2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

A particle filter technique for human pose estimation in case of occlusion exploiting holographic human model and virtualized environment

Costanza Messeri[1], Lorenzo Rebecchi[1], Andrea Maria Zanchettin[1] and Paolo Rocco[1]

Abstract— **In a collaborative scenario, robots working side by side with humans might rely on vision sensors to monitor the activity of the other agent. When occlusions of the human body occur, both the safety of the cooperation and the performance of the team can be penalized, since the robot could receive incorrect information about the ongoing cooperation. In this work, we propose a novel particle filter algorithm that, by merging the data acquired through a RGB-D camera and a MR headset, estimates online the human wrist position. This algorithm allows to significantly reduce the uncertainty of the human pose estimation, in case of both static and dynamic occlusions. To this purpose, the proposed particle filter is integrated with a detailed virtual model of the real workspace. Moreover, additional constraints describing the boundaries of the motion of the human upper body are included in a virtualized framework. The results showed that the proposed technique entails significant improvements, determining a relevant reduction of the estimation error and of the uncertainty of the estimate.**

I. Introduction

In the last few years, the field of collaborative robotics has gained an increasing interest, especially in industrial frameworks. To guarantee an effective cooperation and a safe coexistence, collaborative robots, also known as cobots, must be endowed with the capability of monitoring human motions, while being aware of the environment where they jointly cooperate, [1]. To this purpose, collaborative robots might be equipped with vision sensors, such as the popular RGB-D cameras, which can retrieve both colour and depth images encoding spatial information, [2]. Indeed, the data acquired through these sensors can be used to estimate the actual pose of the human operator within the working environment. This information can be communicated to the robot, so as to provide it with elements to understand the actual position of its partner within the shared workspace. There are however several related issues: bad lighting conditions, unexpected movements, partial occlusions of the human body, as well as the presence of complex geometries inside the workspace, often make the data retrieved by these sensors unreliable.

In this work, we propose a novel technique that allows to manage effectively the occurrence of occlusions. Indeed, this issue represents one of the main sources of uncertainty inside a collaborative scenario. The robot might in fact receive incorrect information about the other agent and react in an inappropriate way, thus reducing the performance of the team, or the safety of the cooperation. In these situations, the only available solution is to rely on human pose estimation techniques, namely a filtering algorithm, designed to estimate the human poses from the knowledge acquired in previous time instants, when the user was properly tracked. To reduce the uncertainty of the human pose estimate, we propose a novel technique that allows to estimate the occluded human pose by applying a constrained version of the well-know particle filtering algorithm.

One of the main contributions of this work is that the novel filtering technique provides an accurate estimate of the occluded human pose, by merging the data retrieved by means of an RGB-D camera with those of a wearable mixed-reality (MR) device. The intuition behind this approach is that the accuracy of human pose estimate can increase if both the constraints related to the kinematics of the human body, and the ones characterizing the working environment, are included in the estimation process. In this regard, the MR device perfectly suits this requirement, since, for instance, it can be endowed with the capability of scanning the real environment and generating a 3D virtual replica of it. By taking into account both kinematic constraints and the model of the environment, we are able to fully characterize the shape of the occlusion, thus reducing significantly the uncertainty related with the human pose. Moreover, by merging the data coming from a fixed (RGB-D) and a mobile (wearable) MR camera, a more complete model of the human and of his workspace can be created.

The rest of the paper is organized as follows. Section II provides an overview of the previous approaches used to estimate the unknown or uncertain human pose. Section III provides some background on a previous formulation of the constrained particle filter, which can be considered the benchmark against which we validate our algorithm. Section IV describes the details of the novel constrained particle filter proposed to enhance the estimate of the position of the human wrist in case of occlusion. Section V describes the experimental set-up used to validate our approach and discusses the achieved results. Finally, in Section VI some conclusions are offered.

II. Human pose estimation: background

Some of the traditional approaches applied to detect the human pose and the surrounding environment still rely on

[1] The authors are with Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Piazza Leonardo Da Vinci 32, 20133, Milano, Italy e-mail: ({costanza.messeri, andreamaria.zanchettin, paolo.rocco}@polimi.it, lorenzo.rebecchi@mail.polimi.it).

