Predicting Human Motion using the Unscented Kalman Filter for Safe and Efficient Human-Robot Collaboration

Original
Predicting Human Motion using the Unscented Kalman Filter for Safe and Efficient Human-Robot Collaboration / Ferrari, Michele; Sandrini, Samuele; Tonola, Cesare; Villagrossi, Enrico; Beschi, Manuel. - (2024), pp. 1-8. (Intervento presentato al convegno IEEE International Conference on Emerging Technologies and Factory Automation tenutosi a Padova (ITA) nel 10-13 September 2024) [10.1109/ETFA61755.2024.10710736].

Availability:
This version is available at: 11583/2993059 since: 2024-10-17T16:36:56Z

Publisher:
IEEE

Published
DOI:10.1109/ETFA61755.2024.10710736

Predicting Human Motion using the Unscented Kalman Filter for Safe and Efficient Human-Robot Collaboration

Michele Ferrari*†, Samuele Sandrini†, Cesare Tonola*, Enrico Villagrossi† and Manuel Beschi*†

* Dipartimento di Ingegneria Meccanica e Industriale, University of Brescia, Italy,
Email: {michele.ferrari, c.tonola001, manuel.beschi}@unibs.it
†Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council of Italy,
Email: {samuele.sandrini, enrico.villagrossi}@stiima.cnr.it

***Abstract*—Predicting human motion is vital for enhancing safety and efficiency in human-robot collaboration. Researchers have dedicated significant efforts to developing accurate human models, often involving optimization and task-specific information. However, regardless of complexity, all models come with uncertainties that robots need to recognize to make informed decisions. This paper examines the performance of two simple models using the Unscented Kalman Filter (UKF) to filter and predict future human poses. Moreover, a combined version of the models is implemented using an Interacting Multiple Model (IMM) estimator. The objective is to evaluate the algorithms' prediction accuracy and uncertainty across various human-robot interaction scenarios under different operating conditions. This analysis identifies suitable settings where the simple model can be effective and highlights situations where a more complex system might be necessary.**

***Index Terms*—Human-motion prediction, Human-robot safe interaction, Unscented Kalman Filter**

I. Introduction

Modern robotic applications are centered on tasks that require seamless human-robot collaboration. In this scenario, recognizing and predicting human motion is fundamental to ensure safe and efficient interaction [1]. Predictive capabilities enable robots to design safe trajectories, adjust paths to prevent collisions [2], and make informed decisions based on anticipated human actions [3]. Safety is the primary aspect of human-robot collaborative applications, preserving operators from dangerous working conditions according to the current safety standards (*i.e.,* ISO 10218-1/2 [4], [5] and ISO/TS 15066:2016 [6]). Productivity is also fundamental, as is avoiding unnecessary stops and maintaining robot nominal working conditions as much as possible. In this context, accurate models of human behavior are essential for reliable trajectory predictions [7]. Various approaches have been employed for predicting future human postures, such as Recurrent Neural Networks (RNNs) [8], [9], Inverse Optimal Control [10], [11], and graphical models like Hidden Markov Models (HMMs) [12], [13]. Despite the advancements in modeling, all predictive models are subject to inherent uncertainties. Human behavior can be unpredictable, and planning based solely on expected movements can lead to unsafe trajectories. Consequently, providing confidence measures for these predictions is crucial, enabling robots to make informed decisions and adjust their strategies accordingly. Most state-of-the-art approaches focus on developing accurate models that capture human motion. At the same time, a limited number of studies investigated methods that can provide probability distributions over human position prediction. They mostly used HMMs, Gaussian mixture regression, and learning-based techniques. These approaches aim to improve model predictions using the temporal behavior of the worker, quantify uncertainties in predictions for safe trajectory planning, or update Bayesian beliefs on model confidence [14]–[17].

Generally speaking, accurate human models require time and financial investments for fine-tuning. For this reason, this study addresses the following questions: *Is it always necessary to use a complex human model? What is the performance of a simple model in different human-robot interaction scenarios?* Therefore, we evaluated the performance of a straightforward human pose filtering and prediction method based on the Unscented Kalman filter (UKF) in human-robot interaction contexts with varying operating conditions [18]. Specifically, we employ a simplified human motion model and a bank of UKF filters implementing an Interacting Multiple Model estimator (IMM) [19]. We perform *multi-step-ahead* prediction and analyze its performance in terms of mean value and covariance. The results provide valuable insights into the capability of this method, identifying the conditions under which it performs well, the achievable prediction horizon, and when, instead, more complex models are necessary.

The paper is organized as follows. Section II defines the human pose prediction problem; Section III outlines the experimental setup, including the testing protocol, filter design, and parameter tuning; Section IV presents and discusses the algorithm's performance results. Finally, Section V draws conclusions and describes potential future developments.

II. Method

The proposed method uses the framework described in Section II-A, while the predictor design is in Section II-B.

