Sequential Monte Carlo Tracking of Body Parameters in a Sub-Space

Thomas B. Moeslund and Erik Granum
Laboratory of Computer Vision and Media Technology
Aalborg University, Denmark
E-mail: tbm@cvmt.dk

Abstract

In recent years Sequential Monte Carlo (SMC) methods have been applied to handle some of the problems inherent to model-based tracking. In this paper two issues regarding SMC are investigated in the context of estimating the 3D pose of the human arm. Firstly, we investigate how to apply a sub-space to representing the pose of a human arm more efficiently, i.e., reducing the dimensionality. Secondly, we investigate how to apply a local method to estimated the maximum a posteriori (MAP). The former issue is based on combining a screw axis representation with the position of the hand in the image. The latter issue is handled by applying a method based on maximising a proximity function, to estimate the MAP. We find that both the sub-space and the proximity function are sound strategies and that they are an improvement over the current SMC-methods.

1. Introduction

The problem of recognising human body language has for some years been in focus in the computer vision community. The obvious reason is the vast amount of HCI applications that requires recognised body language as input, but also the complexity of the involved subproblems draws researchers to this problem domain. Traditionally two different approaches have been taken [12]. Either the recognition is done directly on the (pre-processed) image data, or the pose of the body is firstly captured and *then* recognition is done on these high-level pose parameters. The latter approach has received the most attention and especially by using model-based tracking [12].

Model-based tracking algorithms have produced impressive results, but still no model-based system (or any other system for that matter) has been reported that estimates the body pose parameters in unconstrained environments and for long periods of time. There are many reasons for this, but one of the primary reasons related to model-based tracking is the high dimensionality of the solution space that often appears. By other things, this means that an exhaustive search seldom is practical. Instead, prediction followed by either an iterative search, a Kalman Filter, or an exhaustive search in the proximity of the prediction, is applied. The drawback of these approaches is the risk of ending up in a local extremum, i.e., estimating the wrong state. In recent years, statistical methods such the Condensation algorithm [8, 17], the particle filter [14, 16], and Multiple-Hypothesis tracking [3, 4] have therefore been applied to approximate an exhaustive search, or in statistical terms - approximate the posterior probability density function (PDF). These methods all belong to the class of Sequential Monte Carlo (SMC) methods [6].

A Monte Carlo method represents the posterior by a finite number of weighted state samples (known as particles) each selected from an Importance Function and weighted by the measurements. This sampling principle is known as Importance Sampling. An SMC method is a Monte Carlo method operating on a time sequence of measurements. Here the Importance Function can be defined by predicting the posterior from the previous time instant. In other words, each of the most likely states in the posterior in the previous time instant is sampled, predicted into the current time instant, and compared with the current image in order to obtain a weight. The weight reflects the similarity between the predicted state and the image measurements, i.e., the likelihood. The predicted states and their associated weights define the estimate of the posterior in the current time instant. The current state of the system is defined as the maximum a posteriori (MAP).

How effective the principle of SMC works depends on at least two issues: i) the number of particles used to estimate the posterior PDF, and ii) the MAP estimate.

The number of particles, N, can be tuned to a particular application or even changed during processing. In general N should be kept as low as possible since the computational demands of the algorithm growth exponentially with respect to N [5]. In fact N increases with both the dimensionality of the state-space and the covariance of the posterior PDF.

The other issue in SMC is the quality of the MAP, that is, how the state of the tracked object is estimated at a particular time instant. Since the estimate of the posterior PDF is in the form of N weighted particles the representation is obviously non-parametric. The MAP is therefore estimated via moments, i.e., the mean and covariance of the posterior

PDF. This works well if the posterior PDF is uni-modal, but as one of the key notions behind applying SMC in tracking is to allow multiple-hypotheses, moments might not be the best choice!

1.1. The Content of this Paper

In this paper we try to deal with the two above mentioned problems related to SMC tracking; reducing the number of particles and defining a better way of estimating the MAP. The former problem is dealt with by tracking in a sub-space which requires fewer particles due to a dimensionality reduction of the solution space. The latter is dealt with by suggesting that the estimate of the MAP is done locally as opposed to globally. The context of the tracking problem is to estimate the 3D pose of a human arm via model-based and monocular computer vision. We see this context as a subproblem of the more general problem of estimating the entire human body and we therefore assume that the 3D position of the shoulder is known in advance.

Concretely the paper is structured as follows. In section 2 we define the geometric model of the arm utilised in this work, i.e., the state-space and derive our sub-space. In section 3 we describe how the observation PDF in the SMC algorithm is defined in this work. In section 4 we describe how the prediction in the SMC algorithm is carried out. In section 5 we estimate the MAP. Section 6 presents the results and section 7 discusses our findings.

