Simultaneous Pose and Posture Estimation with a Two-stage Particle Filter for Visuo-inertial Fusion

Nima Mehdi, Vincent Thomas, Serena Ivaldi, Francis Colas

Simultaneous Pose and Posture Estimation with a Two-stage Particle Filter for Visuo-inertial Fusion

Nima Mehdi[1], Vincent Thomas[1], Serena Ivaldi[1], Francis Colas[1]

***Abstract*— We address the problem of human pose and posture estimation without any high precision marker-based motion capture systems, by merging inertial data from wearable sensors and a single RGB camera. Our proposition is based on a biomechanical model of the human body and two coupled filters: the first filter takes advantage of the accurate posture observations provided by wearable sensors and a factorization of joints to estimate the human posture with a reduced number of particles while the second filter uses RGB camera observations to estimate the drift of the wearable sensor so as to estimate the global state (pose and posture). In order to combine those filters, the estimated human posture distribution of the first filter is used as a proposal distribution for the second fusion filter so as to focus on particles with an already high-likelihood posture and to improve the efficiency of the pose estimation. Results showed this approach can perform online estimation of the human posture and the human pose (through the drift of the wearable sensor) and performed better than techniques relying only on inertial sensor or on direct pose estimation.**

I. INTRODUCTION

Human posture and pose estimation is a particular focus in the computer vision and robotics communities. Much progress has been made in tracking the human posture from visual scenes [1]. However, the majority of these methods do not seek to deliver a kinematics-consistent estimation of the human posture where the plausible biomechanical properties of the human body are considered from the body segments to the joint angles. This is particularly critical for collaborative robotics where precise estimation and of the human's body posture is necessary for the robot to plan accurate assistance and estimate elements allowing it to access the human movement such as dynamics or ergonomics [2]. Wearable inertial measurement units (IMUs) have partially solved this problem as they enable a rather precise posture tracking even in industrial contexts [3], but they estimate poorly the 3D pose, i.e, the location of the human in the 3D space as they are subject to drift and often asymmetric and larges errors due to error in odometry caused for instance by irregular terrains, which limit their usability in case of mobile robots operating in close vicinity of humans especially when interacting actively with human agents. Motivated by these observations, we address here the case where the mobile robots need both postural and locational information about the human agent.

To do so, we propose to leverage both wearable sensors and cameras. Our main contribution in this article consists in the definition and evaluation of two coupled particle filters. A first factorized filter constructs a posterior distribution over the posture given inertial measurements, which is used as a proposal distribution in a second filter which merges inertial measurements and camera images to estimate the inertial drift as well as the posture. The contribution proposed in this paper relies on three key ideas: (1) estimating the drift of the inertial sensor (instead of the actual pose of the human) by using camera observations; (2) using the posture estimate based on the inertial observations as a proposal for the fusion estimation filter; (3), and, to reduce the amount of required particles in the posture estimation process, factorize the posture space by taking advantage of independencies between joints belonging to different kinematic chains. It allows estimating robustly the human posture in a biomechanical plausible state as well as locate the human agent in the 3D space in the context of human-robot interaction. In section II, we present related works on pose and posture estimation. Then, in section III, we formally describe the addressed problem, present an overview of our contribution and then detail the assumptions, the models and the equations used for each part of our contribution. The conducted experiments and the obtained results are described in section IV before a conclusion in section V.

II. RELATED WORKS

There are several directions that have been explored in the literature to estimate the 3D pose and posture of a human. They can be sorted based on the kind of sensor they use.

A. IMU-based

With their current size, inertial and magnetic sensors can be easily integrated into wearable sensors with minimal footprint at a relatively low cost. Both Yun *et al.*[4] and Roetenberg *et al.*[5] track the human body using a Kalman filter. A particle filter using directional distributions is proposed to track the gait of a human operator by To *et al.*[6].