A. Problem definition

The human position in the Cartesian space is defined as the set of n body key points $p_i \in \mathbb{R}^3$ in the Cartesian space, typically the joint centers, defined as

$$y \in \mathbb{R}^{3n} = \left[p_1^T, p_2^T, \dots, p_n^T\right]^T . \tag{1}$$

The human model is a highly intricate and multifaceted subject. It encompasses various aspects of human motion and control mechanisms. This model can be effectively divided into two primary components: the human dynamic and control models.

The *human dynamic model* describes the human body's motion for each time instant t without considering how the forces and torques are computed

$$\begin{cases} y(t) = h_h(x_h(t)) \\ \dot{x}_h(t) = f_h(x_h(t), u(t)) \end{cases} \tag{2}$$

where x_h is the dynamic state, u is the input signal, $f_h(\cdot)$ is the dynamic function, and $h_h(\cdot)$ is the output function. The typical choices for the state x_h are the position and the velocity in the Cartesian space or the Configuration space. The acceleration can be conveniently used as input u instead of the joint effort using the feedback linearization [20].

The *control model* focuses on how the body regulates its movements to achieve desired actions

$$\begin{cases} u(t) = h_c(x_h(t), x_c(t), r(t)) \\ \dot{x}_c(t) = f_c(x_h(t), x_c(t), r(t)) \end{cases} \tag{3}$$

where x_c is the controller internal state, r is an exogenous signal (*i.e.,* the target object position during a grasping), $f_c(\cdot)$ is the controller dynamic function, and $h_c(\cdot)$ is the controller output function.

The complete dynamic model is the combination of (2) and (3), that can be discretized as

$$\begin{cases} y = h(x(k)) \\ x(k+1) = f(x(k), r(k)) + w_k \end{cases} \tag{4}$$

with $x = [x_h, x_c]$ the state of the complete model and k the current step index. Model uncertainties are considered by adding a zero-mean white noise $w_k \in \mathcal{N}(0, Q)$.

The vision system provides a measurement $z(k)$ of the human pose affected by the measurement noise v_k modeled as a zero-mean white noise $v_k \in \mathcal{N}(0, R)$ without loss of generality:

$$z(k) = y(k) + v_k . \tag{5}$$

The state $x(k)$ can be estimated by a state observer, which provides the state estimate $x(k|k)$ and the state covariance $P(k|k)$. State observers involve two algorithms:

- the *predict* step, or *a-priori* estimation, estimates the values of $x(k+1|k)$ and $P(k+1|k)$ at the time instant $k+1$ using information at the time instant k using (4). The *predict* step can be executed recursively to obtain the n-step prediction $x(k+n|k)$ with covariance $P(k+n|k)$.
- The *update* step, or *a-posteriori* estimation, corrects the estimate with the actual measurements computing the quantities $x(k|k)$ and $P(k|k)$ at the time instant k using information at the time instant k. This step is normally computed to minimize the trace of the $P(k|k)$ as proposed by Kalman [21].

In this work, the state observer is used to provide the best current estimation $y(k|k)$ of the human position and the n-step prediction of the future position $y(k+n|k)$.

B. Predictor implementation

The *human dynamic model*, as expressed in (2), can be obtained with different approaches; different complexity levels are present in literature.

The first approach acts directly into the Cartesian space of the human key points. Thus, the state vector is identified as $x_h(t) = [y(t)^T, v(t)^T]^T$ where $v(t) = \dot{y}(t) \in \mathbb{R}^{3n}$ is the human velocity. The dynamic system in (2) can be formalized as a double integrator model between acceleration u and position y. Therefore, it is possible to rewrite the model (2) in matrix form as the linear system

$$\begin{cases} y(t) = \mathcal{H}_h x_h(t) \\ \dot{x}_h(t) = \mathcal{F}_h x_h(t) + \mathcal{B} u(t) \end{cases} \tag{6}$$

where $\mathcal{F}_h \in \mathbb{R}^{6n \times 6n}$ is the state-transition model, $\mathcal{B}_h \in \mathbb{R}^{6n \times 3n}$ is the control-input matrix, $\mathcal{H}_h \in \mathbb{R}^{3n \times 6n}$ is the output matrix (observation model), defined as

$$\mathcal{F}_h = \begin{bmatrix} 0_{3n\times 3n} & \mathbb{I}_{3n\times 3n} \\ 0_{3n\times 3n} & 0_{3n\times 3n} \end{bmatrix} \; \mathcal{B}_h = \begin{bmatrix} 0_{3n\times 3n} \\ \mathbb{I}_{3n\times 3n} \end{bmatrix} \; \mathcal{H}_h = \begin{bmatrix} \mathbb{I}_{3n\times 3n} \\ 0_{3n\times 3n} \end{bmatrix}^T \tag{7}$$

where $\mathbb{I}_{3n\times 3n}$ and $0_{3n\times 3n}$ are, respectively, the identity and null matrix of $3n$-dimension.