2. The State-Space

In this section we deal with the first SMC-related problem described above, namely how to reduce the number of particles, N. We assume that reducing the dimensionality of the state-space is equivalent to reducing N and the focus of this section is therefore to find a sub-space wherein the tracking of the body parameters can be carried out.

Our state-space is spanned by the different parameters used to model the arm. The total number of different arm configurations is equal to the number of discrete points in the state-space, given some resolution of each parameter.

The human arm is usually modelled as either the 3D positions of the elbow and hand, or by four angles together with the length of the upper arm (A_u) and the lower arm (A_l). Both representations require six parameters. If we, however, assume the length of the two arm segments to be known, we only require four angular parameters. These can be, e.g., angles around fixed axes or Euler's angles. The latter is often applied as it is similar to the anatomic joint angles of the arm. Ignoring the kinematic constraints of the arm the Euler's angles representation has a state-space with around $1.68 \cdot 10^{10}$ different configurations given an angular resolution of one degree in each dimension.

An alternative approach is to apply the screw axis representation. It is not directly related to the anatomic joints, but nevertheless has the same ability to represent all the different arm configurations as the Euler's angles have. This representation is applied in robotics and computer graphics and recently also in computer vision [2][10][11].

The representation is based on Chasles' theorem [15] which loosely states that a transformation between two coordinate systems can be expressed as a rotation around an axis, called the screw axis (or helical axis), and a translation parallel to the screw axis. In the context of modelling the human arm, we define the screw axis as the vector spanned by the shoulder and the hand. The position of the elbow is defined as a rotation, α, of an initial elbow position around the screw axis [1]. As the length of the upper and lower arm are fixed no translation is required parallel to the screw axis, and the perpendicular distance from the elbow to the screw axis is independent of α and can be calculated without adding additional parameters. Altogether the representation requires four parameters. Three for the position of the hand, H_x, H_y, and H_z, to define the screw axis and one for the rotation around the screw axis, α. The parameters are illustrated in figure 1.

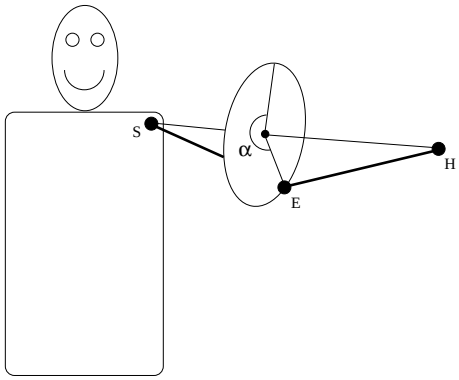

Figure 1: The screw axis representation of the human arm. α is the angle between the top point on the circle and the actual elbow position. S, E, and H represent the 3D shoulder, elbow, and hand positions, respectively.

2.1. Generating the Sub-Space

Little is gained so far in terms of achieving a more compact state-space, i.e., a sub-space. To generate a sub-space we introduce two assumptions. Firstly, we assume that we are working with a calibrated camera. Secondly, we assume we can always locate the hand in the image by utilising skin-color segmentation[1]. Combining the two assumptions we

[1]When this assumption do not hold, our tracker do not break down. Instead, we continue tracking in another space spanned by the four Euler's angles until the assumption holds again.

can map the position of the centroid of the hand found in the image to a line, l, in space passing through the hand. In other words, given the position of the hand in the image, we only require one free parameter to represent its 3D position, namely the displacement, d, along the vector spanned by the position of the hand in the image and the focal point. For each value of d the position of the hand $\overrightarrow{H}$ is defined. This means that we might as well use any of the entries in $\overrightarrow{H}$ to represent the free parameter. In this work we use H_z as the free parameter. So, for each value of H_z d and therefore also H_x and H_y are uniquely determined. By applying this concept to the screw axis representation we can eliminate the parameters H_x and H_y which leaves us with just two parameters, namely α and H_z to model the configuration of the arm.

In this sub-space α is bounded by one circle-sweep ($0^\circ - 360^\circ$) while H_z is bounded by $\pm$ the total length of the arm. Given a total arm length of say $60cm$ and a resolution of 1° for α and $1cm$ for H_z we have a solution space containing $4.32 \cdot 10^4$ different solutions. A large reduction in the size compared to that produced by Euler's angles, see section 2, but still a too large space for, e.g., an exhaustive search.