As small as IMUs can be, they are still intrusive. Hence, solutions were proposed to learn to reconstruct the 3D human pose from a sparse IMU configuration. Marcard *et al.*[7] suggest a join optimization framework and make use of an anthropomorphic constraint for a realistic human model. Huang *et al.*[8] use a Recurrent Neural Network (RNN)-based method to learn human pose with IMU data. Inertial-

This work was partly supported by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 731540 (project AnDy), the French Research Agency (ANR) under Grant No. ANR-18-CE33-0001 (project Flying Coworker), the European FEDER in the context of the CPER Sciarat and the Creativ'Lab.

[1] Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France. firstname.lastname@inria.fr

based methods are especially adapted to retrieve a precise 3D posture as well as the body's dynamics, but they suffer from drift on the absolute pose since it is not observable.

B. Vision-based

Marker-less vision based systems have received major attention as cameras allows low setup and cheaper costs than other motion capture systems. Particle filters were proven to be an effective means for visual tracking and many works were aiming to adapt and perfect these methods to estimate the articulated human body from images: Deutscher *et al.*[9] use an annealed particle filter associated with a hierarchical search framework over images to estimate the human pose. Sedai *et al.*[10] suggest using a Gaussian-process-guided particle filter to estimate a 3D model from a video sequence.

More recently, with the popularity and efficiency of deep learning due to the increase of computational power and dataset sizes, many approaches aim to solve 3D pose estimation with artificial neural networks applied on images. Chen *et al.*[11] propose a Convolutional Neural Network to generate a 2D pose and then match the generated 2D pose to a 3D pose using a non-parametric nearest neighbor model. Pavllo *et al.*[12] introduce a temporal convolution model for neural networks to generate 3D poses from 2D poses. Moon *et al.*[13] estimate the 3D pose using a single RGB image by recovering the camera-centered coordinates using a dedicated neural network. All these methods can be highly effective but often need important amount of data and suffer from occlusion in clustered environments. Furthermore, their precision is often lower as it is easier for inertial units to follow dynamic movement.

C. Hybrid approaches

IMU-based techniques are subject to drift while vision-based ones can suffer from occlusions. Therefore, fusing image data with inertial data allows to obtain a more robust estimation of the human location and kinematics. Trumble *et al.*[14] separately estimate 3D poses from IMUs and images from multiple views cameras before merging them using several neural networks to obtain the estimated pose. Marcard *et al.*[15] assign 2D pose from images to 3D posture estimated from IMUs through optimization before feeding back camera poses to estimate the 3D posture and pose of tracked humans.

We draw inspiration from these approaches and propose to fuse inertial and camera measurements, but we do so using Bayesian filtering. It allows us to avoid the large amount of data needed in machine-learning-based methods and enables us to work with a biomechanical constrained model which can be personalized to the person tracked.

III. METHODS

A. Problem Statement

We want to estimate the full body pose (position, orientation, and posture) of a human based on inertial and camera measurements. Due to uncertainties, we formulate this problem as a Bayesian filtering problem. That is, we aim

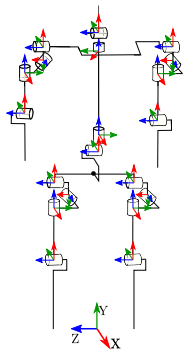

Fig. 1: Digital Human Model used to represent the posture.

to compute the probability distribution over the full pose $\boldsymbol{x}_k$ at time step k given the series of observations $\boldsymbol{z}_{1:k}$ from the beginning up to time step k:

$$p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k}). \tag{1}$$

The observations $\boldsymbol{z}_k$ are composed of the inertial motion capture measurements $\boldsymbol{z}_{x,k}$, which are the position and orientation of all body segments with respect to an arbitrary origin [5], and of the camera measurements $\boldsymbol{z}_{c,k}$, which, preprocessed by OpenPose [1], are the 2D coordinates on the image of the main joints.

