walking), a large number of hypotheses may have to be generated in order to locate the actual pose. When posed in probabilistic terms, the problem is the following: the pose likelihood is sharp but multi-modal, and the (dynamics-based) temporal prior is wide.

Looking under a streetlight to find a lost object at night is an apt metaphor for classic approaches to this task, which typically search within a region of the state space surrounding the estimate at a previous time step. It may not be where the object is, but it's an easy place to search! So goes the rationale of existing Bayesian tracking approaches, which base search on a strong temporal prior. In practice the "streetlight" (i.e., samples from the prior) can be narrow and bright (have high sample density), or be broad and dim (low density); neither is sufficient to find sharp peaks of the true posterior that are far from modes of the prior. Searching under the streetlight, i.e., under the prior, is seemingly desirable, but if the object is actually "in the dark" it is a futile endeavor.

Ideally we would like to evaluate the likelihood of a very broad and dense set of samples from the prior but this is impractical with existing probabilistic filtering methods. Broad search requires an extremely large number of samples, which are too costly to test and propagate individually. However, with a sharp likelihood and a wide prior the shape of the posterior distribution depends much more on the shape of the likelihood than on the temporal prior. Tracking performance may thus be improved by finding modes of the likelihood function first and incorporating prior information later.

In this paper we show how a broad search for modes of the likelihood function can proceed efficiently, mitigating the streetlight effect by considering regions of state space that appear highly likely based on the observation in the current frame. Whereas maintaining and propagating a very large set of samples representing a prior is impractical, we show how modes of the likelihood function can be sought efficiently using fast search methods.

We leverage the recent introduction of view-based or example-based methods [16, 11, 2], in which the dependency between the pose and body appearance is learned directly from large number of appearance/pose examples. Such methods can be used to quickly locate pose samples that are likely to be close to the modes of the likelihood functions. Local, gradient-based search can then find mode peaks, and estimate mode bandwidth. We are thus able to efficiently estimate the complete likelihood function as a mixture of a few Gaussians, each representing a narrow peak in the likelihood.

By reweighting these peaks according to the temporal prior we obtain an estimate of the full posterior model. In contrast to previous view-based tracking methods, our posterior accurately captures the multimodality of the likelihood function when appropriate. In contrast to previous sample-based methods it is able to search more broadly through the state space, rather than only around the prior

(or streetlight, to complete the metaphor).

In the following section we review relevant related work on probabilistic tracking. We then present our method for Exploring Likelihood MOdes (ELMO), and describe mode detection, refinement, and temporal integration in turn. We evaluate our approach with standard sequences from publicly available rendering software and motion capture data, as well as with real image sequences.

2. Prior Work

The core of our algorithm is the exploration of pose space by finding modes of the likelihood function, and weighting them by the prior to form an estimate of the posterior. Modes are estimated by initializing a model-based gradient-ascent algorithm at poses returned by a nearest neighbor matching algorithm.

Pose estimation algorithms often use gradient ascent to optimize the likelihood function (or pose-observation compatibility function in deterministic methods). Since likelihood modes are sharp, the initial hypothesis from which optimization is started is extremely important; gradient ascent is not likely to locate the mode if initialized far from it. Deterministic methods [14, 7, 8] use the previously estimated pose to start the search. While this is reasonable in situations with small interframe motion, such algorithms may lose track when fast motion or occlusion occurs.

While classic sampling-based probabilistic tracking algorithms [17, 15] only evaluate the likelihood function, recent approaches also use local optimization methods initialized at samples from the temporal prior [19, 9, 4]. The Hybrid Monte Carlo method of [5] incorporates gradient information directly into the sampling process. Since the temporal prior is obtained by propagating the pose posterior at the previous time step through the uncertain prior, many samples need to be drawn from it in order to get a good initialization point. The multi-hypothesis tracking approach of [4] is similar to ours in that only modes of the posterior (rather than individual samples) are propagated through dynamics, however it still requires sampling the propagated modes in order to obtain seeds for local optimization. Algorithms such as [20, 18] base their sampling method on the likelihood rather than the temporal prior, but still require generating and evaluating a large number of hypotheses.

As has been shown in [19], a local optimization is often only as good as its starting location, and the wide temporal prior is not the best source for pose samples that are close to a mode of the likelihood. Fortunately, several pose estimation methods have been recently developed that bypass using a human body model altogether. Instead they use a large number of view/pose pairs to directly learn the dependency between the image and the human pose. Relevance vector machine regression on the current observation and the previous pose estimate is used in [1] to find a mode of the posterior. The single-frame pose estimation algorithm of [16] uses parameter sensitive hashing to retrieve several samples with poses similar to the image, followed by robust regression. In [11], a mixture model prior over multi-view shape and pose is used to directly infer the unknown pose of an observed silhouette shape in a single frame.