3. Observation PDF in SMC Tracking

The SMC algorithm is defined in terms of Bayes' rule and by using the first order Markov assumption. That is, the posterior PDF is equal to the observation PDF multiplied by the prior PDF, where the prior PDF is the predicted posterior PDF from time $t-1$:

$$p(\overrightarrow{X_t}|\overrightarrow{\theta_t}) = p(\overrightarrow{\theta_t}|\overrightarrow{X_t})p(\overrightarrow{X_t}|\overrightarrow{\theta_{t-1}}) \quad (1)$$

where $\overrightarrow{X}$ is the state, hence $\overrightarrow{X} = [\alpha, H_z]^T$ and $\overrightarrow{\theta}$ contains the image measurements. The predicted posterior PDF is defined as

$$p(\overrightarrow{X_t}|\overrightarrow{\theta_{t-1}}) = \int p(\overrightarrow{X_t}|\overrightarrow{X_{t-1}})p(\overrightarrow{X_{t-1}}|\overrightarrow{\theta_{t-1}})\, \mathrm{d}\overrightarrow{X_{t-1}} \quad (2)$$

where $p(\overrightarrow{X_t}|\overrightarrow{X_{t-1}})$ is the motion model governing the dynamics of the tracking process, i.e., the prediction, and $p(\overrightarrow{X_{t-1}}|\overrightarrow{\theta_{t-1}})$ is the posterior PDF from the previous frame. The SMC algorithm estimates $p(\overrightarrow{X_t}|\overrightarrow{\theta_t})$ by selecting a number, N, of (hopefully) representative states (particles) from $p(\overrightarrow{X_{t-1}}|\overrightarrow{\theta_{t-1}})$, predicting these using $p(\overrightarrow{X_t}|\overrightarrow{X_{t-1}})$, and finally giving each particle a weight in accordance with the observation PDF.

The observation PDF, $p(\overrightarrow{\theta_t}|\overrightarrow{X_t})$, expresses how alike each state and the image measurements are. In this work the image measurements are the probability of the orientations of the upper and lower arm in the image, respectively, i.e., $\overrightarrow{\theta_t} = [p_u(\theta_u), p_l(\theta_l)]^T$, where $p_u(\theta_u)$ and $p_l(\theta_l)$ are the PDFs of the different orientations of the upper arm and lower arm, respectively. We can now define the observation PDF as,

$$p(\overrightarrow{\theta_t}|\overrightarrow{X_t}) = p_u(\theta_u(\overrightarrow{X_t})) + p_l(\theta_l(\overrightarrow{X_t})) \quad (3)$$

where $\theta_u(\overrightarrow{X_t})$ and $\theta_l(\overrightarrow{X_t})$ map from our representation of the arm, $[\alpha, H_z]^T$, to the orientation of the upper and lower arm in the image, respectively[2].

3.1. Estimating the PDFs of the Orientations of the Arm in the Image

We estimate the PDFs of the orientations of the upper arm, $p_u(\theta_u)$, and lower arm, $p_l(\theta_l)$, respectively, based on edge pixels. As our input images contain background clutter and non-trivial clothes we utilise temporal edge pixels. That is, we find the edge pixels in the current image using a standard edge detector and AND this result with the difference image achieved by subtracting the current- and the previous image. Figure 2.A shows a typical input image. In figure 2.B the temporal edge pixels for this particular image are shown. Those pixels actually belonging to the arm will be located in four classes, two for the upper arm and two for the lower arm, respectively. Our system does not impose restrictions on the clothes of the user. The clothes will in general follow gravity, hence the two classes of pixels originating from the upper sides (with respect to gravity) of the upper- and lower arm will model the structure of the arm better, see figure 2.B. We therefore only consider temporal edge pixels located on the "upper" sides. Concretely we define "upper" and "lower" via two lines described by the position of the shoulder and hand in the image, together with a predicted position of the elbow.

As we wish to estimate $p_u(\theta_u)$ and $p_l(\theta_l)$ independently we separate the temporal edge pixels into two groups, one for the upper arm and one for the lower arm. This is done by calculating the perpendicular distance from each pixel to the two lines. As the prediction of the position of the elbow is uncertain we ignore all pixels within a certain distance from the predicted position of the elbow. Furthermore, we ignore all pixels too far away from both lines. When no predictions are available different possible positions of the predicted elbow are investigated until two representative groups are obtained.

Estimating the orientation of a straight line from data can be carried out in different ways, e.g., via principal component analysis or linear regression. However, as we will not model the distribution of the orientations via Gaussians we can not apply these methods. Instead we apply a dynamic variant of the Hough transform - the dynamic Hough transform (DHT). It estimates the likelihood of each possible

[2]These mappings require the camera parameters as well. But to enhance the concept we have left them out in the expressions.

A B

Figure 2: A: A typical input image (shown in B/W). B: The temporal edge pixels.

orientation, hence allowing multiple peaks in the observation PDF. The choice of the DHT is furthermore motivated by the fact that it adapts to the data. The DHT randomly samples two pixels from one group and calculates the orientation of the line spanned by the two pixels. The more times the groups are sampled the better the estimation of the PDFs. On the other hand many samplings also lead to large processing time as is the case for the standard Hough Transform. The sampling of one group is therefore terminated as soon as the variance of the PDF is stable. To evaluate the stability of the variance after n samplings the variance of the last j variances is calculated as

$$\nu_{jn}^2 = \frac{1}{j}\sum_{i=n-j}^{n}(\sigma_i^2 - \mu_{jn})^2 \tag{4}$$

where σ_i^2 is the variance after i samplings and μ_{jn} is the mean of the last j variances.