The full body pose $\boldsymbol{x}_k$ is composed of the pelvis pose $\boldsymbol{x}_k^{\mathrm{P}}$, as a position and quaternion for the orientation, and of the posture θ_k. Our long-term aim is to use the posture for activity recognition [16] as such, we need the posture to be complete yet compact and expressed as joint angles. The inertial motion capture system we use can output joint angles but with 66 degrees-of-freedom and without biomechanical constraints. We chose to represent our posture using a Digital Human Model (DHM) composed of 13 segments linked by 20 revolute joints (see Figure 1). This choice is a trade-off between a high fidelity biomechanical model and the size of the state space. We refer to the review by Vianello *et al.*[17] on the importance of biomechanical models in context of Human-Robot collaboration. The mismatch between the observation DHM and our state DHM requires a form of retargeting, which will be taken care of by the estimation algorithm.

In our case, we assume that we know the pose of the camera. This can be obtained, for instance, through visual odometry or visual SLAM algorithms [18].

B. Filter Overview

As our problem is neither linear nor Gaussian, we use particle filtering (see tutorial from Arulampalam *et al.*[19]) for our inference of Equation 1. However, the large dimension of the state space (26) would require a huge number of particles.

This can be solved traditionally in two manners. The first one relies on leveraging the structure of the state space by ways of conditional or marginal independencies (as, for instance Djuric *et al.*[20]). However, all our observations

(inertial or visual) depend on the pelvis pose, which prevents defining a partition of the state space. Our first key idea is to preprocess the inertial observations to decouple the postural state between each body part (arms, legs, and trunk). Concretely, it amounts to computing the relative position $\boldsymbol{z}^r_{x,i,k}$ of each segment (e.g., the left hand) with respect to the root of the body part (e.g., the left shoulder). This is feasible with the inertial measurements since they contain the 3D position and orientation of each segment with respect to an arbitrary frame but this is not the case with the visual observation.

The second solution to limit the number of particles need is to optimize their location thanks to an adequate proposal distribution [21]. Our second key idea is to use the output of the individual segment filters as a proposal for a second filter, which uses the camera observations.

Therefore, we design our filter in two parts:

1) a first one, which we call *Multiple Posture Filter* (MPF) using only preprocessed inertial measurements which splits the posture estimation into five body parts (the left and right arms and legs and the trunk with the head),
2) and a second part, which we call *Fusion Filter* with the full state space but using a proposal distribution from the MPF.

This is further justified by the observation that the inertial measurements are quite accurate on the posture but suffer from drift whereas the camera observation is less precise on the posture but exempt from drift. The role of the fusion filter is therefore to compensate for the drift ξ_k of the reference frame of the inertial measurements as well as improve the posture estimate with the camera.

We actually propose two variants of the second filter: *Pose-FF* where the state is composed of the pelvis pose $\boldsymbol{x}^{\mathrm{P}}_k$ and the posture θ_k, and the *Drift-FF* where the state is composed of the drift ξ_k and the posture θ_k. In the latter filter, the pelvis pose can be recovered by composing the estimated drift with the inertial measurement of the pelvis $\boldsymbol{z}^{\mathrm{P}}_{x,k}$.

The overall structure of our filter can thus be summarized as in Figure 2. The remaining of this section details each of those parts.

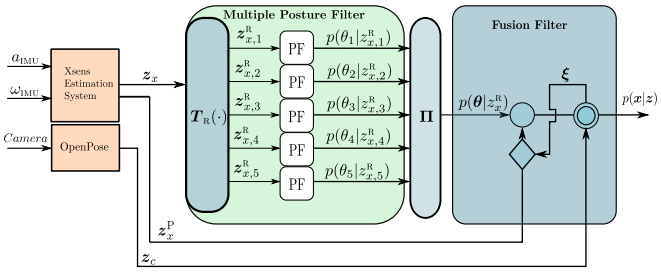

Fig. 2: Method Overview of the Drift-FF. T_{R} transforms inertial poses into relative position and Π computes the join distribution for the proposal density. a_{IMU} and ω_{IMU} are respectively the raw accelerations and angular rates from the IMU.

C. Multiple Posture Filter

As explained above, the first part of the filter estimates the posture by splitting it into five sub-filters, each for a body part: left and right arms and legs and the trunk including the head. Each of those sub-filters is a particle filter estimating the joint angles $\theta_{j,k}, j \in 1\,..\,5$ based on the inertial observations preprocessed into the relative position of each segment with respect to the root of the part.