3. Tracking with Likelihood Modes

We approach online pose estimation in video sequences as filtering in a probabilistic framework. The philosophy of our algorithm is based on two observations regarding the articulated tracking task. On the one hand, body dynamics are often uncertain so the temporal pose prior is wide – it assigns relatively large probability to large regions in the pose space. On the other hand, common likelihood functions (the compatibility between a rendered model and an observed image) are sharp, but multi-modal. A reasonable approximation to a sharply peaked multi-modal likelihood function is a weighted sum of Gaussians with small covariances.

Our algorithm, ELMO, proceeds as follows: we estimate modes of the likelihood function by selecting a set of initial pose hypotheses and refining them using a gradient-based technique which is able to both locate the mode of the likelihood and estimate its covariance. We obtain the temporal prior by propagating modes of the posterior computed at the previous time step through a weak dynamics model. Finally, we compute an estimate of the posterior distribution by reweighting the likelihood modes according to the temporal prior. An overview of the algorithm is shown in Figure 1.

In order for local optimization to succeed, it is important to select starting pose hypotheses that are sufficiently close to the modes. While it is possible to generate initial hypothesis from the wide temporal prior [19, 5, 17], or by uniformly sampling the pose space, in both of these methods a large number of samples would need to be drawn in order to obtain an hypothesis adequately close to the mode. Instead, we use a learning-based search method which, after being trained on a suitable number of image/pose examples, is able to quickly extract pose hypotheses that with high probability correspond to the observed image.

There are significant methodological differences between ELMO and classic particle filtering approaches. At no time is a density represented as a (large) set of samples, and so the need for a large number of likelihood evaluations is avoided. Furthermore, repeated instances of the same hypothesis do not imply a greater probability of that hypothesis. We do assume that at least one pose hypothesis will be extracted for each significant peak in the likelihood function. Thus a mode with low likelihood will have low weight

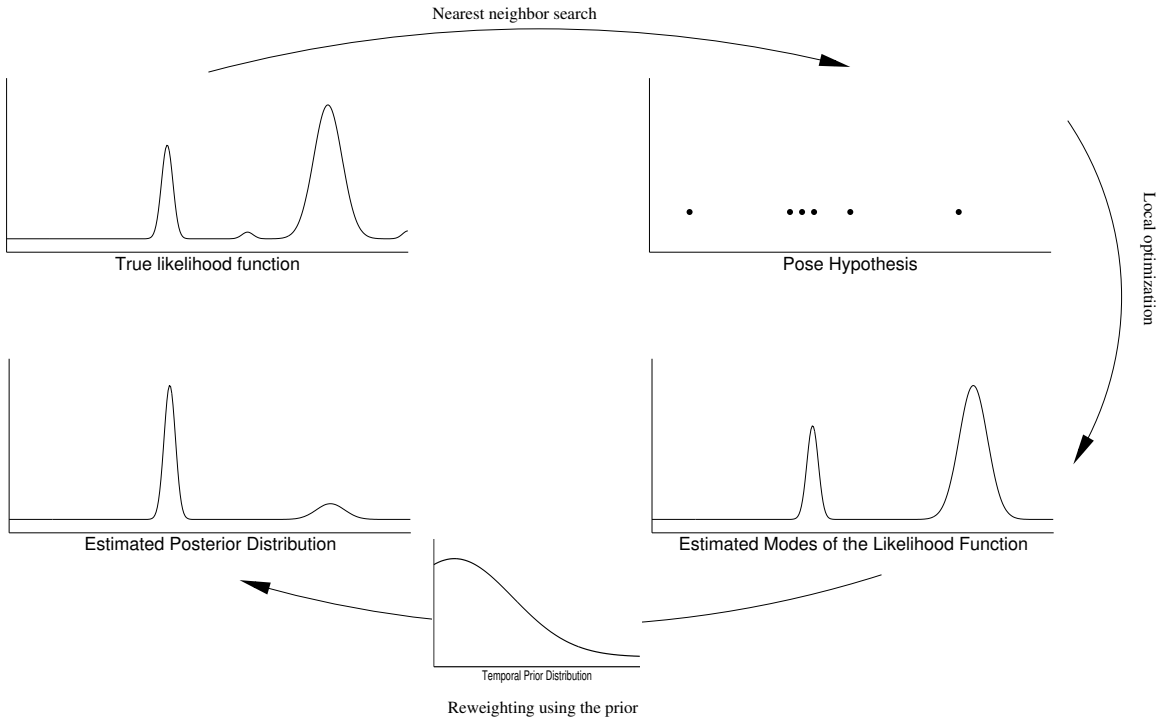

Figure 1: High-level overview of the ELMO algorithm. A set of pose hypotheses near the modes of the likelihood function are extracted using nearest neighbor search. The modes are refined with a gradient ascent algorithm initialized at every hypothesis, and a weighted sum of Gaussians estimate is computed for the likelihood function. Note that the number of hypotheses corresponding to a mode does not impact its estimated value. The posterior is then estimated by reweighting members of the mixture according to the temporal prior.

even if the gradient ascent algorithm converged to it from multiple starting hypotheses.