The stop criterion is defined as the number of samplings, n, where the last j samplings are within the interval $[\mu_{jn} - \lambda, \mu_{jn} + \lambda]$. The distribution of the last j variances will in general follow a Uniform distribution. The theoretical variance of such a distribution in the given interval can be estimated as $\lambda^2/12$ [13]. When the mean of the variances, μ_{jn} is large it indicates large uncertainty in the PDF, which again indicates weak lines in the temporal edge image. A stable variance for such a PDF tends to required a larger value of λ compared to an image with stronger lines. To account for this difference λ is defined with respect to μ_{jn} as

$$\lambda = \frac{\mu_{jn}}{\gamma} \tag{5}$$

where γ is found empirically. Setting the estimated variance equal to the theoretical variance yields $\lambda = \nu_{jn}\sqrt{12}$. Inserting this result into equation 5 and writing it as an inequality yields

$$\nu_{jn}^2 \leq \frac{\mu_{jn}^2}{12 \cdot \gamma^2} \tag{6}$$

Altogether the stop criterion is found as the smallest n for which inequality 6 is true. To speed up the calculations the variance is not recalculated after each new sampling, but rather for every 10th sampling.

Using the above described procedure we obtain two independent PDFs, one for the upper arm, $p_u(\theta_u)$, and one for the lower arm, $p_l(\theta_l)$. Examples of these are illustrated in the figures 5. Different number of samplings might have been used to estimate the two PDFs. The accumulated probability mass for each PDF is therefore normalised to 1. In terms of the SMC algorithm the two normalised PDFs are the weighting functions, used to estimate the observation PDF, see equation 3.

4. Prediction in the Sub-Space

Besides the observation PDF we also need to define how the prediction in the sub-space is carried out. The standard approach is to predict the state-space variables, in this case α and H_z. However, recall that α and H_z only span a local sub-space that is unique for a particular time instant and hence smooth trajectories over time are usually not present. This means that good motion models are virtually impossible to set up. Nevertheless, this could work. In this particular work, however, we do it differently.

Instead of predicting the sub-space parameters we predict the anatomic parameters $\overrightarrow{E}$ and $\overrightarrow{H}$, see figure 1. This has two benefits. Firstly, more smooth trajectories can be expected for these parameters and hence, better motion models can be defined. Secondly, we can apply the measurements of the hand to correct the predictions and hence, obtain particles closer to the true state of the system.

Note that even though we have two different representations $(\overrightarrow{E}, \overrightarrow{H})$ and (α, H_z) our state-space is still only two dimensional, spanned by α and H_z. The former representation is taken into account, for each of the N particles representing the posterior PDF, by storing the corresponding $(\overrightarrow{E}, \overrightarrow{H})$ set for each (α, H_z) set. The mapping between the two representations can be found in [1]. In the following subsection we will show how the anatomic parameters are corrected based on the image measurements.

4.1. Correction the Predictions

First we will show how the prediction of the hand, $\overrightarrow{H}$, is corrected and hereafter we will show how the predicted position of the elbow, $\overrightarrow{E}$, is corrected. The correction of the prediction of $\overrightarrow{H}$ is based on the notion of combining the predictions and the image measurements. In figure 3 the predictions are illustrated using subscript 'p' while the corrected predictions are illustrated using subscript 'c'.

Since we know the camera ray through the hand in the current image, l, we can correct the prediction by projecting

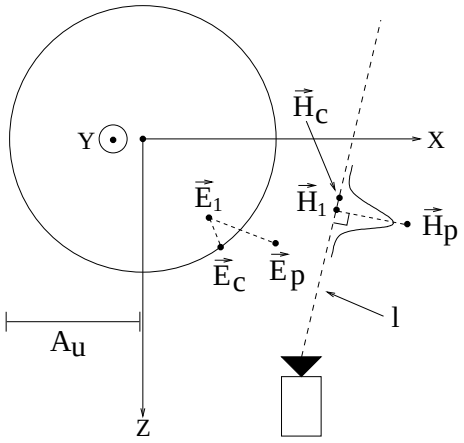

Figure 3: The shoulder coordinate system seen from above. The circle illustrates the sphere that defines the possible positions of the elbow. The large dashed line indicates a camera ray through the hand. See text for a definition of the parameters.

the predicted position of the hand, $\overrightarrow{H_p}$, to the line, l. The projected prediction is denoted $\overrightarrow{H_1}$ and calculated as $\overrightarrow{H_1} = \overrightarrow{P} + ((\overrightarrow{H_p} - \overrightarrow{P}) \cdot \overrightarrow{F})\overrightarrow{F}$ where $\overrightarrow{P}$ and $\overrightarrow{F}$ are the line parameters of l. $\overrightarrow{P}$ is the focal point and $\overrightarrow{F}$ is the vector spanned by the focal point and the position of the hand in the image, i.e., a direction unit vector.