In the second approach to human dynamics modeling, the state vector $x_h = [q_i]$ consists of the joint variables (q_i with $i \in [1, \dots, n_{DoF}]$) describing the kinematic model of the human. Thus, the state-transition model f_h can still be formalized as a double integrator between the joint acceleration and position. In contrast to the previous approach, however, the output function h_h involves the forward kinematics of the human model that relates the state variables (joint variables) to the measurements (Cartesian positions). Thus, the output function h_h can be generally expressed as

$$y(t) = f_{kin}(\mathcal{H}_h x_h(t)) \tag{8}$$

where the $f_{kin}(\cdot)$ is the nonlinear forward kinematics.

The *control model* as expressed in (3) can be given by the combination of many involved variables. Complex nonlinear models can be formulated when exogenous information is known, such as the task to be performed (*i.e.,* the target position or the location of obstacles) [8], [12], [22].

When the exogenous variables are unknown, it is possible to model the law of motion of the human body. For the sake of simplicity, it is possible to use a trapezoidal velocity profile where the acceleration u is a piecewise constant function. Thus, (3) can be defined as $u(t) = 0$ when the velocity is constant, or

$$\begin{cases} u(t) = x_c \\ \dot{x}_c = 0 \end{cases} \tag{9}$$

when the acceleration is constant[1]. Summarizing, if the constant-speed model is used, the state is $x = [y^T, v^T]^T$; if the constant-acceleration model is used, the state is $x = [y^T, v^T, x_c^T]$ where x_c is the acceleration value. The former model is suitable for slow movements, while the latter reacts aggressively to sudden changes [23]. To adapt to various operational conditions during human-robot collaboration, the Interacting Multiple Model (IMM) algorithm [19] can be used. This algorithm efficiently combines multiple models, managing the switch between them by updating the estimated model probabilities using a Markov chain.

The proposed approach uses the UKF due to the possible nonlinearities in (4), which requires as parameters the covariance matrices Q and R, and the initial value $P(0|0)$. The initial state estimate is set equal to the initial measurement assuming their derivatives to be null. The tuning of Q and R is crucial to obtain a reliable value of the prediction uncertainties. The covariance matrix Q can be simplified considering the double integrator as a Wiener Process where the source of uncertainty is the value of the acceleration, and these uncertainties are isotropic for all the components. The matrix Q is computed as in [24], using the acceleration uncertainty (variance) q_a as a tuning parameter.

The covariance R and $P(0|0)$ could be considered isotropic, therefore $R = r_y \mathbb{I}_{3n\times 3n}$ with r_y the measured position uncertainty (variance) and $P(0|0)$ a block diagonal matrix where p_y, p_v and p_a are the initial position, velocity, and acceleration uncertainties (variances). It is worth stressing that the role of $P(0|0)$ is meaningful only during the initial transient.

The quantities q_a and r_y need to be tuned by considering that the predicted uncertainty should match the uncertainty obtained from experimental results. The tuning is described in Section III, where experimental results are divided into identification and validation datasets.

III. Setup and Experiments

A. Setup

The experiments were conducted in an industrial collaborative robotic cell with a Universal Robot UR10e and a StereoLabs ZED RGB-D stereo camera. More details on the design choices and control architecture can be found in [25]. The camera uses proprietary skeletonization software [26] running at about 25 Hz on an NVIDIA GeForce GTX 1060 (6GB) GPU to identify key points and fit a kinematic model to the operator's body as depicted in Figure 1. The algorithm processes the camera data to filter the key point positions (*update* step, Section II), predicts the next state $x(k)$ (*predict* step, Section II), and extends this prediction over a specified horizon (*n-step predict*, Section II). All software was implemented using Python on the Robot Operating System (ROS - *noetic* version).

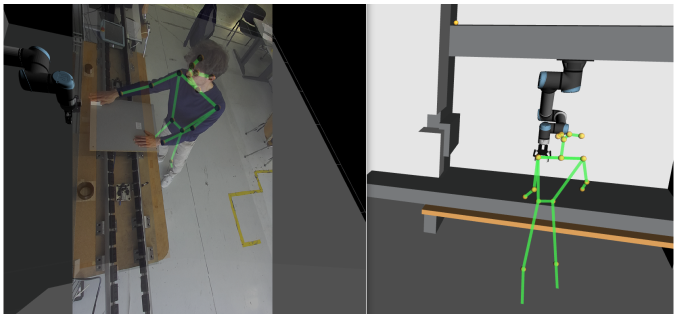


Fig. 1: Representation of the skeletonized operator while performing the experiments in the collaborative robotic cell.

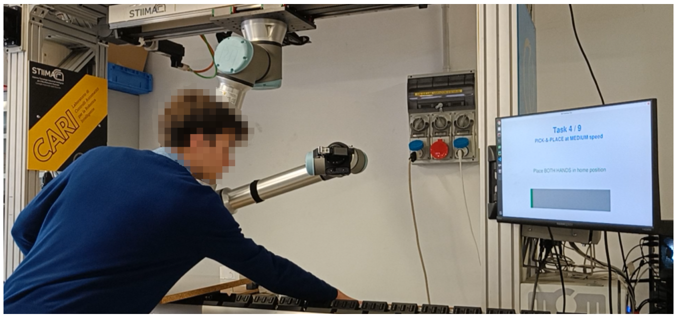

STIIMA
CARI
Task 4 / 9
PICK-&-PLACE at MEDIUM speed
Place BOTH HANDS in home position

Fig. 2: Operator performing the task according to the instructions of the GUI.