The transition model $p(\theta_{j,k+1} \mid \theta_{j,k})$ is chosen to be a multivariate normal distribution centered on the previous joint angles. The next particle is then computed as follows:

$$\theta_{j,k} = \theta_{j,k-1} + \boldsymbol{w}_{j,k}, \quad \boldsymbol{w}_{j,k} \sim \mathcal{N}(\boldsymbol{0}, \Sigma_{\theta,j}) \quad \forall j \in 1\,..\,5.$$

We relate the state $\theta_{j,k}$ to the relative observation $\boldsymbol{z}^r_{x,j,k}$ through forward kinematics calculations. Therefore, the associated observation model is:

$$\tilde{z}_{x,j,k} = h_{fk,j}(\theta_{j,k}), \quad \forall j \in 1\,..\,5$$

$h_{fk,j}$ being the forward kinematics transform associated to the subtree j.

The associated likelihood for weight update $p(\boldsymbol{z}^r_{x,j,k} \mid \theta_{j,k})$ is defined as a multivariate normal distribution of mean $h_{fk,j}(\theta_{j,k})$ and covariance $\Sigma_{x,j,k}$ over $\boldsymbol{z}^r_{x,j,k}$. The weight update for the particle $\theta^i_{j,k}$ is defined as follows:

$$w^i_{k+1} \propto w^i_k \cdot p(\boldsymbol{z}^r_{x,j,k} | \theta^i_{j,k}) \quad \forall j \in 1\,..\,5.$$

This weight update is then followed by a resampling step allowing to minimize high variance in our particle population and propagate more often particle with more important weights [22].

Then, from our hypothesis of independence between each body part, the belief over the whole posture is simply:

$$p(\theta_k | \boldsymbol{z}^r_{x,1:k}) = \prod_j^5 p(\theta_{j,k} | \boldsymbol{z}^r_{x,j,1:k}) \tag{2}$$

D. Pose Fusion Filter

The Pose Fusion Filter (Pose-FF) is a particle filter estimating directly the posterior distribution on the pose of the pelvis $\boldsymbol{x}^{\mathrm{P}}_k$ and the joint angles θ_k through a weighted set of N particles:

$$p(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k \mid \boldsymbol{z}_{1:k}) \approx \sum_i^N w^i_k \delta \left(\boldsymbol{x}^{\mathrm{P}}_k - \boldsymbol{x}^{\mathrm{P},i}_k, \theta_k - \theta^i_k \right), \tag{3}$$

where δ is the Dirac delta function.

Rather than using the transition model to sample new particles $\boldsymbol{x}^{\mathrm{P},i}_k, \theta^i_k$ as for the MPF, we rely on a proposal distribution $\pi(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k)$. The aim of a proposal distribution is to generate particles at the best locations to represent the posterior $p(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k \mid \boldsymbol{z}_{1:k})$. In particular, we can leverage our estimation from the MPF (from Equation 2) to have a good estimate on $p(\theta_k \mid \boldsymbol{z}_{1:k})$. For the pose, we use a multivariate normal transition function:

$$p(\boldsymbol{x}^{\mathrm{P}}_k \mid \boldsymbol{x}^{\mathrm{P}}_{k-1}) = \mathcal{N}\left(\boldsymbol{x}^{\mathrm{P}}_{k-1}, \Sigma_p\right).$$

We can thus build our proposal as:

$$\begin{aligned}
&\pi(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k) \\
&= p(\boldsymbol{x}^{\mathrm{P}}_k \mid \boldsymbol{z}_{1:k}) p(\theta_k \mid \boldsymbol{z}^{r}_{x,1:k+1}) \\
&= p(\theta_k \mid \boldsymbol{z}^{r}_{x,1:k+1}) \int_{\boldsymbol{x}^{\mathrm{P}}_{k-1}} p(\boldsymbol{x}^{\mathrm{P}}_k \mid \boldsymbol{x}^{\mathrm{P}}_{k-1}) p(\boldsymbol{x}^{\mathrm{P}}_{k-1} \mid \boldsymbol{z}_{1:k}) \\
&= p(\theta_k \mid \boldsymbol{z}^{r}_{x,1:k+1}) \sum_{i}^{N} p(\boldsymbol{x}^{\mathrm{P}}_k \mid \boldsymbol{x}^{\mathrm{P},i}_{k-1}) w^i_{k-1} \delta \left(\boldsymbol{x}^{\mathrm{P}}_{k-1} - \boldsymbol{x}^{\mathrm{P},i}_{k-1} \right)
\end{aligned} \tag{4}$$

where Equation 4 is obtained by substituting the previous posterior according to Equation 3 and marginalizing on the pose.