one fixed sensing device. In this regard, the methods based on Time of Flight (ToF) technology, such as the Microsoft Kinect, are definitely the mostly adopted. Many features related to the human silhouette can be easily retrieved by these sensors, such as the hands' pose or the reconstruction of the human face, [3]. In these situations, the complete pose of the human skeleton is obtained by comparing the data retrieved by the depth camera with a predefined human database, [4]. However, when the user is not completely visible to the sensing device, the pose of some human joints cannot be inferred correctly. To overcome the occlusion issues, several approaches can be applied. In [5] multiple depth cameras are used to compute in real-time the distance from dynamic objects. In [6] multiple Kinect cameras are simultaneously exploited to avoid the occurrence of occlusion, by merging the overall data acquired. However, this methodology leads to the accumulation of a huge amount of redundant data that are difficult to manage. Moreover, the method requires a sophisticated architecture to integrate the different devices. [7] exploited a Kalman Filter to track the human posture. However, despite being an efficient algorithm from a computational perspective and quite simple to implement, it was unsuitable for managing occlusions. An alternative technique commonly used to address the human pose estimation is the Particle Filter (PF), which works effectively in face of any kind of nonlinearities and non-Gaussian distributions. This algorithm recursively approximates, through a finite set of discrete values, the marginal distribution of the process state to be estimated, whenever a new measure is available, [8], [9]. As shown in [10], this technique is used in a variety of situations including the Simultaneous Localization and Mapping (SLAM) problem. In [11] the PF is used to improve the accuracy of the Kinect in case of occlusions by merging the colour information with the depth data. In [12] a method based on the combination of PF and KF is proposed to track linear and non-linear motion of a target object. In [13] a constrained version of the particle filter is presented. The proposed method is effective in limiting the uncertainty associated with the occluded human pose by taking into account anatomic distances of the human body and the information retrieved by the depth camera of the Kinect sensor. However, the method does not take into account the geometrical shape of the environment and the boundaries of human joints motion to limit the propagation of the uncertainty in the estimation process. Moreover, the robustness of the method fails when the user is occluded by a complex geometry, for instance, a concave object that cannot be fully characterized by a single sensing device.

Motivated by these considerations, we decided to adopt a constrained particle filter technique, and include novel constraints to enhance that version, by merging the data coming from the Kinect with an additional wearable MR device, Microsoft HoloLens.

III. Constrained Particle Filter: background

In this Section, we briefly summarize the formulation of the constrained particle filter, from now on referred to as 'CPF', presented in [13]. The expression 'Particle Filter' refers to a class of non-parametric algorithms used to infer the posterior distribution of the state of a dynamic process. Differently from the well-known KF, the PF does not need a model of the system or of the noise. In fact, the posterior distribution of the state is recursively approximated through a set of discrete values, also known as 'particles', which evolve independently from each other and are used to simulate all the possible evolutions of the unknown state. Each particle is associated with a specific weight. As the number of particles (N) tends to infinite, the result is exact. In [13], due to the computational costs of the algorithm, the estimation process was limited to the case of estimating the pose of a single human joint, the wrist. Moreover, a hierarchical model of the human silhouette was adopted. In this way, when, for instance, the wrist position was occluded, the estimation process was performed based on the knowledge of the previous joint position of the hierarchy that was correctly detected, i.e the elbow. Given the above, the state of the system can be represented by vector $s = (x, y, z, \dot{x}, \dot{y}, \dot{z})$, which are the position and the velocity, respectively, of the human wrist (expressed in Cartesian space), with respect to Kinect camera reference frame. Therefore, the following discrete-time state-space system (see [13]) was adopted:

$$\begin{cases} s_{k+1} = \boldsymbol{A} s_k + \nu_k \\ y_k = \boldsymbol{C} s_k + \psi_k \end{cases} \tag{1}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{I}_{3x3} & \boldsymbol{I}_{3x3}\Delta t \\ \boldsymbol{0}_{3x3} & \boldsymbol{I}_{3x3} \end{bmatrix} \qquad \boldsymbol{C} = \begin{bmatrix} \boldsymbol{I}_{3x3} & \boldsymbol{0}_{3x3} \end{bmatrix} \tag{2}$$

where $\nu_k \sim \mathcal{N}(0, \boldsymbol{Q})$ and $\psi_k \sim \mathcal{N}(0, \boldsymbol{R})$ represent the process noise and output noise, respectively.

The steps of the algorithm are reported in the following.

At time step k = 0:

- *Initialization phase*: select the desired probability distribution (see [13]), also called 'proposal distribution'. Draw N samples from that distribution and assign to each of them a weight equal to $1/N$. Acquire the anatomic distance between each pair of consecutive joints of the hierarchical human model described previously.

For future time instants k = $1, \ldots, K$:

- *State evolution*: the state of each particle is propagated a step further in time according to the distribution $p(s_k|s_{k-1})$;
- *Measurement update*: check whether a new measurement, y_k, of the target human joint position has been retrieved by the sensing device. If yes, update the weights of the samples according to the 'Closed loop' (CL) procedure, otherwise, according to the 'Open loop' (OL) one, as will be clarified later one.
- *Weight evolution*: Check if the distance between each particle (representing a candidate position of the wrist) and the position of the elbow lays inside a spherical crown, whose center is the position of the elbow and whose ray is equal to the distance between the elbow

and the wrist retrieved in the *Initialization phase*. This operation will be referred to as 'Skeletal distance' check. Then, if a particle passes the test, its weight is updated following the CL or OL procedure, based on the outcome of the *Measurement update* phase.

- CL: assign a weight to the $i\text{-}th$ particle, w_k^i, proportionally to the likelihood of the new measurement given the sample:

$$w_k^{(i)} = p(y_k|s_k^{(i)})w_{k-1}^{(i)} \tag{3}$$

$$p(y_k|s_k^{(i)}) = \frac{e^{-\frac{1}{2}(y_k - Cs_k^{(i)})\boldsymbol{R}^{-1}(y_k - Cs_k^{(i)})}}{\sqrt{2\pi\, det(\boldsymbol{R})}} \tag{4}$$

  where $p(y_k|s_k^i)$ is the likelihood of the new measurement, given the particle, and $\boldsymbol{R}$ is a diagonal matrix whose elements are the standard deviations of the noise acting on the measurements.
- OL: the particle is subject to the occlusion detection test (see [13]). If it survives, since a new measurement is not available, its weight is set equal to $1/N$.