The prediction in the SMC algorithm contains both a deterministic as well as a stochastic term. The deterministic term models the dynamics of the system while the stochastic term models the process noise in the system, i.e., the uncertainty of the deterministic model. Usually both predictions are applied at the same time, but we do not apply the stochastic prediction until we have corrected the deterministic prediction. This allows us to provide an estimate of the actual process noise as opposed to applying off-line training data. Concretely, we apply the stochastic prediction by randomly sampling from a Gaussian distribution located along the line, l, see figure 3. The mean of the Gaussian is defined by $\overrightarrow{H_1}$ and the standard deviation controlled by the error vector, hence the standard deviation $= k_1 \cdot ||\overrightarrow{H_p} - \overrightarrow{H_1}||$, where k_1 is a predefined constant. As the standard deviation and hence the process noise is controlled by the error vector we obtain a more accurate stochastic prediction.

After this operation we have the corrected prediction of the hand, $\overrightarrow{H_c}$. The difference between the predicted and corrected vector yields a measure of the prediction error, denoted $\overrightarrow{H_e}$ and calculated as $\overrightarrow{H_e} = \overrightarrow{H_c} - \overrightarrow{H_p}$.

The predicted position of the elbow can not directly be corrected by the image measurements. However, we know the elbow is likely to have a prediction error closely related to that of the hand as the hand and elbow are part of the same open-looped kinematic chain. We therefore calculate the corrected position, $\overrightarrow{E_c}$, by first adding the prediction error of the hand to the predicted value of the elbow, yielding $\overrightarrow{E_1} = \overrightarrow{E_p} + \overrightarrow{H_e}$, and then finding the point closest to $\overrightarrow{E_1}$ that results in a legal configuration of the arm. In mathematical terms $\overrightarrow{E_c} = \textit{arg}\min_{\overrightarrow{E}} \left|\left|\overrightarrow{E} - \overrightarrow{E_1}\right|\right|$ subjected to the constraints $\left|\left|\overrightarrow{E}\right|\right| = A_u$ and $\left|\left|\overrightarrow{EH_c}\right|\right| = A_l$. The solution to this problem can be found in [1]. As the error vector has already been subjected to diffusion we do not introduce yet another diffusion of the position of the elbow.

Evidently the corrected predictions will be more accurate as they are biased by the current measurements. Furthermore, the prior PDF, see equation 1, will be more accurate and, hence fewer particles are required to model the state of the system.

5. Estimating the MAP

Having dealt with the first issue raised in section 1.1, we now address the second issue, namely how to estimate the MAP.

Finding *the* correct state in a particular time instant is an important problem when an explicit representation of the tracked object is required. That is, when at some point all those fancy SMC tracking algorithms developed in different research-labs around the world are to be applied in real-life applications they have to be able to estimate the actual state at a particular time instant.

The most common method to estimating the current state is by finding the weighted average of all sampled particles, i.e., the first moment [8]. This is simple, but has the major drawback that it assumes the posterior PDF to be unimodal. In situations where this assumption do not hold, the estimated state might be located in a region where no samples are present at all or, even worse, in an impossible region of the solution space. In figure 4 an example of this is shown. The figure shows a (designed) 1D posterior PDF represented by weighted particles. The state indicated by '$\mathbf{x}$' is the first moment. Clearly this is an incorrect solution in this case. A slightly better solution in this line of thinking is to define the current state as the particle closest to the first moment.

Another common approach is to estimate the current state as the particle with the highest weight. As the weight of a particular state might be the sum of several particles this is a very sensitive approach. A better solution in this line of thinking is to estimate the current state as the state with the highest weight, indicated by 'Δ'. This is exactly the MAP, defined in section 1.

The MAP is a statistically sound concept that is well-suited in the context of Bayes' rule. However, in the context of estimating the current pose, i.e., the most likely configu-

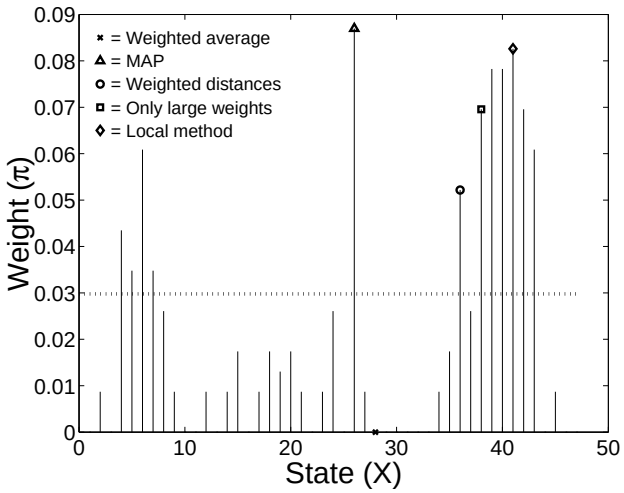

Figure 4: A posterior PDF represented by particles. The symbols refer to different estimates of the current state.

ration of an object, the MAP might not be the best choice. Three reasons exist.

