3D Hand Tracking in the Presence of Excessive Motion Blur

Gabyong Park, Antonis Argyros, Juyoung Lee, and Woontack Woo, Member, IEEE

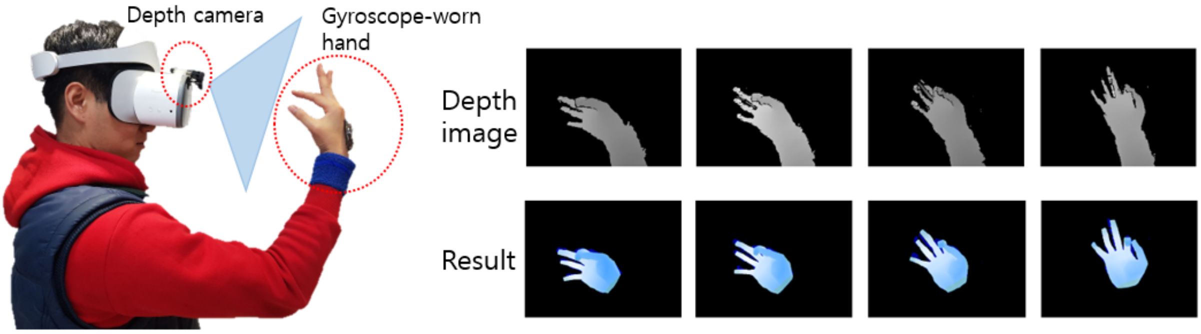


Fig. 1. Our solution supports 3D hand tracking in the presence of excessive motion blur. Left: the depth camera captures the gyroscope-worn hand and the gyroscope measures the hand's angular velocity. Right: in the case of fast moving hands, the depth image is distorted due to motion blur. The proposed method tracks successfully the articulation of the hand despite fast hand rotations.

Abstract—We present a sensor-fusion method that exploits a depth camera and a gyroscope to track the articulation of a hand in the presence of excessive motion blur. In case of slow and smooth hand motions, the existing methods estimate the hand pose fairly accurately and robustly, despite challenges due to the high dimensionality of the problem, self-occlusions, uniform appearance of hand parts, etc. However, the accuracy of hand pose estimation drops considerably for fast-moving hands because the depth image is severely distorted due to motion blur. Moreover, when hands move fast, the actual hand pose is far from the one estimated in the previous frame, therefore the assumption of temporal continuity on which tracking methods rely, is not valid. In this paper, we track fast-moving hands with the combination of a gyroscope and a depth camera. As a first step, we calibrate a depth camera and a gyroscope attached to a hand so as to identify their time and pose offsets. Following that, we fuse the rotation information of the calibrated gyroscope with model-based hierarchical particle filter tracking. A series of quantitative and qualitative experiments demonstrate that the proposed method performs more accurately and robustly in the presence of motion blur, when compared to state of the art algorithms, especially in the case of very fast hand rotations.

Index Terms—3D hand tracking, 3D hand pose estimation, sensor fusion, depth camera, gyroscope, motion blur

1 Introduction

Articulated hand motion tracking is a widely studied problem in computer vision, Virtual Reality (VR) and Augmented Reality (AR). Most of the contemporary research has focused on tracking hands that move relatively slowly and smoothly. However, in several scenarios such as sign language understanding, object manipulation, playing musical instruments, etc, hands move very fast. Fast hand motions introduce two important problems. First, they break the temporal continuity of 3D hand poses, an assumption several tracking approaches rely upon. Moreover, they introduce image artifacts such as motion blur, that degrade considerably the quality of the available observations. Despite its importance, tracking the articulation of hands in the case of rapid hand motions is very much under-explored. In this research, we study the problem of tracking fast-moving hands in order to support VR and AR applications that involve rapid hand motions.