In both cases the cumulative weight vector, $w_{cumul}^k(i)$, is computed according to (5).

$$w_{cumul}^k(i) = \sum_{j=0}^{i} w_k^j \tag{5}$$

- *State estimate*: compute the estimated state:

$$\hat{s}_k = \frac{1}{w_{cumul}^k(N)} \sum_{i=0}^{N} w_k^i s_k^i \tag{6}$$

- *Resampling*: a new set of particles is generated from the posterior belief, which is computed based on the values assumed by their weights, [13]. This step is crucial to avoid the so-called 'degeneration phenomenon'.

The constrained particle filter just presented allowed to estimate the human position in real-time. However, some weaknesses in the formulation can be observed:

- the particles are allowed to propagate through objects: this can be misleading and increase the uncertainty in the estimation process.
- the joint motion limits of the human body are not explicitly included in the PF constraints. This allows the samples to propagate also outside the regions which constitute the natural range of motion of the human articulations. Therefore, particles are still allowed to propagate to unrealistic positions.

IV. Constrained particle filter: new formulation

In the following, we propose the novel formulation of the CPF. In order to solve the first issue of the previous version, we include the model of the environment where the user works. In this way, the region where the particles are allowed to propagate is better shaped and further restricted, thus obtaining a reduction of uncertainty in the estimate. For what concerns the second issue, we propose to create a 3D virtual model of the human silhouette, according to the practice known as 'avateering', [14]. Indeed, in a MR application, the operator's avatar could be appropriately located inside the model of the environment generated previously. Hence, in this MR framework, we can avoid the propagation of the particles through the surfaces of the real environment mapped previously. Moreover, in this virtual framework, we can associate each joint of the human avatar with a virtual bounding volume that represents his joint limit. This operation is possible if the samples are interpreted as virtual objects that can collide with the surrounding surfaces present in the virtual environment, as will be clarified later on.

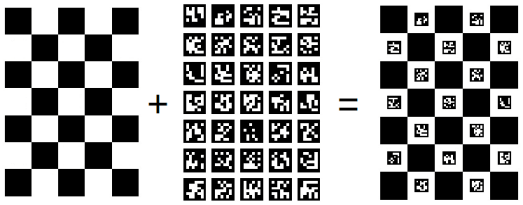

Fig. 1: ChArUco board is obtained by merging chessboard pattern and ArUco markers

The formulation of the constrained particle filter that we propose in this paper exploits as input data sources the information obtained by merging the data retrieved by HoloLens to those of the Kinect, as will be clarified in the following. Since our work is framed within a multi-sensor scenario, a method to merge the data coming from different sources, i.e Kinect and HoloLens is required. Section IV-A addresses this problem. Section IV-B describes in detail the formulation of the virtualization of the model of the human and the one of the environment to manage occlusions.

A. System calibration

In this subsection, we describe the method adopted to relate the data acquired through the Kinect sensor to the ones retrieved by the MR headset (HoloLens). Clearly, the purpose of this phase is to retrieve the homogeneous transformation matrix that allows to relate the data coming from the two sensors to one another. To do that, a quite popular technique is to use a fiducial marker, such as a QR code, that can be appropriately recognized by the sensing devices. In this work, we decided to exploit a particular fiducial marker, also known as ChArUco board, that consists in a chess of ArUco markers, see Fig. 1. The motivation underlying the choice of a ChArUco is that it composes the best features of ArUco markers, [15], with the classical chessboard pattern. In fact, the pose of a chessboard can be better estimated thanks to the greater number of points available, corresponding to the internal vertices between black and white squares. Hence, we developed a MR application that enables HoloLens to recognize the ChArUco marker.

Once the marker has been appropriately detected by each sensing device, the homogeneous transformation matrix expressing the pose of the marker with respect to the camera reference system is obtained. As illustrated in Fig. 2, these

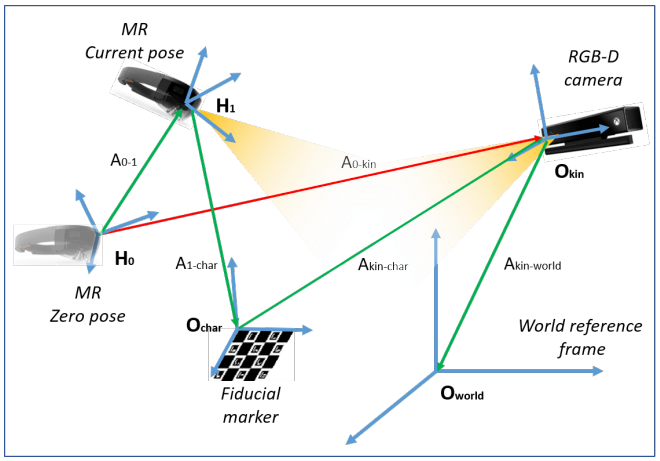

Fig. 2: Schematic representation of the calibration procedure

matrices are represented by $\boldsymbol{A_{1-char}}$ and $\boldsymbol{A_{kin-char}}$, for HoloLens and Kinect, respectively. The schematic representation of the complete set-up is reported in Fig. 2, where $\boldsymbol{H_0}$ is the initial holographic reference system which coincides with pose of the holographic headset, worn by the user, at the initial time instant, when the application is launched. This is assumed by HoloLens as the fixed reference frame with respect the successive motions of the user, which can be tracked and extracted using built-in methods. The actual pose of the device with respect to $\boldsymbol{H_0}$, is thus represented by $\boldsymbol{A_{0-1}}$. Finally, we can relate the data coming from HoloLens to those of the Kinect, by composing the homogeneous transformation matrices, as described by (7):