B. Experimental Protocol

The testing protocol involved the operator executing consecutively the following tasks at LOW, MEDIUM, and FAST speed:

- REACH-TO-GRASP movements to four different locations using one or both arms;
- WALKING and stopping within the cell;
- PASSING-BY, *i.e.,* walking parallel to the shared environment, entering and exiting the field of view of the camera.

The testing tasks were selected to reflect typical user movements in a robotic collaborative application. Ten subjects, chosen from the laboratory students with diverse experience in human-robot cooperation, performed the sequence following a brief training session. Camera data was continuously recorded throughout the test, with the initial and final timestamps for each motion at various speeds automatically logged by a graphical user interface (GUI), as depicted in Figure 2. The GUI also dictated the sequence of operations and their timing to ensure consistent test replication.

[1] x_c is unknown to the state-observer.

The experimental campaign aimed to evaluate the conditions under which the proposed model remains effective. This was achieved by assessing the algorithm's predictive accuracy, uncertainty bounds, and the influence of the prediction horizon on the results.

C. Filter Design

To mitigate measurement noise and integrate camera data with model predictions, three filters were implemented: a constant-velocity Unscented Kalman Filter (UKF), a constant-acceleration UKF, and an Interacting Multiple Model (IMM) filter, the latter comprising a bank of filters. The IMM approach was chosen based on the categorization of measurements into nearly-constant-speed movements, acceleration ramps, and intervals where key points were relatively stationary. Consequently, the constant-velocity filter excelled during slow or stationary phases, while the constant-acceleration filter was more responsive to abrupt position changes, albeit with significant overshooting towards the end of each motion. Additionally, larger diagonal elements of the model covariance matrix Q decreased the trust in the model in favor of the measurements, enhancing tracking accuracy but resulting in noisier filtering. This heightened reactivity is beneficial when the model's reliability is compromised due to sudden direction changes.

To achieve an optimal balance and design a filter with superior overall performance, the IMM incorporated the following filters: a constant-acceleration UKF (denoted as CA) with $q_a = 0.00225 \text{ m/s}^2$, another constant-acceleration UKF (CA_Q) with no model uncertainty ($q_a = 0 \text{ m/s}^2$), and a constant-velocity filter (CV) with velocity variance $q_v = q_a \cdot dt$, where $q_a = 0.00225 \text{ m/s}^2$ and $dt = 0.1$ s (the sampling time of the filtering loop). The q_a values were fine-tuned according to the procedure described in Section III-D. The value of dt was chosen to balance computational load and estimation reactivity. During this interval, updates to the UKF state based on new camera measurements are ignored. The performance of these three filters is illustrated in Figure 3, depicting the *REACH-TO-GRASP* task at *FAST* speed for a selected subject. Notably, the IMM filter output is smoother than that of the CA, which is particularly significant for the *n-step-ahead* prediction, resulting in reduced overshooting (see Figure 4a).

D. Parameter Tuning

Data from seven subjects were used for filter tuning, while three subjects were reserved for validation of the filter performance. Among the five uncertainty parameters (p_y, p_v, p_a, q_a, and r_y), only q_a underwent an iterative tuning process. Given that the relative weight of q_a and r_y determines the filter's confidence in the model versus the measurements, r_y was kept fixed whereas q_a was fine-tuned. Specifically, r_y was set to 0.0025 m/s^2, assuming a standard deviation (σ) of 0.05 m for the measured position of each key point.

To optimize computational efficiency and consider the worst-case scenario, the tuning routine for q_a focused solely on tasks performed at the *FAST* speed and predictions over the longest horizon, i.e., 0.5 s (five sampling periods). Starting from an initial value, q_a was iteratively reduced until approximately 68% of the filtered samples fell within the band $\pm\sigma$ around the *n-step* predicted values. This ensured that, under the assumption of white model noise, the predicted uncertainty aligned with that obtained from experimental results. As detailed in Section III-C, this procedure resulted in $q_a = 0.00225 \text{ m/s}$.

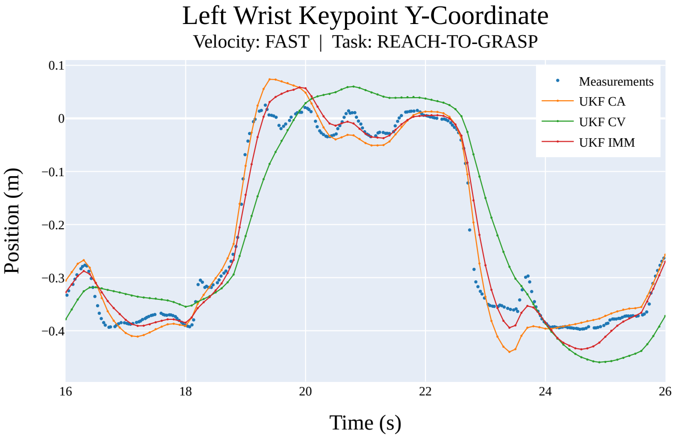


Fig. 3: Filter output for the constant-acceleration UKF (CA), the constant-velocity UKF (CV), and the IMM estimator.