Most current approaches to track hand articulations can be categorized into generative [21, 23, 27, 28, 36] and discriminative ones [4, 6, 10, 24, 40, 42–44, 49]. Generative methods track the hand pose online by optimizing the fit of a virtual 3D hand model to the actual observations. However, in the case of fast hand motion, they are likely to loose track of the hand as they rely on the solution in the previous frame to explore a small part of the hands' configuration space. Discriminative methods do not rely on temporal continuity and are able to estimate the solution in a single frame. However, their solution is very much affected by the quality of the input images that is severely degraded in the case of motion blur. Recently, hybrid approaches [14, 29, 31, 38, 45] have been studied to combine the merits of both strategies. For example, even though the hand rotates quickly, a generative method can reinitialize the hand pose based on the output of a discriminative method [31, 38, 45] or based on the detection of fingertips [34]. This might be a solution to the temporal continuity problem, but not a solution to the motion blur problem. For example, if part of the finger is distorted in the image, fingertip detection cannot be performed. To directly address motion blur, image deblurring is a possible alternative. However, most relevant algorithms [5, 16, 35] address an RGB image rather than a depth image, and are time-consuming, which makes them inappropriate for applications requiring real-time 3D hand tracking. Although there are several works [13, 18, 37] to deblur a depth image, the application is limited to specific time-of-flight cameras rather than general depth cameras. In addition, the publicly available datasets [9, 50] for the discriminative approaches do not include the ground truth on frames during fast motion.

To address the hand tracking problem in the presence of excessive motion blur we utilize both a depth camera and a gyroscope. The use of the depth camera is justified because the depth map of a hand

- *G. Park, J. Lee, and W. Woo are with KAIST, Daejeon, S. Korea. E-mail: {gypark,ejuyoung,wwoo}@kaist.ac.kr*
- *A. A. Argyros is with the Institute of Computer Science, FORTH, Heraklion GR-700 13, Greece and with the Computer Science Department, University of Crete, GR-700 13, Greece. E-mail: argyros@ics.forth.gr*

Manuscript received 10 Sept. 2019; accepted 5 Feb. 2020.
Date of publication 18 Feb. 2020; date of current version 27 Mar. 2020.
Digital Object Identifier no. 10.1109/TVCG.2020.2973057

model can be easily rendered based on a hand mesh, hand pose, and camera information. Thus, the discrepancy between a hand articulation hypothesis and the actual observations can be measured and used in the objective function of an optimization process that drives 3D hand pose estimation. Using RGB and IR images for the same purpose requires much more information that is difficult to have (light source, material properties, texture etc). Furthermore, the variability of an RGB image due to the lighting conditions (e.g., strongly illuminated vs dark scenes) might lead to tracking instabilities. However, depth information might be affected by motion blur that occurs as objects in a scene move over the period of exposure determined by the camera shutter speed. Although recently developed depth cameras with global shutter alleviates motion blur, images acquired by ordinary depth cameras with a rolling shutter suffer from severe artifacts due to fast motion, as shown in Fig. 1. In such a situation, a gyroscope can provide valuable hand motion information as it does not suffer from fast motion and is not affected by visual occlusions. At the same time, a gyroscope that is worn in the back side of the palm is much less invasive compared to e.g., reflective markers placed on the hand's fingertips.

In this paper we focus on combining a gyroscope attached to the hand and a depth camera for hand articulations tracking that is robust to motion blur. We present a novel sensor-fusion method for tracking hand articulations. Recently, sensor-fusion algorithms have been proposed for the problem of tracking the 3D pose of the human body [11,12,22,33,47,51]. Our approach consists of two major steps: sensor calibration and 3D hand tracking. The calibration step estimates the time delay and pose offset between the depth camera and the gyroscope in an offline process. Since each sensor uses a different timeline, the time delay from the gyro's timeline to the camera's one is estimated to achieve their synchronization. Besides, the coordinate systems of the two sensors differ because the gyroscope is attached to the user's hand, and the camera observes the hand from a point of view external to the hand. The tracking step estimates the hand pose based on hierarchical particle filter in an online fashion. Initially, sampling for hand orientations (called slerp-based sampling) is performed by spherical interpolation of the trajectory among the poses estimated by the gyroscope and the orientation estimated for the previous frame. The sampled particles (candidate hand pose orientations) are evaluated based on their likelihood which is computed by considering the hand depth image and the calibrated gyroscope information. Following that, the pose of each part of the hand is hierarchically estimated based on a particle filtering approach.