$$\boldsymbol{A_{0-kin}} = \boldsymbol{A_{0-1}}\boldsymbol{A_{1-char}}\boldsymbol{A_{kin-char}^{-1}} \tag{7}$$

B. Constrained Particle Filter: virtualized framework

In this subsection, we describe in detail the methodology used to virtualize the PF framework, which is a key point of our novel formulation.

1) Environment virtualization: The aim of this phase is to generate a set of data that provides a virtual volumetric description (replica) of the real environment where the user works. To do so, we created a MR app in Unity. This is a game-engine that allows the user to generate custom holograms, associate them with a specific behaviour, and simulate it before the deployment of the generated model of the environment on HoloLens. This MR app has three key features: the capability of tracking the user's head pose, scanning the surrounding environment (through the so-called 'Spatial Mapping' capability), and displaying as hologram the 3D model of the room previously generated. Thus, once the device is enabled, the operator wearing the MR headset can perform the scanning process by moving his/her head around him/her, so as to cover the desired surrounding environment, see Fig. 3. While the operator is scanning the environment, a 3D replica of the workspace is instantaneously produced and continuously updated. These volumetric data can be acquired by HoloLens thanks to the built-in Time of Flight (ToF) sensor. The operation just

Fig. 3: Picture of the real environment scanned by HoloLens

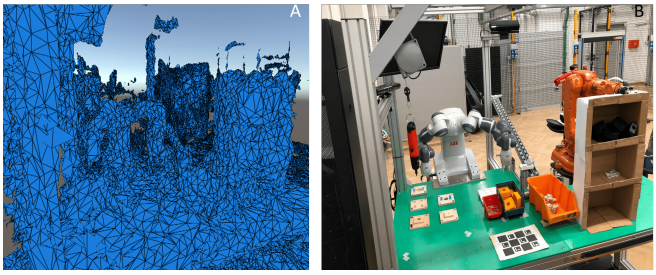

Fig. 4: Example of the obtained meshed workspace (A) and the corresponding real environment (B)

described produces a replica of the environment in the form of meshes which are displayed to the user in real-time in the form of holograms, see Fig. 4. Each mesh is basically a polygon whose coordinates are expressed with respect to the initial reference system of the HoloLens, $\boldsymbol{H_0}$.

2) Human model virtualization: In the CPF, the particles represent the candidate positions where the unknown human wrist could be located. To limit the volumes where the particles are allowed to propagate, we need to discard not only the samples that pass through an existing geometry of the real environment, but also the ones that violate the physical constraints of the human body. To do that, we create inside the Unity simulation environment an avatar (hologram) of the human, as illustrated in Fig. 5. The position of the joints and the length of each link are estimated based on the distance between two consecutive joint positions retrieved by the Kinect. Moreover, a hologram representing the joint boundaries, from now on referred to as 'joint bounding volume' (JBV), is created. Since we want to avoid the particles to model erroneous poses for the human body, we are interested in determining if they violate the physical range of motion of the joint they are associated with. To do that we exploited the concept of kinematic chain, that allows to proceed to the estimation of a specific joint, relying on the knowledge of the position of the previous ones. In this way, for instance, after computing the pose of the shoulder and of the elbow, we know that the wrist should lie on a specific plane, as illustrated in Fig. 6. Since, when the state of the particles is updated, the JBV is already in place, the particles will be restricted to propagate inside it. The shape of the JBV

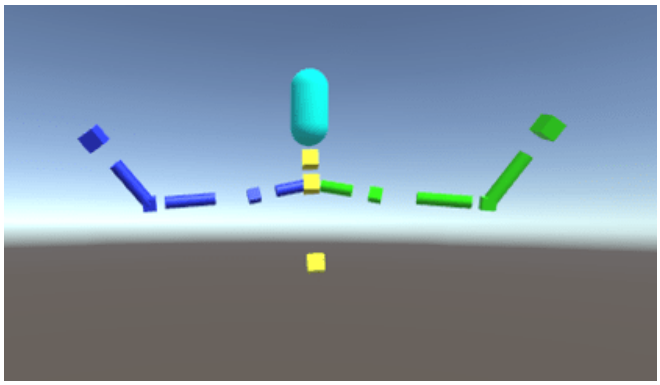

Fig. 5: Virtual model (avatar) of the human animated using the RGB-D camera

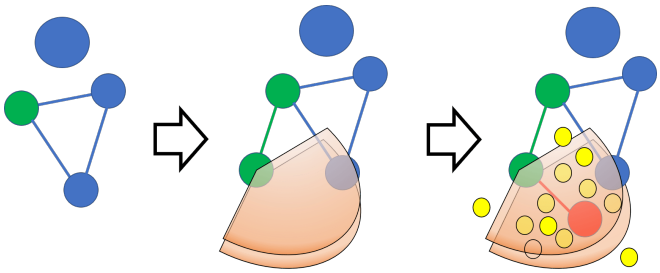

Fig. 6: Sequential procedure to position the wrist JBV

for each joint can be defined, based on the knowledge of the joints boundaries, [16]:

$$-9^\circ \leq \alpha_1 \leq 160^\circ \tag{8}$$

$$-43^\circ + \frac{\alpha_1}{3} \leq \alpha_2 \leq 153^\circ - \frac{\alpha_1}{6} \tag{9}$$

$$-90^\circ + \frac{7\alpha_1}{9} - \frac{\alpha_2}{9} + \frac{2\alpha_1\alpha_2}{810} \leq \alpha_3 \tag{10}$$

$$\alpha_3 \leq 160^\circ + \frac{4\alpha_1}{9} - \frac{5\alpha_2}{9} + \frac{5\alpha_1\alpha_2}{810} \tag{11}$$

$$20^\circ \leq \alpha_4 \leq 180^\circ \tag{12}$$

where α_1, α_2, α_3 and α_4 are the joint variables of the human upper-arm (see Fig. 7). Once α_4 is retrieved, since the pose of the wrist depends only on α_4, we could fix a volume according to (12). From a theoretical perspective, the JBV obtained for the wrist should be a plane. However, this hypothesis turned out to be too strict, since, when the position of the wirst is orthogonal to that plane, the number of particles that satisfy the constraint provided by the JBV would decrease dramatically. Therefore, this hypothesis was relaxed and the volume extended also to the third dimension. Hence, by merging the constraints illustrated in Section III with the new JBV, the remaining volume within which the samples are allowed to propagate corresponds to a 3D sector of an arc (see Fig. 8). To animate the virtual model of the human, a technique known as 'avateering' is applied. This allows to couple the virtual avatar of the human with the skeletal poses acquired by the Kinect sensor. Therefore, the joint positions acquired by the RGB-D camera are

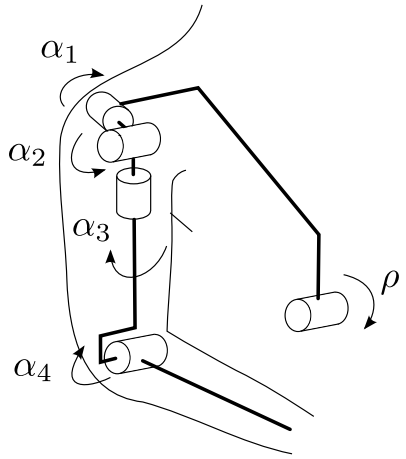

Fig. 7: Joint variables of the human arm

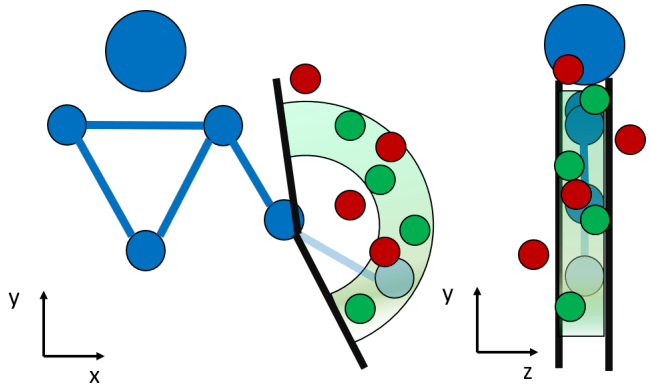

Fig. 8: Example of JBV: frontal view (left) and lateral view (right)

used within the PF framework, while the joint orientations retrieved are exploited to determine the orientation of the JBV.

3) Collision detection: In the following, we describe one of the main improvements with respect to the CPF illustrated in Section III. This is the collision detection capability, which is a method introduced in the virtual framework to determine whether the particles are interacting and, potentially, colliding with other virtual objects (i.e environment and JBV). In this way we can avoid propagating the particles in an unrealistic way beyond the existing boundaries (see Fig. 9). In fact, in this framework, each particle is reinterpreted inside Unity as a full-fledged holographic object which can interact with the other holograms. To associate a hologram with the capability of recognizing the occurrence of a collision with another hologram, we exploited a methodology based on the use of the so-called 'raycasting' technique, [17]. The latter enables the process of projecting a line, denoted as 'ray', from a desired starting position along a specific direction of the 3D space. Then, if the ray encounters an obstacle on its way, it returns the position where the collision has occurred.