Sampling from this proposal amounts to:

- sampling from the output of the MPF according to Equation 2 for the joint angles,
- sampling from $p(\boldsymbol{x}^{\mathrm{P}}_k \mid \boldsymbol{x}^{\mathrm{P},i}_{k-1})$ for the pelvis pose.

Sampling the joint angles can be further simplified by reusing the very same particle locations as the individual filters in the MPF.

Now, the weights of the particles need to be computed according to this proposal. More precisely, we have:

$$w^i_k := \frac{p(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k \mid \boldsymbol{z}_{1:k+1})}{\pi(\boldsymbol{x}^{\mathrm{P}}_k, \theta_k)}. \tag{5}$$

If we substitute Equation 4 into Equation 5, we can simplify the expression of the weights into:

$$w^i_k \propto w^i_{k-1} p(\boldsymbol{z}_{c,k} \mid \boldsymbol{x}^{\mathrm{P},i}_k, \theta^i_k), \tag{6}$$

where $p(\boldsymbol{z}_{c,k} \mid \boldsymbol{x}^{\mathrm{P}}_k, \theta_k)$ is defined as a multivariate normal distribution centered on the 2D projection of the 3D positions of each joint as computed by the forward kinematics and with a covariance Σ_C. With these weights, it is then possible to compute an estimate as the mode or the average of the weighted samples and to sample new particles for the next time step.

E. Drift Fusion Filter

The pose fusion filter can estimate the pose but relies in particular on a uninformed Gaussian transition model for the pelvis. Without odometry, it needs to be wide enough to account for the possible motions of the human and the variance of the particles might be relatively high.

However, the inertial measurements do provide an estimate of the pelvis position up to its global drift. An alternative expression of fusion filter can thus be obtained by estimating the drift ξ_k instead of the pelvis pose (Drift-FF).

The expressions are the same as for the pose fusion filter, except that the covariance of the transition model Σ_d can be assumed to be significantly smaller than Σ_p. A second, minor, difference is that the observation function also now depends on the inertial measurement of the pelvis $\boldsymbol{z}^{\mathrm{P}}_{x,k}$.

Finally, the drift fusion filter is described in 1.

Algorithm 1 Drift Fusion Filter

Require: $\{\boldsymbol{x}^i_{k-1}, w^i_{k-1}\}_{i=1}^N$,$\boldsymbol{z}_{x,k}$,$\boldsymbol{z}_{c,k}$
with $\boldsymbol{x}^i_{k-1} = [\theta^i_{k-1}, \xi^i_{k-1}]$
$p(\theta_k | \boldsymbol{z}_{x,k}) = \mathrm{MPF}\left(\{\theta_{j,k}\}_{j\in 1..5} \mid \boldsymbol{z}_{x,k}\right)$
for $i \in 1..N$ **do**
$\theta^i_k \sim p(\theta_k \mid \boldsymbol{z}_{x,k})$
$\xi^i_k \sim p(\xi_k \mid \xi^i_{k-1})$
$w^i_k \leftarrow w^i_k p(\boldsymbol{z}_{c,k} \mid \xi^i_k, \theta^i_k, \boldsymbol{z}^{\mathrm{P}}_{x,k})$
Resample with replacement $\{\boldsymbol{x}^i_k\}$ using $\{w^i_k\}$
end for
return $\{\boldsymbol{x}^i_k, w^i_k\}_{i=1}^N$

IV. EXPERIMENTS AND RESULTS

A. Implementation and Experimental Setup

Setup: Experiments were conducted with 3 healthy adults (3 males, 0 female, aged 24-26), recruited by word of mouth in the University. Each participant had its height, sole length as well as different anthropometric measurements taken in order to parametrize the DHM corresponding to their physiology.