The proposed method is evaluated by quantitative and qualitative experiments based on datasets we compiled. It should be noted that existing datasets are not adequate for our purposes because they do not cover fast hand motions and motion blur and do not provide simultaneously data from a depth camera and a gyroscope. We use the acquired dataset as well as synthetic data to assess the accuracy of calibration and to support the ablative study of the main components of the proposed algorithm. Finally, a quantitative/qualitative comparison with state-of-the-art methods is conducted. For the evaluation of the tracking accuracy, we exploit the ground truth based on infrared (IR) images since the IR image is not distorted despite fast motion. The experimental results demonstrate that when hand motion is rapid, the proposed tracking method exhibits superior performance to other tracking algorithms.

Overall, the contribution of this research can be summarized as follows:

1. We present the first method that performs 3D hand tracking of a fast moving hand by explicitly dealing with hand observations that are degraded due to motion blur. This is achieved based on the fusion of gyroscopic and visual information.

2. *Sensor calibration:* We estimate time delay and pose offset between a depth camera and a gyroscope and we evaluate the calibration result based on synthetic and real data.

3. *Sensor-fusion based 3D hand pose tracking:* The rotation information of the calibrated gyroscope is fused to generative tracking based on hierarchical particle filtering. Slerp-based sampling and gyro-based regularization are applied to the particle filter to address fast motion.

4. *Method evaluation in the case of fast moving hands:* We compile new datasets to evaluate the proposed method and we show experimentally that it exhibits superior performance compared to state-of-the-art methods.

2 Related work

A number of methods have been proposed for solving the problem of 3D hand pose estimation. According to Erol et al. [8], they can be divided into generative and discriminative approaches. Additionally, we discuss some representative hybrid methods as well as methods for sensor-fusion-based tracking.

Discriminative Approaches The discriminative approaches estimate the hand joint 3D locations in a single frame from RGB or depth input. Such methods require a classifier/regressor that is trained on a large dataset. Some works [40,42–44] use Random Decision Forests to estimate joint positions in a single frame, but are difficult to generally estimate hand pose, requiring hand-crafted features to describe the hand pose. Recently, the works [4,6,10,24,49] achieved significant accuracy without hand-crafted features based on deep learning. Methods that are based on Convolutional Neural Networks (CNNs) also fall in this category. The generalization beyond their training set is challenging since a large number of parameters need to be learned. Moreover, there are significant difficulties in obtaining large, accurately annotated datasets. The works [9,50] have produced large datasets of real hand depth images annotated with joint locations based on magnetic sensors. Although they cover various hand postures, it is not easy to sample evenly the space of hand poses. Datasets based on a rendered hand model can alleviate this problem. The work [25] proposed an approach that generates synthetic hand images that follow the same statistical distribution as real-world hand images. Although the current discriminative approaches perform well when trained on a large dataset, they do not perform equally well in the case of fast hand motion and motion blur, a situation that is treated successfully by the method we propose in this paper.

Generative Approaches The generative, model-based approaches estimate the 3D hand pose by optimizing the fit of a rendered 3D hand model to the observed data. The approaches fit a hand model constructed based on geometric primitives to RGBD data with various optimization and filtering methods. For example, the methods of Oikonomidis et al. [27,28] use Particle Swarm Optimization (PSO), the works [21,36] use Hierarchical Particle Filter (HPF), and other works [23,41] use Iterative Closet Point (ICP). Although they exhibit adequate performance, such methods rely highly on the parameters of the selected hand model and on the initialization of the starting pose (usually the solution of the previous frame). Methods that adapt the hand model's shape in an online fashion have already been proposed [20,46]. This improves tracking accuracy and removes some of the complexity of the manual fine tuning of the hand model.