4) CPF algorithm: Starting from the system described by (1), the proposed CPF algorithm, in closed loop, works as

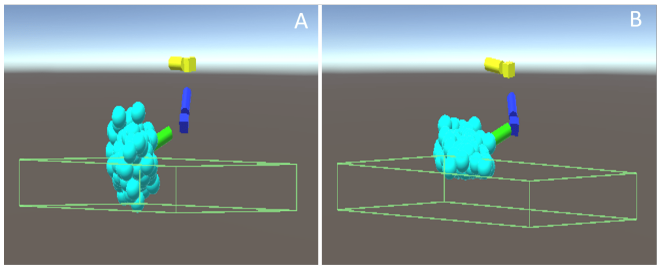

Fig. 9: Particle propagation through a volume without collision detection (A) and with collision detection (B)

follows.
Denote the vector of N particles at time step k as:

$$S_k = \{s_k^{(1)}, \ldots, s_k^{(N)}\} \tag{13}$$

and the vector of the particles weights at time step k as:

$$W_k = \{w_k^{(1)}, \ldots, w_k^{(N)}\} \tag{14}$$

Clearly, the samples represent hypotheses on the true value of the human wrist position.
At time step k=0:

- *Initialization phase*: when the human silhouette is completely tracked, compute the Euclidean distance, $s_{ew}^{(0)}$ between the true position of the elbow and the true position of the wrist. Draw N particles from the desired proposal distribution q and assign each of them with a weight equal to $1/N$.

For future time instants k=$1, \ldots, K$:

- *Propagation step*: propagate the samples a step further in time. A new set of particles $\bar{S}_k$ is obtained starting from S_{k-1}, where each particle is drawn from the proposal distribution, as follows:

$$\bar{s_k}^{(i)} \sim q(s_k^{(i)} | s_{0:k-1}^{(i)}, y_{0:k-1}) \tag{15}$$

  where, as done in [13], $q(s_k^{(i)} | s_{0:k-1}^{(i)}, y_{0:k-1})$ is set equal to $p(s_k^{(i)} | s_{k-1}^{(i)})$.

- *Collision detection check*: project a raycast from the true elbow position, E_k, towards the direction of the $i\text{-}th$ candidate wrist position $\bar{s_k}^{(i)}, \forall i = 1, \ldots, N$. Then, for each sample, if the raycast related to the $i\text{-}th$ particle is colliding in position χ_k^i with one of the virtual constraints, i.e the JBV or the surfaces of the virtualized environment, then $\bar{s_k}^{(i)} = \chi_k^i$.

- *Measurement update*: the new measurement y_k is retrieved by the sensor;

- *Weights computation*: update the weights of the particles as follows:

$$\begin{cases} w_k^{(i)} = p(y_k | s_k^{(i)}) w_{k-1}^{(i)}, & \text{if } \bar{s_k}^{(i)} \neq \chi_k^i \\ w_k^{(i)} = 0, & \text{if } \bar{s_k}^{(i)} = \chi_k^i \end{cases} \tag{16}$$

  where $p(y_k | s_k^{(i)})$ is the likelihood function which represents the probability that, given the sample $s_k^{(i)}$, the output is the measured one, i.e y_k. As suggested in [13], the likelihood function is shaped so as to include the constraint related to the skeletal distance.

- *Weights normalization*: normalize the weight of each particle with respect to the sum of the weights.
- *Bootstrap resampling*: draw a new set of N samples from the distribution obtained at the previous step, which is proportional to the values of the weights. Assign each sample with a weight equal to $1/N$.

The result of this procedure is that the volume where the particles are allowed to propagate is strictly reduced, as reported in Fig. 10. The collision detection check based on the raycasting technique turned out to be fast enough, in fact the projection of 500 rays increases the computational time of less than 1 ms.

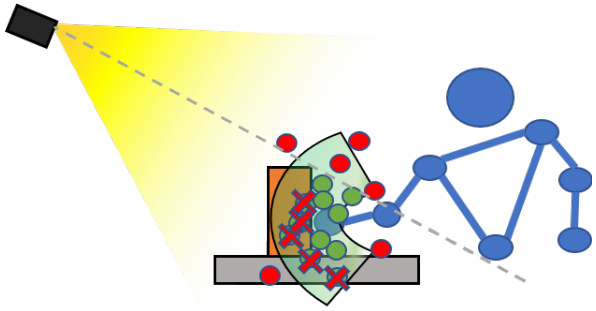

Fig. 10: Result of the tests in case of occlusion of the wrist pose: the spherical crown allowed by the 'skeletal distance' limitation is cropped to a section of an arc, due to the boundaries of the joint variables. The particles (red) which are visible to the Kinect camera, are discarded. Those representing colliding positions (marked with red crosses) are even eliminated

V. Experiments

We evaluated the performance of the proposed CPF algorithm in a realistic assembly task. The task consisted in assembling the components of an industrial emergency stop button. During the execution of the task the human operator was required to perform a certain sequence of reaching motions toward a predefined set of target positions, some of which caused the occurrence of the occlusion of the human wrist position from the perspective of the Kinect camera. The experimental set-up was composed by a dual-arm cobot (ABB YuMi), a Kinect camera, used to monitor the operator's silhouette and a MR headset, worn by the user. All these devices communicate the acquired data to a CPU which runs the CPF described in Section IV. A layout of the workstation, from the user's perspective, is displayed in Fig. 4B. We assume that in a real industrial scenario, two types of occlusions can occur (see Fig. 11):