The initial covariance parameters, p_y, p_v, and p_a, were manually set to reasonable values. Detailed tuning was deemed unnecessary since the initial covariance $P(0|0)$ primarily affects the filter's initial transient response. The parameter p_y was set equal to r_y because the filter state is initialized with the first measured position. The parameter p_v was set to 0.02844 m/s, assuming $3\sigma = 1.6 \text{ m/s}$, and p_a was set to 1.1111 m/s^2, assuming $3\sigma = 10 \text{ m/s}^2$. These values align with the experiments detailed in Section III-B, as no prescribed motion exceeded 1.15 m/s (Table I).

The parameters of the IMM estimator were manually fine-tuned to achieve the desired response. The transition matrix M of the Markov chain of the IMM was selected as

$$M = \begin{bmatrix} 0.55 & 0.15 & 0.30 \\ 0.15 & 0.75 & 0.10 \\ 0.60 & 0.30 & 0.10 \end{bmatrix}, \tag{10}$$

and the vector μ of the initial filter modes was set to $\mu = \begin{bmatrix} \mu_{CA}, & \mu_{CA_Q}, & \mu_{CV} \end{bmatrix} = [0.55, \ 0.40, \ 0.05]$. The elements $m_{i,j}$ of M represent the probability of transitioning to filter j from filter i. The chosen values prioritize the constant-acceleration filters, CA and CA_Q, over the constant-velocity UKF, allowing some flexibility for switching to a constant-velocity model when it better describes the motion. All filter parameters were fixed throughout the experiments, encompassing various tasks at different speeds and across different subjects. This approach aims to emulate a typical human-robot collaboration context, where one or more operators frequently switch between tasks.

TABLE I: Prescribed average velocity in $\mathrm{m/s}$ and selected key points for each task (COCO18 format [27]).

	SLOW	MEDIUM	FAST	KEY POINTS
REACH-TO-GRASP	0.15	0.25	0.70	2, 3, 4, 5, 6, 7*
WALKING	0.45	0.60	0.90	0, 1, 2, 5, 8, 11**
PASSING-BY	0.60	0.85	1.15	0, 1, 2, 5, 8, 11**

* Left-Right Arm. ** Nose, Neck, Shoulders, Hips.

IV. Results and discussion

The objective of the data analysis was to evaluate the performance of both the Constant Acceleration (CA) and Interacting Multiple Model (IMM) filters across various tasks (*REACH-TO-GRASP*, *WALKING*, *PASSING-BY*), velocities (*SLOW*, *MEDIUM*, *FAST*), and prediction horizons (0.1 s, 0.3 s, 0.5 s). The prescribed velocities and the specific key points analyzed for each task, drawn from the pool of 18 available, are detailed in Table I.

For the *REACH-TO-GRASP* task, emphasis was placed on key points within the upper kinematic chain (arms and shoulders), as their motion is paramount in such operations. Similarly, only key points associated with the trunk and head were considered during walking, as the swinging motion of the upper and lower limbs introduces noise to the tracking of the body's global position.

Denoted as $e(k) = y(k|k) - y(k|k-n)$ the (key-point-wise) error between the current filter estimate and the *n-step-ahead* prediction, these error metrics were evaluated for the selected key points in all operating conditions:

- the **mean value** $\mu_e = \mathbb{E}(e)$ of the key-point-wise error. This metric verifies the assumption of estimator *mean-unbiasedness*. A lower μ_e indicates a less biased *n-step-ahead* prediction.
- the **standard deviation** $\sigma_e = \sqrt{\mathbb{E}\left[\,(e-\mu_e)\,(e-\mu_e)^T\,\right]}$ of the key-point-wise error. This value represents the spread of the prediction error around its mean value and thus assesses the *accuracy* of the *n-step* estimate.

These metrics were chosen to evaluate the quality of the estimate instead of using the Root-Mean-Square Error (RMSE), as they separate the effects of bias and variance. For an unbiased estimator, the RMSE corresponds to the standard deviation.

Table II presents error metrics for each condition, aggregated by task-specific key points. Task velocity aggregation was also performed to derive a comprehensive global metric.