Hybrid Approaches The tracking accuracy of the generative approaches drops in the case of rapid hand motion. Regarding the selection of a proper starting pose for optimization, learning-based methods or fingertip detection are additionally applied, resulting in hybrid approaches that have both generative and discriminative elements. The methods [31,38,45] generated multiple hypotheses from the previous solution and the pose classified by the trained model, and the work of Qian et al. [34] reinitialized the hand pose based on finger detection. They can reinitialize the hand pose even when the tracking fails. In the works [29,32] the 3D hand pose is estimated by a generative approach that operates on the joint locations that are estimated by a robust discriminative approach. In any case, the tracked hand pose would not be accurate if pose re-initialization performs inadequately. This happens commonly when the obtained image is severely distorted due to motion blur. The work of Mueller et al. [26] optimizes both 3D hand pose and hand shape for interacting hands after correspondence regression. However, although it successfully handles complex hand-hand interactions, it also suffers from motion blur induced due to fast hand motion.

Sensor Fusion Approaches IMU (Inertial Measurement Unit) is widely used in combination with a visual sensor for human body pose estimation. The works [33,47] proposed sensor fusion algorithms that combine inertial data and multi-view markerless motion capture data for full-body tracking. In contrast, Helten et al. [12] proposes a hybrid method with a single depth camera and inertial sensors. This requires a less complicated hardware setup, but the inertial sensors are only used to query poses in a database rather than complementing the information provided by the visual sensor. The work of Malleson et al. [22] proposed a real-time optimization-based framework for full-body pose estimation, which incorporates constraints from the IMUs, cameras and a prior pose model. Zheng et al. [51] proposed an algorithm for non-rigid surface reconstruction for fast motions and challenging poses with severe occlusions, combining a single depth sensor and sparse IMUs. Gilbert et al. [11] also proposed the method of fusing visual and inertial information for 3D human pose estimation based on deep learning.

While gyroscopes are widely used for human body pose estimation, their use for hand pose estimation has not been investigated yet. As an example of applying the sensor-fusion method to hand tracking, Kim et al. [15] uses the pose estimated by an IMU sensor to assist model-based tracking. This results in a more robust performance compared to using vision-based hand tracking, only. However, in that work, the IMU-based rotation replaces the one estimated from the visual information, rather than being fused with it, as in our work. To the best of our knowledge, there has been no previous works that used gyroscopic information that is fused with visual model-based tracking for pose estimation of fast moving hands.

3 Methodology

Our goal is to track the 3D hand pose which is modeled as a 27-dimensional vector, given gyroscopic and RGB-D camera inputs. Our approach consists of two major steps: offline calibration and online sensor-fusion hand tracking. An overview of the proposed method is shown in Fig. 2. Input RGB-D images from the camera and angular velocity from the gyroscope are preprocessed. During offline calibration, the time delay and pose offset between the camera and gyroscope are estimated. Then, for online sensor-fusion hand tracking, the calibrated gyroscopic information is efficiently fused to a Hierarchical Particle Filter (HPF) tracking method to estimate the 3D hand pose.

Preprocessing: We segment the hand by setting that the user wears a blue wristband. Hand segmentation is not the focus of this work and can be achieved by several other methods [1,48]. We choose to perform hand segmentation as in the work of Tkach et al. [46]. Specifically, the position of the wristband is estimated by color segmentation in HSV space, the 3D points in the proximity of the wristband are identified and their principal axis is calculated. The axis is then used to segment the hand part.

The angular velocity $\vec{w}(t)$ measured by the gyroscope is converted to the quaternion representation as follows:

$$Q(\vec{w}(t)) = \left(cos\frac{|\vec{w}(t)dt|}{2}, \frac{\vec{w}(t)}{|\vec{w}(t)|} sin\frac{|\vec{w}(t)dt|}{2} \right), \tag{1}$$

where dt is the sampling time of the gyroscope.

Thereby, the rotation at time t relative to an initial pose is calculated as the product of quaternions from the initial time step to the present time t, that is, as $\prod_{i=0}^{t} Q(\vec{w}(i))$.