Since our algorithm is less reliant on the temporal prior for initializing search, it is likely to handle occlusions better then standard filtering methods. Indeed, ELMO can directly find the correct likelihood modes in the post-occlusion frames rather than starting with a (necessarily) wide prior.

3.1. Sampling with Parameter-Sensitive Hashing

A key component of our approach is the ability to quickly search the pose space for the small set of samples that lie close to the modes of the likelihood function. While there are a variety of fast regression or nearest neighbor search methods that are appropriate for our task, in this paper we rely on parameter-sensitive hashing (PSH) [16]. PSH is a randomized algorithm for the indexing and retrieval of data that allows very fast search of a large database of examples for instances similar to a query in a parameter space. In our case it means that from a database of images labeled with the corresponding articulated poses, we can quickly retrieve examples that with high probability have pose similar to the unknown pose in the input image. This is done by learning, from examples of images with similar and dissimilar poses, a set of hashing functions under which collision is correlated with pose similarity, rather than directly with appearance similarity.

Thus, the pose examples returned by PSH typically lie close to the modes of the likelihood function and should be an appropriate set of initial hypotheses for a local optimization algorithm even if the the training algorithm uses features different from those used to compute the likelihood. Furthermore, PSH is a modification of a locality-sensitive hashing algorithm [10] and shares its sublinear running time. Searching over tens of thousands of examples with PSH is orders of magnitude faster than propagating and evaluating an equivalent number of samples in a particle filter. As a result, the number of likelihood mode hypotheses that we can search is much larger than the number of samples that we could possibly maintain in a particle filter (as shown in the experiments below).

3.2. Local Optimization

We would like the likelihood $p(y|x)$ to represent the compatibility between the observed visual data y and the shape of a 3D articulated model corresponding to the pose x.

In this paper, visual observations y consist of calibrated stereo image pairs which are used to build a 3D reconstruction of the scene. The shape of the human body in pose x is given by a 3D articulated model $\mathcal{B}(x)$. Intuitively, the best fit $\hat{x}$ is obtained when the surface of the articulated model $\mathcal{B}(\hat{x})$ lies closest to the observed scene points. Therefore we define the likelihood $p(y|x)$ based on the distance between the articulated model and the observed scene. Such criteria has been commonly used for stereo-based tracking [3, 13]. In the case of monocular data, an adequate likelihood model could be defined [17] by the reprojection error of the 3D articulated model onto the images.

Let $\mathcal{M}(y) = \{M_i(y)\}$ be the set of 3D points of the scene reconstructed from the stereo image pair. Let $\{N_j(x)\}$ be a set of sample points from the articulated model $\mathcal{B}(x)$. In practice, the distance $d(\mathcal{M}(y), \mathcal{B}(x))$ between the scene points and the articulated model can be written as:

$$d^2(\mathcal{M}(y), \mathcal{B}(x)) = \sum_j d_E^2(\mathcal{M}(y), N_j(x)) \qquad (1)$$

where $d_E^2()$ is the Euclidean distance between the point cloud $\mathcal{M}(y)$ and the point $N_j(x)$.

A likelihood model $p(y|x)$ naturally follows as:

$$p(y|x) \propto \exp\{-\lambda d^2(\mathcal{M}(y), \mathcal{B}(x))\} \qquad (2)$$

where λ a parameter depending on the uncertainty of the 3D reconstruction.

Given a set of pose hypotheses returned by PSH and mode locations propagated from the previous time step, we fit a sum of Gaussians (3) to the approximate likelihood at time t, $p(y^t|x^t)$ defined in eq.(2).

We apply a local search algorithm using initializations $\{x_{init}\}$ from both the centers of the modes μ_i^{t-1} of the likelihood $p(y^{t-1}|x^{t-1})$ at the previous time step as well as pose estimates provided by a global search algorithm such as PSH. For each initialization x_{init_k}, we look for a local maximum μ_k^t (with covariance C_k^t) of $p(y^t|x^t)$. In many cases, the local optima μ_k^t converge to the same peaks of the likelihood $p(y^t|x^t)$. Only the highest optima (μ_k^t,C_k^t) are kept to represent the full likelihood model $p(y^t|x^t)$. In practice, an average of 5 modes is usually kept.