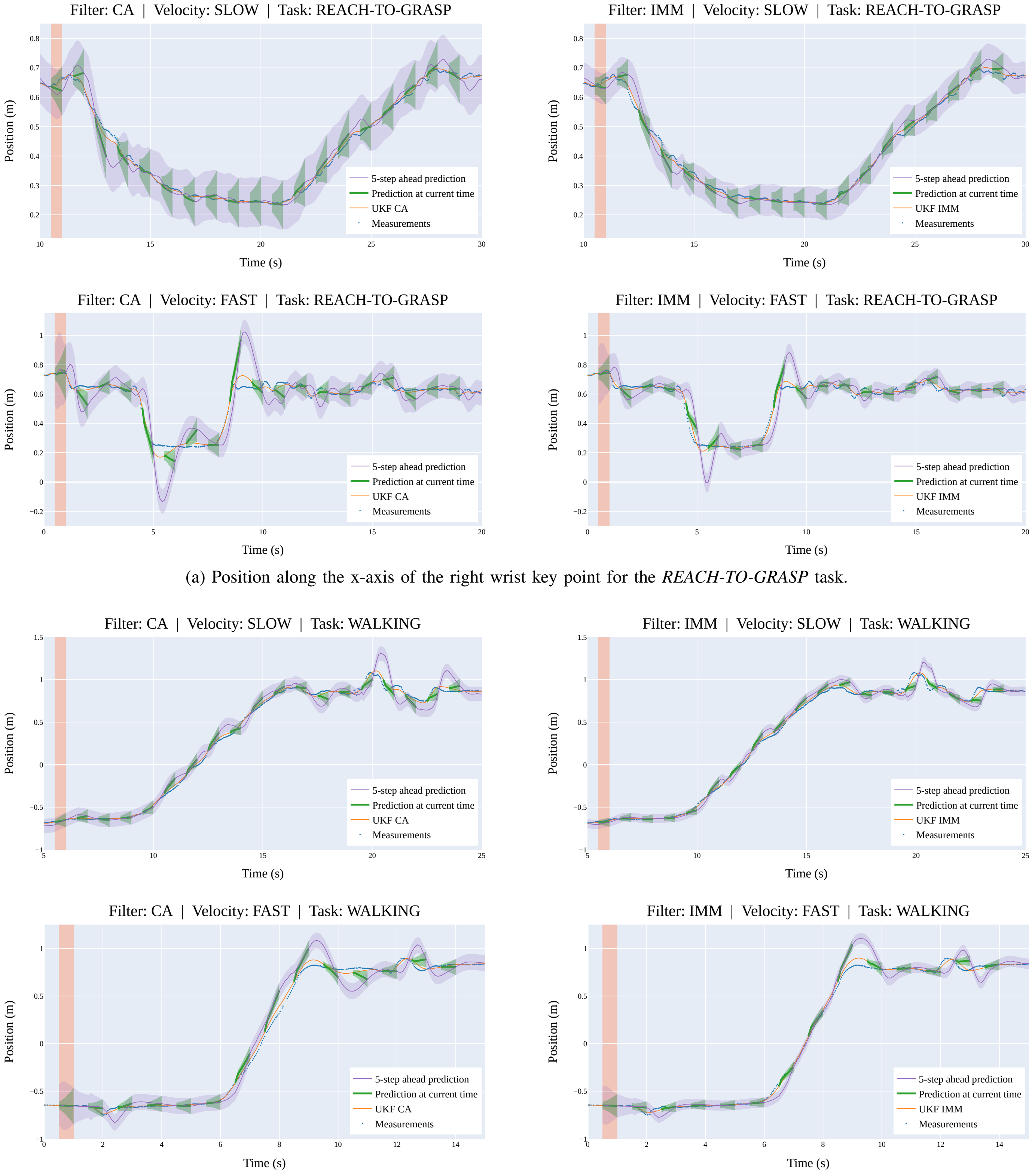
Overall, the mean error (μ_e) is notably small across tasks, indicating minimal bias in the predicted estimates. Specifically, the error magnitudes are in the order of 1×10^{-4} m for the *REACH-TO-GRASP* task, 1×10^{-3} m for *WALKING*, and 1×10^{-2} m for *PASSING-BY*. While both the Constant Acceleration (CA) and Interacting Multiple Model (IMM) filters exhibit comparable μ_e, the latter demonstrates a smaller standard deviation of the error (σ_e), resulting in a reduced Root-Mean-Squared-Error (RMSE) for all tasks and prediction horizons. Moreover, σ_e is generally lower for *REACH-TO-GRASP* compared to *WALKING* and *PASSING-BY*, likely due to the smaller movement amplitudes involved in the former task. Notably, as expected, σ_e increases with the prediction horizon, owing to the decreased accuracy of estimations over longer time intervals.

Qualitative assessments depicted in Figures 4a and 4b align with these quantitative findings. These figures illustrate relevant position changes for a specific subject in the validation set over time. For the *REACH-TO-GRASP* task, the motion of key point 4 (right wrist) along the x-coordinate is plotted. This coordinate, aligned with the world frame, parallels the operator's sagittal plane when facing the shared workbench, with movements predominantly occurring within this plane for such operations. Conversely, for the *WALKING* task, Figure 4b focuses on key point 0 (head center). This choice stems from its robust tracking by the camera, with the analysis centered on its motion along the y-axis. Notably, this axis aligns parallel to the collaborative workbench, affording ample movement within the camera's field of view (see Figure 1). Given space constraints, the *PASSING-BY* task was not included in this discussion due to its similarity to the *WALKING* task.