The experiments were carried out inside an arena of approximately 4×4 meters, whose area is completely tracked by an OptiTrack Motion Capture system with 8 cameras. Each participant was equipped with the Xsens MVN suit with 17 embedded IMUs of the system. For ground truth, reflective markers from the OptiTrack Motion Capture system were placed on the motion tracking suit in the locations of pelvis, head, shoulder, elbows and wrists. A calibrated Intel RealSense Camera (RGB-D), marked as well with reflecting markers in order to retrieve its pose, was placed in the arena.

Tasks: The participants were instructed to perform a 5 to 10 minutes walk inside the arena, following rectangle shaped path. They were also instructed various hand motions both while standing still and walking.

Before each experimentation, the Xsens system was calibrated by walking forward and then back before staying put in a neutral pose, as instructed by the Xsens software. Each calibration was to be validated good enough to proceed by the software.

Implementation: The algorithm was implemented in Python 3.8 and run on a Ubuntu 20.04 with a 8-core Intel i7. The kinematic human model was encoded using the Unified Robot Description (URDF) which the angle joints defined in the files will constitute the posture part of the state space and the pose of the root frame will be used to estimate the drift. Forward kinematics calculations were performed using the Pinocchio library [23]. The 2D pose estimation by OpenPose was ran on camera data from the experimentation, on a Ubuntu 20.04 machine equipped with a NVIDIA 2070 RTX GPU. The captured data were downsampled at 5Hz and run offline. But the current implementation has the capability to run at 10Hz online for 1200 particles totals (200 for the fusion filter and 1000 in total for the MPF). One of the limiting elements is the 2D pose estimation, which can run at maximum around 10fps on our system.

B. Results

To confirm that our method can estimate the pose and posture while negating the drift, we carried different tracking scenarios. These scenarios compare 3 main methods: (1) our method, based on the sensor fusion and the proposal over the posture distribution, using the coupled Multiple Posture Filter (MPF) and estimating the drift, denoted *Drift-FF*, (2) a similar method with the same exact architecture, but without using the inertial odometry and tracking directly the pose instead of the drift, denoted *Pose-FF*, and, (3) the inertial based method corresponding to the raw Xsens observations and denoted *XS-IMU*. It must be noted that the Xsens system has already been designed to handle drift issues and to provide an accurate estimate of its user pose, therefore, the following experiments have the objective to assess if our system can improve the already efficient Xsens solution, which constitutes our baseline.

Drift Estimation and pose correction

As our system uses inertial based observations, we wanted to confront the method's resilience to drift. We ran both the Pose-FF and the Drift-FF on data gathered during an experiment where the human agent walked forward, circling in the dedicated space. We then computed the root squared error in position for all three previously enumerated methods. The results corresponding to approximately 30 minutes total of walking were compiled within Figure 3

Fig. 3: Comparison in Positioning Error between 3 systems: a hybrid system fusing information between camera and imu (Drift-FF), a system based on vision alone (Pose-FF) and a pure inertial method (XS-IMU)

From these results, Drift-FF provides an overall better estimate than Pose-FF and XS-IMU. On the right of Figure 3, we can notice that both fusion filters (Drift-FF and Pose-FF) provide better long-term results than XS-IMU with best results when the drift is estimated (Drift-FF). It can also be noticed that the error distribution of XS-IMU is very acceptable the 2 first cycles but this error and its interquartile range increase with time. The significant (apparently linearly) increase of the median value can be explained from the inertial drift. The larger interquartile range of the XS-IMU error distribution can be partially explained by the high sensitivity of the observed error to the Xsens calibration. Nevertheless, our method based on the drift estimate (Drift-FF) manages to generate better results with the exact same data, being more robust to the calibration step.The experiments, as illustrated by Figure 4, showed the drift is large and asymmetrical, expanding more significantly in a particular direction. Our method (Drift-FF) computes a good estimate of the pose through the drift, while not considering yet that assymetry. Further investigations will be considered in order to improve our drift estimate.