The local optimum μ_k can be found using standard optimization techniques such as gradient ascent or Levenberg-Marquardt. However, in the particular case of likelihood functions based on a 3D metric error such as $d^2(\mathcal{M}(y), \mathcal{B}(x))$, approximative techniques such as those based on the Iterative Closest Point (ICP) algorithm [3] can be used in order to estimate the optimum μ_k and covariance C_k (see [7, 8]). Such algorithms are proven to converge (when initialized close to the solution) and are less computationally intensive than standard optimization techniques.

3.3. Temporal Integration

In typical articulated tracking tasks, as discussed above, the temporal prior provides less information about the posterior distribution than the likelihood function. Given a sum of Gaussians representation of the likelihood function, we show here how to efficiently integrate information over time and estimate an instantaneous posterior.

A key challenge when propagating mixture models is the combinatorial complexity cost. Indeed, if the posterior distribution at the previous time step (and thus the temporal prior, as we assume simple diffusion dynamics) is estimated as a mixture of K Gaussians, and the likelihood is a sum of L Gaussians, then it is reasonable to expect that the posterior estimate at the current time step will be a mixture of $L \times K$ Gaussians. We will show, however, that when the temporal prior is wide (i.e. the noise covariance is much greater than the covariance of the likelihood modes), then the estimate of the posterior may be obtained simply by modifying the weights of the likelihood Gaussians according to the prior.

Let y^t be the observation at time t, and x^t be the pose. Let the pose likelihood and temporal prior be

$$p(y^t|x^t) = \sum_{i=1}^{L} \hat{w}_i^t N(x^t; \mu_i^t, C_i^t),) \qquad (3)$$

$$p(x^t|y^0, y^1, \ldots, y^{t-1}) = \sum_{j=1}^{K} w_j^{t-1} N(x^t; \mu_j^{t-1}, C_j^{t-1} + C_\eta) \qquad (4)$$

$$\text{where } N(x; \mu, C) = \frac{1}{\sqrt{(2\pi)^D |C|}} e^{-(x-\mu)^T C^{-1} (x-\mu)}.$$

The ith mode in the likelihood has mean μ_i^t, covariance C_i^t and value $\frac{\hat{w}_i^t}{\sqrt{(2\pi)^D |C_i^t|}}$. Each component of the temporal prior has arisen from the posterior modes estimated at the previous time step (characterized by means μ_j^{t-1}, covariances C_j^{t-1} and weights w_j^{t-1}) after combination with Gaussian noise with covariance C_η.

In general the posterior distribution $p(x^t|y^0, y^1, \ldots, y^t) \propto p(y^t|x^t) p(x^t|y^0, y^1, \ldots, y^{t-1})$ would be a mixture of $L \times K$ terms of the form $N(x^t; \mu_i^t, C_i^t) N(x^t; \mu_j^{t-1}, C_j^{t-1} + C_\eta)$. Each such product can be expressed as:

$$\begin{aligned} &N(x^t; \mu_i^t, C_i^t) N(x^t; \mu_j^{t-1}, C_j^{t-1} + C_\eta) \\ &\quad = k N(x^t; \hat{\mu}_i, \hat{C}_i), \text{where} \\ &k = N(\mu_i^t; \mu_j^{t-1}, C_i^t + C_j^{t-1} + C_\eta) \\ &\hat{C}_i = ((C_i^t)^{-1} + (C_j^{t-1} + C_\eta)^{-1})^{-1} \\ &\hat{\mu}_i = \hat{C}_i ((C_i^t)^{-1} \mu_i^t + (C_j^{t-1} + C_\eta)^{-1} \mu_j^{t-1}) \end{aligned}$$

Since we assume that the noise covariance is much

greater that covariance of the likelihood modes, the following is true:

$$C_i^t + C_\eta \approx C_\eta$$
$$(C_i^t)^{-1} + (C_\eta)^{-1} \approx (C_i^t)^{-1}$$

The product can be approximated as

$$N(x^t;\mu_i^t, C_i^t)N(x^t;\mu_j^{t-1}, C_j^{t-1} + C_\eta) \approx \quad (5)$$
$$N(\mu_i^t;\mu_j^{t-1}, C_\eta)N(x^t;\mu_i^t, C_i^t)$$

and the posterior distribution is reduced to

$$p(x^t|y^0, y^1, \ldots, y^t) \approx \frac{1}{\sum_i^L w_i^t} \sum_{i=1}^{L} w_i^t N(x^t;\mu_i^t, C_i^t), \quad (6)$$
$$w_i^t = \hat{w}_i^t \sum_{j=1}^{K} w_j^{t-1} N(\mu_i^t;\mu_j^{t-1}, C_\eta)$$

Intuitively, we can expect that the wide temporal prior does not vary much over the region of support of each Gaussian in the likelihood, and the posterior distribution is then the mixture of the same Gaussians but with their weights modified by the probabilities assigned to their means by the temporal prior.