I) The posteriori PDF is discrete. If the PDF was a continuous function the MAP would always be the state with the highest probability. However, for all practical matters the PDF is discrete and the MAP can therefore change due to different resolutions, i.e., the MAP is dependent on the resolution. Especially, in the case of multiple modes the MAP can change significantly.

II) The state with the highest probability might not always represent the "best" state. In figure 4 the MAP is represented as 'Δ'. Clearly a better estimate of the current state can be found if also the probability in the proximity is considered. In the figure a more dense probability mass is present to the right in the figure suggesting that a better estimate might be found here compared to the MAP.

III) The problem of an explicit representation of multiple-hypotheses. Clearly, SMC can represent multiple-hypotheses in an implicit manner, but how would we answer the following question: *"Which are the five most likely configurations of the objects in a particular instant of time?"* Applying the MAP we could find the five states with the highest probabilities. However, these are likely to originate from just one or two modes in the posterior PDF, i.e., one or two configurations, and by the question was meant, the five most likely *and* different configurations, i.e., different modes.

For the three above mentioned reasons we seek an alternative to the MAP that takes the probabilities in the proximity into account. An estimate of the current state which takes the probabilities (weights) of other states into account can be defined as the particle the state of which minimises the sum of the weighted distances between this state and all others states (particles).

$$\text{state} = \textit{arg}\min_{\overrightarrow{X}} \sum_{j=1}^{N} \pi_j \left|\left| \overrightarrow{X} - \overrightarrow{X_j} \right|\right| \tag{7}$$

This expression finds the optimal solution in terms of the weighted sum of absolute differences and is illustrated as state '**o**' in figure 4. In general, expression 7 gives a better estimate of the current state compared to the above mentioned methods. However, it has two major drawbacks; it is computationally demanding and tends to favour particles close to the first moment.

The high complexity is due to the high number (N^2) of weighted distances that needs to be calculated. To avoid a computational explosion only those particles having a large weight, suggesting that they are part of a large peak, are investigated. That is, particle a_i is investigated if $\pi_i > \frac{k_2}{N}$, where the constant k_2 directly determines the number of particles to be investigated, i.e., the computational complexity. In practise a new list containing all particles with a high weight is constructed and used instead of the original list containing all particles. In the example in figure 4 all particles above the dotted line are considered. The state found in this manner is illustrated in figure 4 as '$\Box$'. Besides lowering the complexity this method actually also "filters" the posterior PDF by removing all particles with a small weight and hereby improving the result, see figure 4.

Even though only particles having a large weight are considered, expression 7 still tends to favour particles close to the first moment. The reason being that expression 7 is a global method. To avoid this problem we instead suggest the local method in expression 8, which only considers the particles close to the particle being evaluated. Note that only those particles having a high weight are considered. Expression 8 defines the current state to be the state where the probability density in the proximity is highest. The state found using this expression is illustrated by '$\Diamond$' in figure 4. We denote this approach MOLAP, most likely a posteriori.

$$\text{MOLAP} = \textit{arg}\max_{\overrightarrow{X}} \sum_{V} \pi_j \tag{8}$$

where $V = \left\{ j \in [1, N] \;\middle|\; \text{abs}\big(X_k - X_{k,j}\big) < \Omega, \forall\, k \right\}$, and $k \in \left\{1, 2, ..., dim(\overrightarrow{X})\right\}$ is an index into the state vector, and Ω defines the proximity threshold.

In general expression 8 is an accurate estimation of the current state and since it considers the proximity it handles the first two problems mentioned above (I and II). The third problem (III) is handled by ignoring all particles contributing to the best MOLAP, when searching for the second best MOLAP, etc. Furthermore, the expression also ensures that the estimated state is always in an legal region of the state-space. Finally, the expression allows for a parametric rep-

resentation of the posterior PDF. We could represent each mode in the posterior PDF by a Gaussian with mean given by the MOLAP and covariance given by the sum in expression 8. The 'X' best MOLAPs (modes) could then be found and the posterior PDF represented by the sum of the Gaussians representing the best modes.