- *static*: when the object occluding the human wrist is

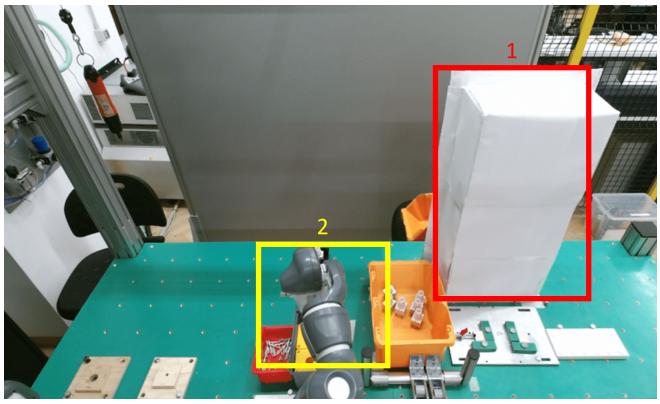

Fig. 11: Representation of a static occluding object (1) and of a dynamic occlusion (2) in a collaborative workstation from the perspective of the Kinect camera

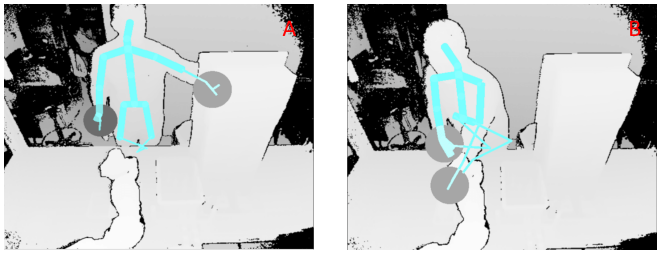

Fig. 12: Occlusions displayed in the Kinect depth map

due to the intrinsic geometry of the environment and remains a fixed element of the workspace, which, in our framework, can be mapped a priori during the spatial mapping phase. This is the case of the shelf reported in Fig. 11.

- *dynamic*: when the occlusion of the human wrist is due to an object that moves dynamically during the various phases of the assembly task; thus, it cannot be mapped a priori. This occurrence can be generated, for instance, by the motion of the robot arm which could interpose between the camera and the user or by a self-occlusion of the operator.

During the execution of the assembly task the operator's wrist is occluded when it inserts his/her hand into the concave surfaces of the shelf where some components needed to build the button are located. In addition, at certain time instants, his/her wrist can be occluded by the robot if the human is heading towards one of the boxes laying on the table (see Fig. 11) and the robot is moving its arm in the neighbourhood of those boxes. A picture describing the occurrence of these occlusions is reported in Fig. 12. To test the performance of our algorithm, we decided to adopt the following evaluation metrics:

- *propagation volume of the particles*: this quantity is described in terms of confidence ellipsoid. Indeed, the volume of the ellipsoid containing the 95% of the particles is computed and it is used to characterize the volume of the particles dispersion, which represents an indicator of the uncertainty in the estimation process.
- *estimation error*: it represents the distance between the estimate returned by the proposed CPF algorithm and a reliable measurement of the wrist position. To perform this test and obtain the measurement's ground truth, we removed the occlusion and we stored the wrist position returned by the Kinect. Then, we insert again the occluding object, and we evaluate the estimate returned by the algorithm.

To have a complete overview of the performance of our CPF algorithm, we compared the following three formulations of the CPF algorithm:

- 'Original': refers to the formulation of the CPF presented in Section III;
- 'Joint': refers to a formulation of the CPF which coincides with the aforementioned 'Original' algorithm to which the JBV and the avateering constraints have been added;
- 'Complete': refers to the novel formulation of the CPF algorithm that includes also the virtual model of the workspace. This formulation coincides with the CPF technique presented throughout Section IV.

We performed 17 trials of the same assembly task per each considered technique. The results obtained in terms of propagation volume (see Fig. 13) and in terms of estimation error (see Fig. 14) confirmed our expectations. In fact, Fig. 13 shows a relevant reduction (about one order of magnitude) of the volume dispersion of the particles when the JBV constraint is included as additional constraint to the 'Original' formulation. The uncertainty of the estimate is further reduced by the constraint related to the geometries characterizing the environment. In particular, this latter constraint reduces the variance of the volume, thus increasing the robustness of the proposed algorithm. For what concerns the estimation error, a similar conclusion can be drawn: when the JBV constraint is included, the estimation error decreases by 21.7% with respect to the 'Original' CPF. While, the 'Complete' CPF further enhances this result, by reaching a reduction of 28.3% with respect to the 'Original' CPF.