4. Implementation and Experiments

In order to validate our approach, we performed various experiments to compare our algorithm (ELMO) against both its component algorithms PSH and ICP, as well as the particle filtering method Condensation [12].

The feature space over which PSH hash functions were constructed consisted of concatenated multiscale edge direction histograms (EDH) as in [16]. The EDH of an image is computed by applying an edge detector, assigning each edge pixel to one of the fixed directional bins (four in our case), counting the number of edge pixels for each direction falling in each of a number of subwindows of various sizes taken at various locations, and finally concatenating the obtained counts in a single feature vector. For images of 200 by 200 pixels used in our database, with 3 scales (8, 16 and 32 pixels) and with location step size of half the scale, the EDH consisted of $N = 13,076$ bins. We then selected $M = 3547$ features for which the true-positive rate [16] was above 0.65 and the true-position/false-positive gap was at least 0.1. The data were then indexed by $l = 50$ hash tables with $k = 18$ bit keys. For every frame, we retrieve $K = 50$ training examples and use their poses to initialize the ICP.

The labeled pose database indexed by PSH in our system consists of 60,000 images of humanoid models in randomly sampled poses created with Poser [6]. The models were constrained to an upright posture, but the articulation in the upper limbs as well as the orientation of the torso was constrained only by anatomical feasibility. We rendered the images from a viewpoint consistent with the camera settings of the tracker, and for each image saved the articulated pose information (3D locations of key body joints: neck, shoulders, elbows etc.). Pose similarity when training PSH was defined as less than 5 cm difference between any two joints.

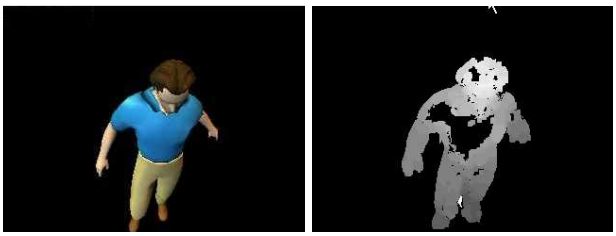

Figure 3: Example of color and disparity images used in the synthetic sequences.

4.1. Synthetic Sequences

The first set of experiments evaluates the ground truth error relative to an extensive set of synthetic sequences.

Testing data consisted of a collection of synthetic sequences of people performing various kinds of activities (e.g. walking, playing sports, greeting). The synthetic sequences were generated from motion capture data taken from a public website[1] and rendered using Poser [6] to produce stereo image pairs. Then, standard correlation-based stereo was performed on the image pairs to produce a "realistic" disparity image as shown in Figure 3.

Some of the sequences contain many challenges for articulated tracking algorithms, including perspective effects (e.g. images taken from a 45 degree angle, hands moving very close to the camera), multiple self-occlusions (e.g. body turned on the side, completely hiding one of the arms), partial visibility (e.g. arms out of the field of view of the camera) and fast motions. Also note that the synthetic sequences have been rendered with characters and features different from the ones used in the PSH training set.

The synthetic sequences' images were used as input for the Condensation, PSH, ICP, and ELMO algorithms. The Condensation algorithm was implemented as described in [12] and run using $N = 1000$ particles. We use the same likelihood function for Condensation and ELMO. The PSH and ICP algorithms were implemented following [16][2] and [8] respectively. We fixed the number of candidates returned by PSH to 50 and computed the pose as the candidate with highest likelihood. Note that in order to run the ICP and Condensation algorithms, the articulated model

[1] `http://www.mocapdata.com`

[2] Except that we omitted the local regression step.