6. Results

In figure 5 the PDFs of the orientations of the upper- and lower arm are shown for the same image but for different number of samples in the DHT, i.e., different n. The choices of j and γ (see section 3.1 for definitions) directly control the computational complexity of the DHT. In this work we have set $\gamma = 0.05$ and $j = 100$, resulting in $n \in [500, 1500]$. Obviously these values need to be tuned to the particular application at hand.

The sub-figures in the second row show the estimated PDFs according to the stop criterion. Assuming the PDFs in the last row to be the "ground truth" a visual comparison reviles that the PDFs estimated by the stop criterion are not identical to the ground truth. Nevertheless, it is evidently that the primary tendencies are kept even though only a fraction of the samples have been applied, hence the dynamic stop criterion is valid.

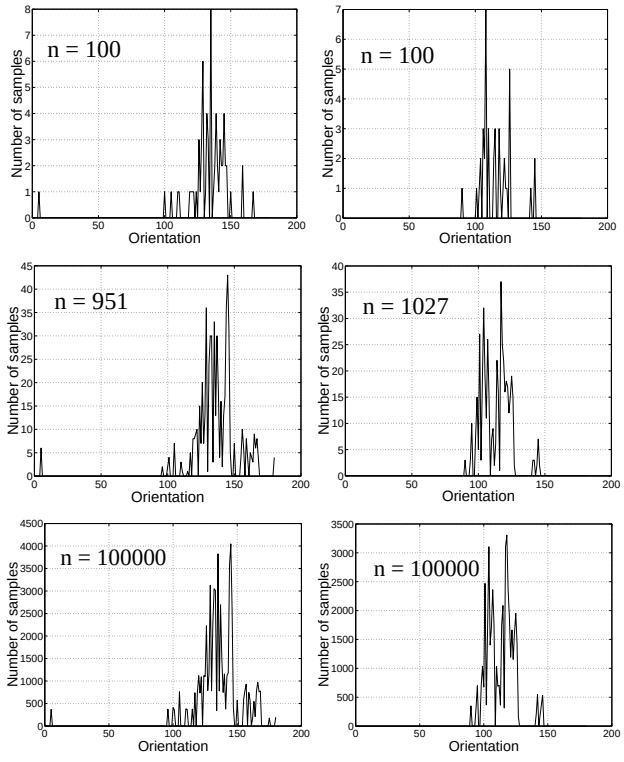

Figure 5: The estimated PDFs after n samples for a particular image. The left column is for the upper arm and the right column is for the lower arm.

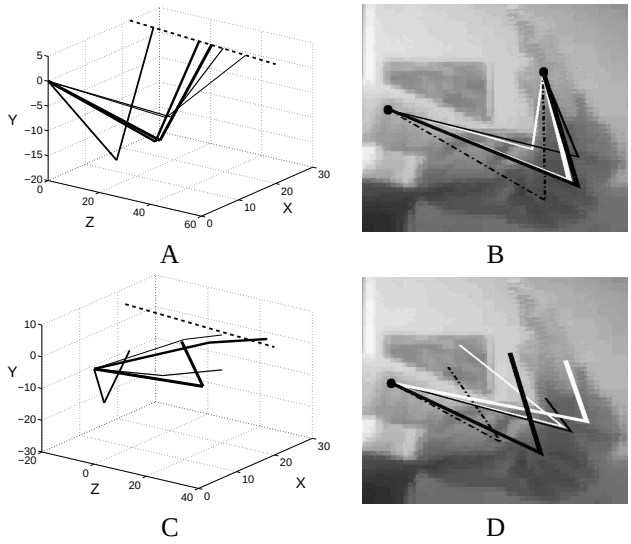

Figure 6: The five most likely configurations of the arm in the image in figure 2.A with sub-space SMC (A and B) and with standard SMC (C and D). For the 3D plots: the thicker the line the higher the likelihood. The dotted line illustrates the camera ray passing through the hand. For the 3D configurations projected into the image: the probability of the lines are in the following order (smallest probability first): thin white, thin black, dash-dotted, thick white, thick black). Note that only the relevant part of image 2.A is shown.

In figure 6 the effect of sub-space tracking in the SMC-algorithm is illustrated by comparing it to a standard SMC-tracker. For both algorithms 100 particles are applied.

After tracking the arm for 100 frames the five best MOLAPs are illustrated in figure 6 with- and without sub-space tracking[3]. The five best MOLAPs are illustrated in both a 3D plot and projected into the image. The best MOLAP is found utilising equation 8. The second best is also found using equation 8 but without considering the states in the hypercube (a square in our case) of the best MOLAP, etc.

The figure shows that tracking in our sub-space gives an improvement compared to tracking in the standard state-space. The figure also shows that the MOLAP (for both state-spaces) finds five distinct configurations and not just slightly different variations of the same configurations, hence our tracker truly handles multiple hypotheses.

The best MOLAP might not always be the correct state. However, it is always located at a (often large) peak with high probabilities in the proximity. The standard estimate of the current state, on the other hand, tends to be located in the centre of the state-space and sometimes at a position where no peaks are present at all.