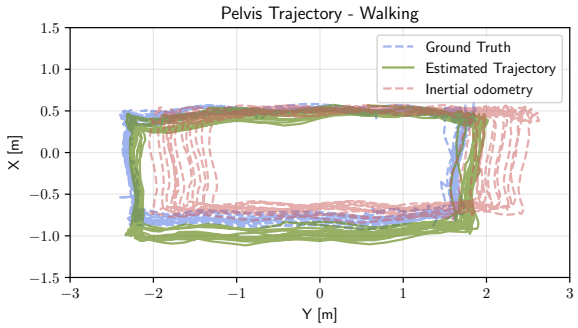

Fig. 4: The drift is well visible from the trajectory estimated from XS-IMU (inertial odometry)

Hand position and posture

One of the typical tasks in collaborative robotics is the hand-over of an object. It requires a rather accurate estimate of the position of the hand. The position of the head can also be of interest to avoid particularly dangerous collisions or even to estimate the human focus of attention. Both require accurate estimates of the global pose of the pelvis and of the position of the hand or head with respect to the pelvis. In absence of ground-truth on the joint angles, checking the accuracy of those positions is also an indirect means to evaluate the quality of the posture estimate.

Figure 5 compares the errors on the position of the hand for the three methods. On the left, we can see that the error is not significantly different for both our fusion filters, which is expected since they both use the same proposal from the MPF corrected with the same observation. We can also see that this error on the relative position is higher for our filters than for the raw Xsens observation. This is due to additional variance by the MPF in the retargeting process from the full Xsens model to our more biomechanical 20-dof DHM. The camera observation reduces a bit the variance but not enough to reach the same accuracy as XS-IMU.

On the right of Figure 5, are compared the errors on the absolute position of the hand. Those are the most meaningful errors and, as expected from the results on the drift, we can see that Drift-FF presents a significantly lower error than XS-IMU.

V. CONCLUSION

We present an approach to estimate the full pose of a human, including their pelvis position, pelvis orientation, and joint angle. This approach is based on two first ideas:

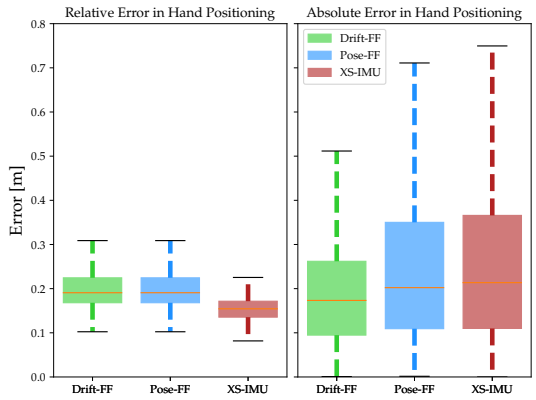

Fig. 5: Comparison of the errors on the position of the hand. Left: error on the relative position between the hand and the pelvis. Right: error on the absolute hand position.

decoupling inertial observation to split the filtering, and using the posture thus estimated as a proposal distribution for the fusion particle filter. In addition, a third key idea consists in estimating the drift instead of the pose to lower the variance induced by the motion model.

The resulting algorithm, Drift-FF, is shown to be effective to estimate a drift-free pose of the pelvis and of the hand. This was demonstrated on a collection of different trajectories taken with an Xsens inertial motion capture suit and a RGB camera. Due to its structure, it is also robust to temporary occlusions such as when the human goes out of the field of view of the camera.

We plan to use this system to estimate the activity of the human in a collaborative robotics scenario. A remaining challenge would be to reduce the number of inertial sensors required by our method, so as to lower costs and ease deployment.

ACKNOWLEDGMENT

The authors wish to thank Pauline Maurice for fruitful discussions on the digital human model.

REFERENCES

[1] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[2] W. Gomes, P. Maurice, E. Dalin, J.-B. Mouret, and S. Ivaldi, "Multi-Objective Trajectory Optimization to Improve Ergonomics in Human Motion," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 342–349, Jan. 2022.