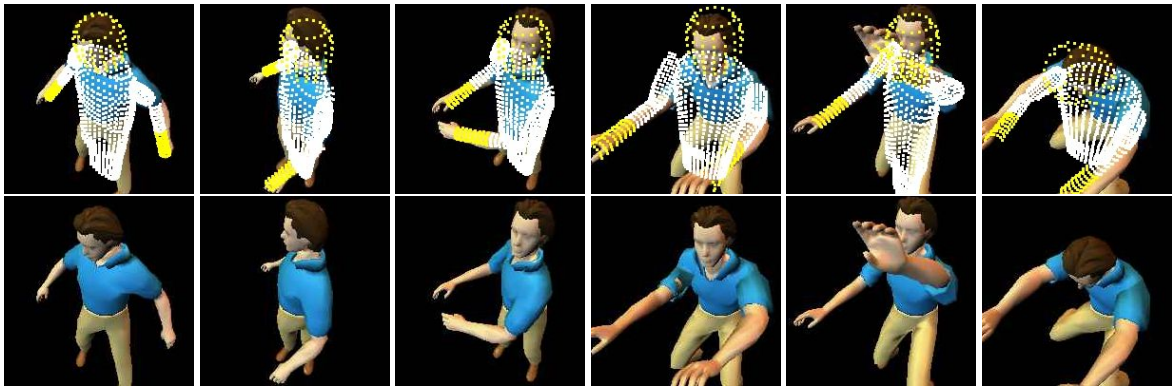

Figure 2: Tracking results extracted from the synthetic test sequences. These images show clearly the complexity of the motions and the challenge for articulated tracking (e.g. perspective effect, self-occlusions).

was manually initialized at the beginning of the sequences. For each algorithm, the pose estimation was compared to the ground truth. The average ground truth errors per joint over all the sequences are reported in Figure 4. Our algorithm outperforms the Condensation, ICP and PSH algorithms by having a smaller error and variance (the average error is about 5 cm) and by automatically initializing tracking. As shown in Table 1, which shows the run time per frame of each algorithm in this experiment, ELMO is significantly faster than Condensation.

Figure 5 shows the variation of the average error over time in three sequences. The error corresponding to the ELMO algorithm is almost always smaller than that of the other algorithms. In sequences corresponding to poses with few challenges (e.g. all limbs visible, small motions), ICP and ELMO give similar results. However, in harder sequences, such as the "bye" sequence, the ICP algorithm eventually loses track after following an incorrect local optimum of the likelihood function. This can be seen in the graph by a sudden increase of the error function around frame 300.

We note that the pose estimation from PSH seems to have been biased by some implementation issues. First PSH was trained on a domain more restricted than the testing sequences (e.g. the examples in the PSH training set did not include persons with a bent torso or bent legs such as Figure 2. This explains, for instance, the large errors obtained for the "karate" sequence, which contains multiple images of a person bending the torso and legs. PSH also seems to have a constant offset error due to a misalignment and scaling between the PSH referential and the coordinate system used to estimate the likelihood $p(y|x)$. In spite of implementation deficiencies, the results show that PSH is still able to provide good initialization for a local search of the optimum of the likelihood.

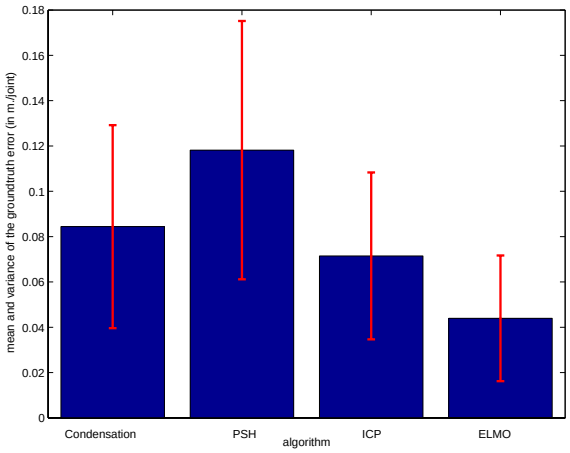

Figure 4: Average and standard deviation of the ground truth error obtained using Condensation, PSH, ICP, and ELMO on six sequences of 1000 images each. Our algorithm outperforms the Condensation, ICP and PSH algorithms. The average error per joint for ELMO is less than 5 cm.

Condensation (1000 particles)	PSH	ICP	ELMO
120	1	0.1	2

Table 1: Amount of time required by the algorithms to process a single frame (in sec.)