[3]In the case of standard SMC-tracker our state-space was spanned by the four Euler's angles and the tracker was manually initialised to the correct pose 100 frames earlier than the image shown in figure 2.

In images such as the one in figure 2.A the posterior PDF is in general ambiguous. In our sub-space a number of correct poses can be found by increasing α as the distance between the hand and camera increases. This tendency can be seen in figure 6 which also shows that standard SMC fails to capture this tendency.

7. Conclusion

In this paper we have suggested how to improve the performance of SMC tracking. Concretely, we have made two contributions. Firstly, we have showed how image measurements from the current frame can be applied to generate a sub-space and, i.e., reduce the dimensionality. We showed it for the pose parameters of the arm, but the concept is more general and applies in all applications where discrete image features can be estimated independent of each other. Besides the tests presented above the improvement achieved by our sub-space approach can also be understood intuitively. Just imagine the complex nature of the posterior PDF utilising four Euler's angles or the screw axis representation.

Secondly, we have suggested how to estimate the current state of the system based on a local method as opposed to the standard global method. The suggested method also provides a way of estimating the different modes in a posterior PDF.

A future extension of our work is to use the sub-space approach on articulated objects with more degrees of freedom, e.g., the entire human body. The image features applied to span the sub-space will then be in the form of the position of the two hands, the head, the feet, and possibly other extremities of the human body.

It should be noted that our concept of sub-space tracking is comparable to [7][9][11]. They reduce the dimensionality of the tracking problem by first estimating the base of their articulated object and then estimating the remaining body parts, i.e., a partition of the state-space. Our approach differs as we estimate the end-effector of the articulated object as opposed to the base. This allows us to define a lower dimensional state-space as opposed to a partitioned state-space. Our work also differs as we use the end-effector to correct the prediction as opposed to [7][9][11] who use the base as a reference in the state-space when estimating the remaining state-space parameters.

Another way of describing the difference is to say that we are trying to combine the usually distinct low-level and high-level tracking approaches, where [7][9][11] apply pure high-level approaches.

References

[1] Technical report by the authors.

[2] C. Bregler and J. Malik. Tracking People with Twists and Exponential Maps. In *International Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, 1998.

[3] T.J. Cham and J.M. Rehg. A Multiple Hypothesis Approach to Figure Tracking. In *Conference on Computer Vision and Pattern Recognition*, Fort Collins, Colorado, USA, 1999.

[4] Y. Chen, Y. Rui, and T. Huang. Mode-based Multi-Hypothesis Head Tracking Using Parametric Contours. In *International Conference on Automatic Face and Gesture Recognition*, Washington D.C., USA, 2002.

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

[6] A. Doucet, N. Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[7] D.M. Gavrila and L.S. Davis. 3-D Model-Based Tracking of Humans in Action: A Multi-View Approach. In *Conference on Computer Vision and Pattern Recognition*, San Francisco, USA, 1996.

[8] M. Isard and A. Blake. CONDENSATION - conditional density propagation for visual tracking. *International Journal on Computer Vision*, pages 5–28, 1998.

[9] J. MacCormick and M. Isard. Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking. In *European Conference on Computer Vision*, Dublin, Ireland, 2000.

[10] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman. Human Body Model Acquisition and Motion Capture Using Voxel Data. In F.J. Perales and E.R. Hancock, editors, *AMDO 2002*, LNCS 2492. Springer-Verlag, 2002.

[11] J. Mitchelson and A. Hilton. From Visual Tracking to Animation using Hierarchical Sampling. In *Conference on Model-based Imaging, Rendering, image Analysis and Graphical special Effects*, Rocquencourt, France, 2003.

[12] T.B. Moeslund and E. Granum. A Survey of Computer Vision-Based Human Motion Capture. *Computer Vision and Image Understanding*, 81(3), 2001.

[13] S.M. Ross. *Introduction to Probability and Statistics for Engineers and Scientists*. Wiley Series in Probability and Mathematical Statistics, 1987.

[14] H. Sidenbladh, M.J. Black, and L. Sigal. Implicit Probabilistic Models of Human Motion for Synthesis and Tracking. In *European Conference on Computer Vision*, Copenhagen, Denmark, 2002.

[15] V.M. Zatsiorsky. *Kinematics of Human Motion*. Champaign, IL: Human Kinetics, 1998.

[16] Z. Zeng and S. Ma. Head Tracking by Active Particle Filtering. In *International Conference on Automatic Face and Gesture Recognition*, Washington D.C., USA, 2002.

[17] S. Zhou, V. Krueger, and R. Chellappa. Face Recognition from Video: A CONDENSATION Approach. In *International Conference on Automatic Face and Gesture Recognition*, Washington D.C., USA, 2002.