VI. Conclusions

In this paper, we proposed a novel constrained particle filtering technique that allows to manage effectively occlusion issues occurring in collaborative industrial frameworks. To limit the propagation of the particles and the uncertainty of the estimate, we included in the PF formulation the constraints related to the geometry of the real environment where the operator works, and those describing the boundaries of the human joints motion. To do that, we merged the data retrieved by the Kinect with the ones acquired by the MR headset. This allowed to interpret the particles of the PF as holograms that are propagated from a holographic model of the human body into a virtual replica of the user workspace, and to check if and where they are colliding with these virtual constraints, so as to better shape the occlusion. The data coming from the Kinect camera and from the HoloLens are related to each other, based on a custom marker detection

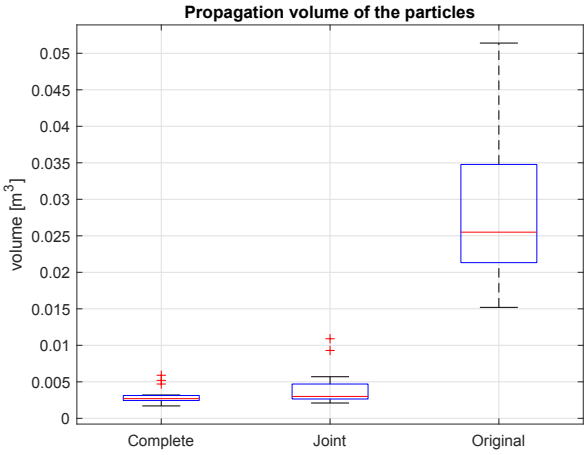

Fig. 13: Boxplot describing the distribution of the propagation volumes of the particles according to the three formulations of the CPF

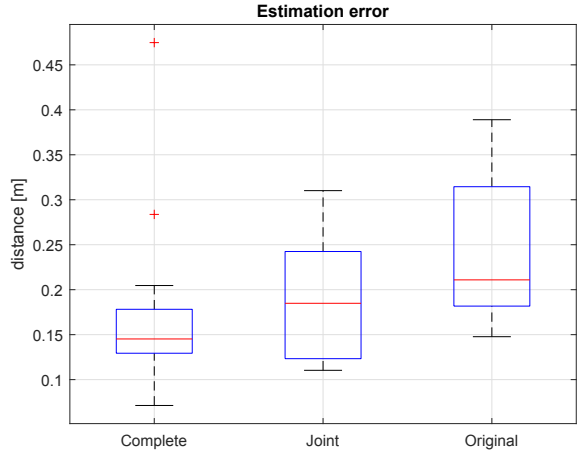

Fig. 14: Boxplot describing the distribution of the estimation error according to the three formulations of the CPF

algorithm. Even though the architecture of the proposed approach is more complex than that of the state-of-the-art methods, since it involved the presence of a wearable MR camera in addition to the traditional RGB-D camera, it resulted more effective at managing occlusions. Indeed, the performance of the proposed CPF technique turned out to be quite satisfactory, in terms of reduction, with respect to the state-of-the-art methods, of the volume of propagation of the particle and of the estimation error.

REFERENCES

[1] A. Hentout, M. Aouache, A. Maoudj *et al.*, "Key challenges and open issues of industrial collaborative robotics," in *2018 The 27th IEEE International Symposium on Workshop on Human-Robot Interaction: from Service to Industry (HRI-SI2018) at Robot and Human Interactive Communication. Proceedings. IEEE*, 2018.

[2] A. M. Zanchettin and P. Rocco, "Probabilistic inference of human arm reaching target for effective human-robot collaboration," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 6595–6600.

[3] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, "A survey on human motion analysis from depth data," in *Time-of-flight and depth imaging. sensors, algorithms, and applications*. Springer, 2013, pp. 149–187.

[4] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys, "Accurate 3d pose estimation from a single depth image," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 731–738.

[5] F. Flacco and A. De Luca, "Real-time computation of distance to dynamic obstacles with multiple depth sensors," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 56–63, 2016.

[6] N. Chen, Y. Chang, H. Liu, L. Huang, and H. Zhang, "Human pose recognition based on skeleton fusion from multiple kinects," in *2018 37th Chinese Control Conference (CCC)*. IEEE, 2018, pp. 5228–5232.

[7] M. Ragaglia, A. M. Zanchettin, and P. Rocco, "Trajectory generation algorithm for safe human-robot collaboration based on multiple depth sensor measurements," *Mechatronics*, vol. 55, pp. 267–281, 2018.

[8] L. Turner and C. Sherlock, "An introduction to particle filtering," *Lancaster University, Lancaster*, 2013.

[9] 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, 2002.

[10] S. Thrun, "Particle filters in robotics," in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 511–518.

[11] A. Yoshida, H. Kim, J. K. Tan, and S. Ishikawa, "Person tracking on kinect images using particle filter," in *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*. IEEE, 2014, pp. 1486–1489.

[12] Y. Xu, K. Xu, J. Wan, Z. Xiong, and Y. Li, "Research on particle filter tracking method based on kalman filter," in *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. IEEE, 2018, pp. 1564–1568.

[13] A. Casalino, S. Guzman, A. M. Zanchettin, and P. Rocco, "Human pose estimation in presence of occlusion using depth camera sensors, in human-robot coexistence scenarios," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 6117–6123.

[14] K. Apostolakis and P. Daras, "Natural user interfaces for virtual character full body and facial animation in immersive virtual worlds," vol. 9254, 08 2015, pp. 371–383.

[15] R. M. Salinas, "Aruco: An efficient library for detection of planar markers and camera pose estimation," 2019.

[16] J. Lenarcic and A. Umek, "Simple model of human arm reachable workspace," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 24, no. 8, pp. 1239–1246, Aug 1994.

[17] T. Schroeder, "Collision detection using ray casting," *Game Developer*, vol. 8, no. 8, pp. 50–56, 2001.