An orange vertical shaded area indicates the span of the receding horizon utilized for computing the *n-step-ahead* prediction. Specifically, only the *5-step-ahead* estimate is presented, representing the worst-case scenario. It is worth noting that more precise mean values and narrower error bands can be achieved by employing shorter prediction windows, as detailed in Table II. In each plot, both the filtered value and the *n-step-ahead* predicted value are depicted, accompanied by $\pm\sigma$ error bands. Additionally, with a sampling interval of 1 s, the subsequent five predicted states at the current timestep and their corresponding covariance cones are shaded in green. As anticipated, the fifth prediction corresponds to the violet curve, representing the *5-step-ahead* estimate computed five sampling intervals earlier. Notably, the fifth predicted upper and lower confidence limits (UCL / LCL) within each covariance cone seamlessly merge into the violet uncertainty band.

As expected, performance deteriorates with increasing key point velocity in both the *REACH-TO-GRASP* and *WALKING* tasks. Overall, the Interacting Multiple Model (IMM) filter provides more accurate and reliable *5-step-ahead* predictions, evidenced by smaller errors (e) and narrower uncertainty bands. This advantage is especially pronounced at *FAST* speeds, where the IMM filter significantly reduces overshoots compared to the Constant Acceleration (CA) UKF. However, overshoots remain substantial when predicting this far ahead using such a simple and general model. Achieving better predictions requires complicating both the *human dynamic* and *control model* with exogenous and task-specific information.

The code to execute the algorithm, conduct data analysis, and generate plots for all operating conditions can be accessed in the project's public repository [28].



(a) Position along the x-axis of the right wrist key point for the *REACH-TO-GRASP* task.

(b) Position along the y-axis of the head center key point for the *WALKING* task.

Fig. 4: Results for various tasks at *SLOW* and *FAST* speeds for the CA and IMM filters. Measured positions (blue dots), filtered values (orange lines), *5-step-ahead* predictions (purple lines with purple-shaded confidence interval), and following five predicted states at a given time instant (green lines with green-shaded expanding covariance cones). The vertical orange rectangle marks the sliding prediction horizon.

TABLE II: Performance Metrics computed on the Validation Set

Prediction Window	Model	REACH-TO-GRASP		WALKING		PASSING-BY	
		μ_e ($\times 10^{-4}$ m)	σ_e ($\times 10^{-2}$ m)	μ_e ($\times 10^{-4}$ m)	σ_e ($\times 10^{-2}$ m)	μ_e ($\times 10^{-4}$ m)	σ_e ($\times 10^{-2}$ m)
0.1 s **(one step)**	CA	-1.79	1.06	-32.4	1.36	-55.4	1.64
	IMM	-1.46	0.78	-31.5	1.07	-54.5	1.38
0.3 s **(three steps)**	CA	0.530	3.25	-35.4	4.03	-112	7.33
	IMM	-2.37	2.64	-39.5	3.93	101	6.78
0.5 s **(five steps)**	CA	1.81	6.42	-40.2	7.86	-171	11.1
	IMM	-2.23	4.93	-43.3	6.72	-148	10.0

V. Conclusions and future works

This study introduces a human motion prediction framework that demonstrates promising results in forecasting future human states within a 0.5 s time window. By utilizing a simple double-integrator model and constant-acceleration or constant-velocity motion laws, the framework forms a robust and general basis for predicting human movements. Notably, the uncertainty band provided by the *n-step-ahead* prediction algorithm could be used to enhance motion-replanning strategies by incorporating a stochastic estimate of future human motion.

Future research aims to refine the *human dynamic model* to improve the accuracy of kinematic descriptions. Leveraging the Unscented Kalman Filter (UKF) is expected to reduce computational overhead while accommodating nonlinearities and enriching the model's complexity. Additionally, developing a comprehensive *control model* is essential for understanding how the human body regulates its motion, considering factors like goal location, obstacle avoidance, and task-specific knowledge to enhance motion prediction accuracy.

Integration of this framework with motion replanning algorithms holds promise for generating human-aware trajectories in real-world scenarios. Furthermore, incorporating various sensors, including industrial safety radars, aims to capture diverse aspects of human movement and establish a dynamic safety strategy with discrete safety levels, ensuring effective human-robot collaboration across different environments.

Acknowledgment

The author(s) thank all the users who participated in the experiments. This study was partially carried out within the MICS (Made in Italy-Circular and Sustainable) Extended Partnership and received funding from Next-Generation EU (Italian PNRR-M4 C2, Invest 1.3-D.D. 1551.11-10-2022, PE00000004). CUP MICS D43C22003120001.

References

[1] H. Liu and L. Wang, "Human motion prediction for human-robot collaboration," *Journal of Manufacturing Systems*, vol. 44, pp. 287–294, 2017, special Issue on Latest advancements in manufacturing systems at NAMRC 45. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612517300481

[2] C. Tonola, M. Faroni, N. Pedrocchi, and M. Beschi, "Anytime informed path re-planning and optimization for human-robot collaboration," in *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, 2021, pp. 997–1002.