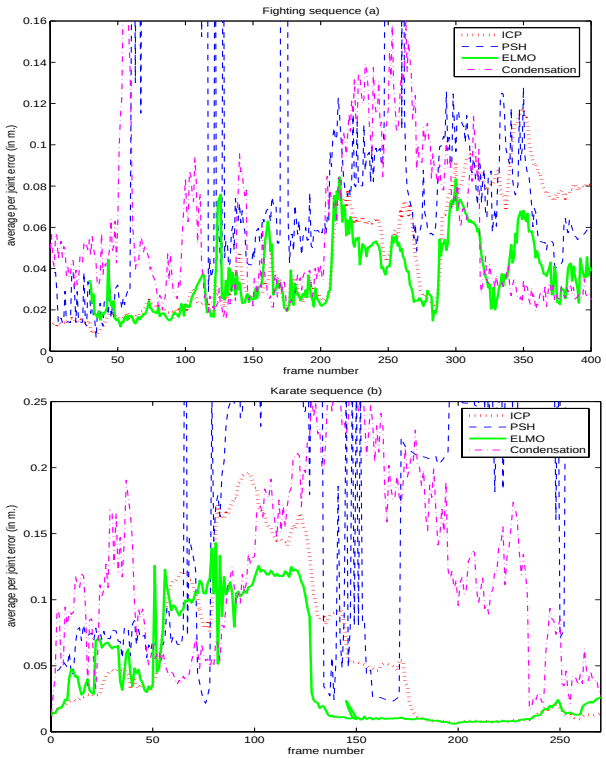

Figure 5: Tracking results on two of the six test sequences (for better clarity, only segments of the sequences are shown). The graphs report the ground truth error (vs. frame number) corresponding to Condensation, PSH, ICP and ELMO. (a) Fighting sequence, (b) Karate sequence. The error corresponding to the ELMO algorithm is almost always the smallest.

4.2. Laboratory Sequences

In order to further validate our approach, we also collected real sequences of people moving in front of a stereo camera and used them as input for the ELMO algorithm. Figures 6 and 7 show tracking results obtained with the ELMO algorithm on two sequences. In the first sequence, a person is performing dance moves; in the second one, a person is standing in front of a whiteboard and explaining a diagram to a virtual audience. Both sequences were recorded at a slow frame rate (less than 4 Hz), producing large image motions between consecutive images. The reconstruction of the 3D articulated model shows the good quality of pose estimation provided by ELMO, in spite of the difficulty of the sequences (e.g. large motions, complex poses).

The ELMO algorithm has very low computational complexity, and should be implementable in real time. For each frame, ELMO takes about one second to obtain up to 50 hypotheses with PSH and to perform local optimization with typically 50 to 60 initial hypotheses. Since the number of modes estimated at every time step is small, the temporal integration cost is negligible.

5. Conclusions

We have presented ELMO, a method for tracking articulated human bodies by exploring likelihood modes. Likelihood mode search is made feasible by a fast approximate nearest neighbor method; the modes are further refined by a local optimization method that estimates mode location as well as bandwidth. An approximate posterior distribution is computed with an efficient mode reweighting scheme. In contrast to classic sampling approaches, our method can explore a much larger region of the pose space since searching a vast number of examples with an approximate nearest neighbor search and refining a few modes is much more efficient than maintaining a particle set of a sufficient size. On real and synthetic sequences containing challenging body motions ELMO outperformed local optimization, view-based search approaches, and Condensation.

References

[1] A. Agarwal and B. Triggs. Learning to Track 3D Human Motion from Silhouettes. In *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, July 2004.

[2] V. Athitsos and S. Sclaroff. Database Indexing Methods for 3D Hand Pose Estimation. In *Gesture Workshop*, April 2003.

[3] P.J. Besl and N. MacKay. A Method for Registration of 3D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256, Feb. 1992.

[4] T.-J. Cham and J. Regh. A Mutiple Hypothesis Approach to Figure Tracking. Technical report, Compaq Cambridge Research Laboratory, 1998.

[5] K. Choo and D. Fleet. People Tracking Using Hybrid Monte Carlo Filtering. In *Proceedings of the IEEE International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[6] Curious Labs, Inc., Santa Cruz, CA. *Poser 5 - Reference Manual*, 2002.

[7] Q. Delamarre and O. D. Faugeras. 3D Articulated Models and Multi-View Tracking with Silhouettes. In *Proceedings of the IEEE International Conference on Computer Vision*, Corfu, Greece, Sept. 1999.

[8] D. Demirdjian, T. Ko, and T. Darrell. Constraining Human Body Tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003.

[9] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2000.