[3] A. Malaisé, P. Maurice, F. Colas, F. Charpillet, and S. Ivaldi, "Activity Recognition With Multiple Wearable Sensors for Industrial Applications," *ACHI 2018 - Eleventh International Conference on Advances in Computer-Human Interactions*, p. 7, 2018.

[4] X. Yun and E. R. Bachmann, "Design, Implementation, and Experimental Results of a Quaternion-Based Kalman Filter for Human Body Motion Tracking," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1216–1227, Dec. 2006.

[5] D. Roetenberg, H. Luinge, and P. Slycke, "Xsens MVN: Full 6DOF Human Motion Tracking Using Miniature Inertial Sensors," *Xsens Motion Technologies BV, Tech. Rep*, p. 9, 2013.

[6] G. To and M. R. Mahfouz, "Quaternionic Attitude Estimation for Robotic and Human Motion Tracking Using Sequential Monte Carlo Methods With von Mises-Fisher and Nonuniform Densities Simulations," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 11, pp. 3046–3059, Nov. 2013.

[7] T. Von Marcard, B. Rosenhahn, M. J. Black, and G. Pons-Moll, "Sparse inertial poser: Automatic 3d human pose estimation from sparse imus," in *Computer Graphics Forum*, vol. 36, no. 2. Wiley Online Library, 2017, pp. 349–360.

[8] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll, "Deep inertial poser: learning to reconstruct human pose from sparse inertial measurements in real time," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–15, Jan. 2019.

[9] J. Deutscher, A. Davison, and I. Reid, "Automatic partitioning of high dimensional search spaces associated with articulated body motion capture," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, Dec. 2001, pp. II–II, iSSN: 1063-6919.

[10] S. Sedai, M. Bennamoun, and D. Q. Huynh, "A Gaussian Process Guided Particle Filter for Tracking 3D Human Pose in Video," *IEEE Transactions on Image Processing*, vol. 22, no. 11, pp. 4286–4300, Nov. 2013, conference Name: IEEE Transactions on Image Processing.

[11] C.-H. Chen and D. Ramanan, "3D Human Pose Estimation = 2D Pose Estimation + Matching," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, July 2017, pp. 5759–5767.

[12] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7753–7762.

[13] G. Moon, J. Y. Chang, and K. M. Lee, "Camera Distance-Aware Top-Down Approach for 3D Multi-Person Pose Estimation From a Single RGB Image," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South): IEEE, Oct. 2019, pp. 10 132–10 141.

[14] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse, "Total Capture: 3D Human Pose Estimation Fusing Video and Inertial Sensors," in *Procedings of the British Machine Vision Conference 2017*. London, UK: British Machine Vision Association, 2017, p. 14.

[15] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering Accurate 3D Human Pose in the Wild Using IMUs and a Moving Camera," in *Computer Vision – ECCV 2018*, Cham, 2018, pp. 614–631.

[16] A. Malaisé, P. Maurice, F. Colas, and S. Ivaldi, "Activity Recognition for Ergonomics Assessment of Industrial Tasks with Automatic Feature Selection," *IEEE Robotics and Automation Letters*, p. 9, 2019.

[17] L. Vianello, L. Penco, W. Gomes, Y. You, S. M. Anzalone, P. Maurice, V. Thomas, and S. Ivaldi, "Human-Humanoid Interaction and Cooperation: a Review," *Curr Robot Rep*, vol. 2, no. 4, pp. 441–454, Dec. 2021.

[18] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.

[19] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[20] P. M. Djuric, T. Lu, and M. F. Bugallo, "Multiple Particle Filtering," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, vol. 3, Apr. 2007, pp. III–1181–III–1184, iSSN: 2379-190X.

[21] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and computing*, p. 12, 2000.

[22] T. Li, M. Bolic, and P. M. Djuric, "Resampling Methods for Particle Filtering: Classification, implementation, and strategies," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 70–86, May 2015.

[23] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The Pinocchio C++ library – A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *International Symposium on System Integration (SII)*, 2019.