[3] S. Sandrini, M. Faroni, and N. Pedrocchi, "Learning action duration and synergy in task planning for human-robot collaboration," in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2022, pp. 1–6.

[4] International Organization for Standardization, "ISO 10218-1:2011 Robots and robotic devices — Safety requirements for industrial robots — Part 1: Robots." https://www.iso.org/standard/51330.html, 2011, [Accessed: 09 June 2024].

[5] ——, "ISO 10218-2:2011 Robots and robotic devices — Safety requirements for industrial robots — Part 2: Robot systems and integration." https://www.iso.org/standard/41571.html, 2011, [Accessed: 09 June 2024].

[6] ——, "ISO/TS 15066:2016 Robots and robotic devices — Collaborative robots." https://www.iso.org/standard/62996.html, 2016, [Accessed: 09 June 2024].

[7] C. Fang, L. Peternel, A. Seth, M. Sartori, K. Mombaur, and E. Yoshida, "Human Modeling in Physical Human-Robot Interaction: A Brief Survey," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 5799–5806, 2023.

[8] W. Liu, X. Liang, and M. Zheng, "Dynamic model informed human motion prediction based on unscented kalman filter," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 5287–5295, 2022.

[9] P. Kratzer, M. Toussaint, and J. Mainprice, "Prediction of human full-body movements with motion optimization and recurrent neural networks," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 1792–1798.

[10] T. B. Pulikottil, S. Pellegrinelli, and N. Pedrocchi, "A software tool for human-robot shared-workspace collaboration with task precedence constraints," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 102051, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584520302623

[11] J. Mainprice, R. Hayne, and D. Berenson, "Goal set inverse optimal control and iterative replanning for predicting human reaching motions in shared workspaces," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 897–908, 2016.

[12] J. Elfring, R. van de Molengraft, and M. Steinbuch, "Learning intentions for improved human motion prediction," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 591–602, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889014000062

[13] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[14] A. Kanazawa, J. Kinugawa, and K. Kosuge, "Adaptive motion planning for a collaborative robot based on prediction uncertainty to enhance human safety and work efficiency," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 817–832, 2019.

[15] W. Liu, S. Tian, B. Hu, X. Liang, and M. Zheng, "A recurrent neural network enhanced unscented kalman filter for human motion prediction," 2024.

[16] D. Fridovich-Keil, A. Bajcsy, J. F. Fisac, S. L. Herbert, S. Wang, A. D. Dragan, and C. J. Tomlin, "Confidence-aware motion prediction for real-time collision avoidance," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 250–265, 2020.

[17] L. Vianello, J.-B. Mouret, E. Dalin, A. Aubry, and S. Ivaldi, "Human posture prediction during physical human-robot interaction," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6046–6053, 2021.

[18] H. Afshari, S. Gadsden, and S. Habibi, "Gaussian filters for parameter and state estimation: A general review of theory and recent trends," *Signal Processing*, vol. 135, pp. 218–238, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165168417300014

[19] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[20] M. Seron, S. Graebe, and G. Goodwin, "All stabilizing controllers, feedback linearization and anti-windup: a unified review," in *Proceedings of 1994 American Control Conference - ACC '94*, vol. 2, 1994, pp. 1685–1689 vol.2.

[21] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 03 1960. [Online]. Available: https://doi.org/10.1115/1.3662552

[22] C. Schultz, S. Gaurav, M. Monfort, L. Zhang, and B. D. Ziebart, "Goal-predictive robotic teleoperation from noisy sensors," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2017, p. 5377 – 5383. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85027969665&doi=10.1109%2fICRA.2017.7989633&partnerID=40&md5=2a087dde6733f187b1ae54f033681741

[23] A. Aalerud and G. Hovland, "Dynamic augmented kalman filtering for human motion tracking under occlusion using multiple 3d sensors," in *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2020, pp. 533–540.

[24] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation for Kinematic Models*. Wiley, Ltd, 2002, ch. 6, pp. 267–299. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/0471221279.ch6

[25] A. Umbrico, A. Orlandini, A. Cesta, M. Faroni, M. Beschi, N. Pedrocchi, A. Scala, P. Tavormina, S. Koukas, A. Zalonis, N. Fourtakas, P. S. Kotsaris, D. Andronas, and S. Makris, "Design of advanced human–robot collaborative cells for personalized human–robot collaborations," *Applied Sciences (Switzerland)*, vol. 12, no. 14, 2022.

[26] Stereolabs, "Body tracking overview," 2024. [Online]. Available: https://www.stereolabs.com/docs/body-tracking

[27] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015.

[28] JRL-CARI-CNR-UNIBS, "hri_predict_ros," 2024. [Online]. Available: https://github.com/JRL-CARI-CNR-UNIBS/hri_predict_ros.git