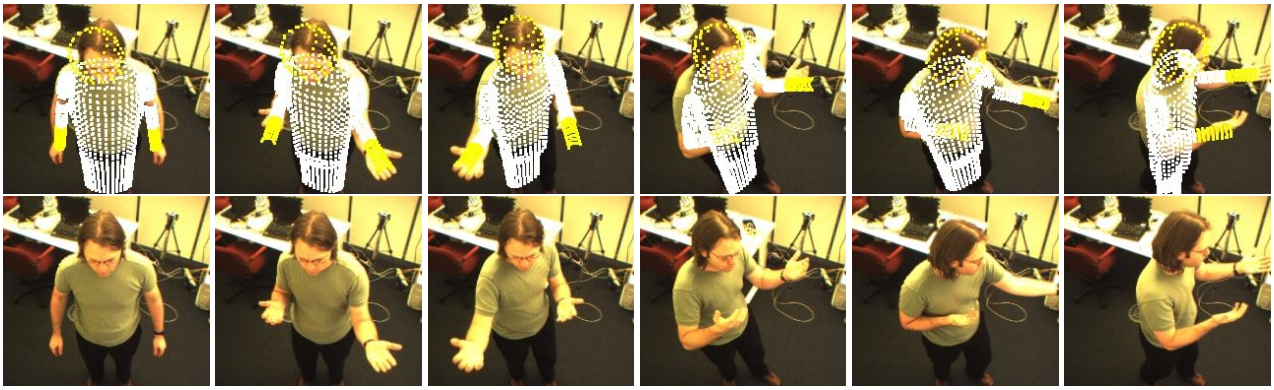

Figure 6: Tracking results extracted from the *dance* sequence.

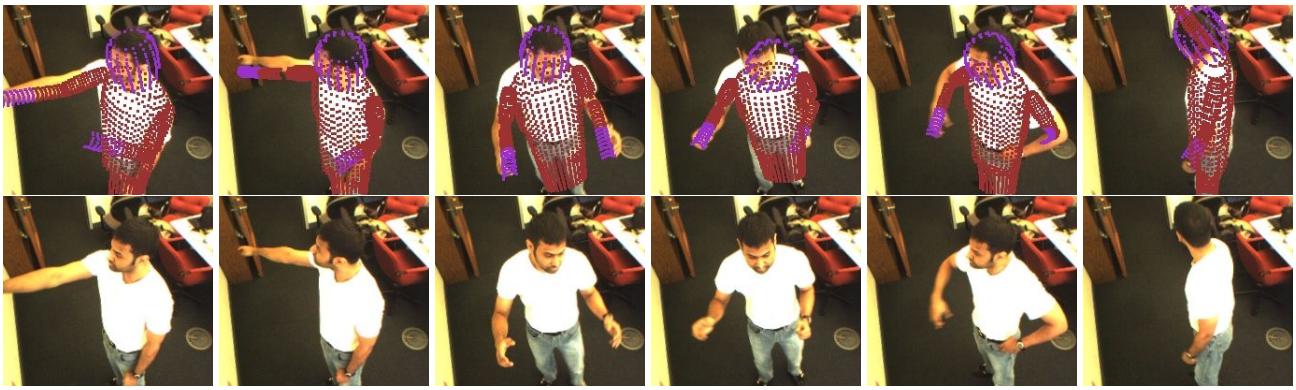

Figure 7: Tracking results extracted from the *whiteboard* sequence.

[10] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. In *The VLDB Journal*, pages 518–529, 1999.

[11] K. Grauman, G. Shakhnarovich, and T. Darrell. Inferring 3D Structure with a Statistical Image-Based Shape Model. In *Proceedings of the IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003.

[12] M. Isard and A. Blake. Condensation – Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision*, 29:5–28, 1998.

[13] N. Jojic, M. Turk, and T.S. Huang. Tracking Articulated Objects in Dense Disparity Maps. In *Proceedings of the IEEE International Conference on Computer Vision*, Corfu, Greece, Sept. 1999.

[14] R. Plänkers and P. Fua. Articulated Soft Objects for Video-Based Body Modeling. In *Proceedings of the IEEE International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[15] K. Rohr. Towards Models-Based Recognition of Human Movements in Image Sequences. *CVGIP*, 59(1):94–115, Jan 1994.

[16] G. Shakhnarovich, P. Viola, and T. Darrell. Fast Pose Estimation with Parameter-Sensitive Hashing. In *Proceedings of the IEEE International Conference on Computer Vision*, Nice, France, Oct. 2003.

[17] H. Sidenbladh, M. Black, and D. Fleet. Stochastic Tracking of 3D Human Figures Using 2D Image Motion. In *Proceedings of the European Conference on Computer Vision*, Dublin, Ireland, June 2000.

[18] L. Sigal, M. Isard, B. Sigelman, and M. Black. Attractive People: Assembling Loose-Limbed Models using Non-Parametric Belief Propagation. In *Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2003.

[19] C. Sminchiesescu and B. Triggs. Estimating Articulated Human Motion with Covariance Scaled Sampling. *International Journal on Robotics Research*, 22:371–391, June 2003.

[20] L. Taycher and T. Darrell. Bayesian Articulated Tracking Using Single Frame Pose Sampling. In *Proc. 3rd Int'l Workshop on Statistical and Computational Theories of Vision*, Nice, France, Oct 2003.