Hand model: We adopt the parametric hand model for both calibration and tracking. The kinematic model of the hand is represented as a vector of 27 parameters modeling 26 degrees of freedom (DoFs), consisting of global 3D translation, global 3D rotation encoded as a quaternion, and 3D rotation of the fingers. For each finger, three joints are modeled; one for the saddle joint at the base, and two for the two remaining hinge joints. Each saddle joint has two DoFs, and each hinge joint has one DoF. To calculate the discrepancy between a hand pose hypothesis and a given set of visual observations, the hand model is rendered with the shader of the OpenGL pipeline by taking into account the hypothesized hand pose and the camera calibration information.

3.1 Calibration

Fig. 3 illustrates the relationship between the coordinate systems of the camera and the gyroscope. We individually estimate the time delay and the pose offset between the gyroscope and the depth camera. The calibration is achieved by exploiting the hand orientations estimated by each of the sensors. We first calculate the pose offset and then estimate the time delay. For the estimation of pose offset, we gather the orientations by a 3D hand tracker in depth images and those of gyroscope. The hand orientation $h_c(t)$ relative to the reference frame of the depth camera is obtained by the solution in the work [21], and the gyro rotation $h_g(t)$ relative to the reference frame of the gyroscope is calculated by $h_g(0)\prod_{i=0}^{t} Q(\vec{w}(i))$ where $h_g(0)$ is initialized to an identity quaternion. For the estimation of time delay, we gather the hand depth images $D_o(0..t)$ and those $D_r(0..t)$ rendered from the gyro sensor. For D_r, we fix the remaining parts such as the hand's translation and the fingers' rotation so that the only orientations of the hand are considered. The proposed calibration approach is presented in the following in more detail.

3.1.1 Pose offset

The relation between h_c and h_g at a time step t is defined by the two offset rotations (o_1, o_2) based on:

$$h_c(t) = o_1 h_g(t) o_2. \tag{2}$$

From Eq. 2, the rotation $\Delta h_c(t)$ relative to the initial pose $h_c(0)$ is derived by:

$$\begin{aligned} \Delta h_c(t) &= h_c^{-1}(0) h_c(t) && (3) \\ &= (o_2^{-1} h_g^{-1}(0) o_1^{-1})(o_1 h_g(t) o_2) && (4) \\ &= o_2^{-1} h_g^{-1}(0) h_g(t) o_2 && (5) \\ &= o_2^{-1} \Delta h_g(t) o_2. && (6) \end{aligned}$$

As derived in Eq. 6, we can simplify the problem of estimating two pose offsets (o_1, o_2) to the one of estimating a pose offset o_2. Next, we express Eq. 6 as a homogeneous 4×4 matrix based on:

$$\begin{aligned} o_2 \Delta h_c(t) - \Delta h_g(t) o_2 &= 0 && (7) \\ H_c o_2 - H_g o_2 &= 0 && (8) \\ (H_c - H_g) o_2 &= 0 && (9) \\ H o_2 &= 0, && (10) \end{aligned}$$

where H_c is a 4×4 matrix corresponding to $\Delta h_c(t)$ and H_g is a 4×4 matrix corresponding to $\Delta h_g(t)$.

The solution of Eq. 10 such that $o_2^T o_2 = 1$ is in the null space of the matrix H. Using the Singular Value Decomposition (SVD) of the matrix H, we get $H = UDV^T$ where U, D and V are 4×4 square matrices. Finally, the solution can be obtained as a row vector of V corresponding to the smallest singular value of D. The solution o_2 is unique if the matrix H is a 8×4 matrix constructed from the accurate two pairs of Δh_c and Δh_g where the last element (the smallest element) in the diagonal of D is zero. However, in a real problem, they are contaminated by noise since $\Delta h_c(t)$ is estimated by the hand tracker and $\Delta h_g(t)$ is calculated by the multiplication of the angular velocities, which accumulates error. To alleviate this problem, we gradually construct the matrix H from more than two pairs with the following conditions:

1. We construct the matrix H when the hand tracking result is likely to be adequate by checking the depth energy E_d in Eq. 19.

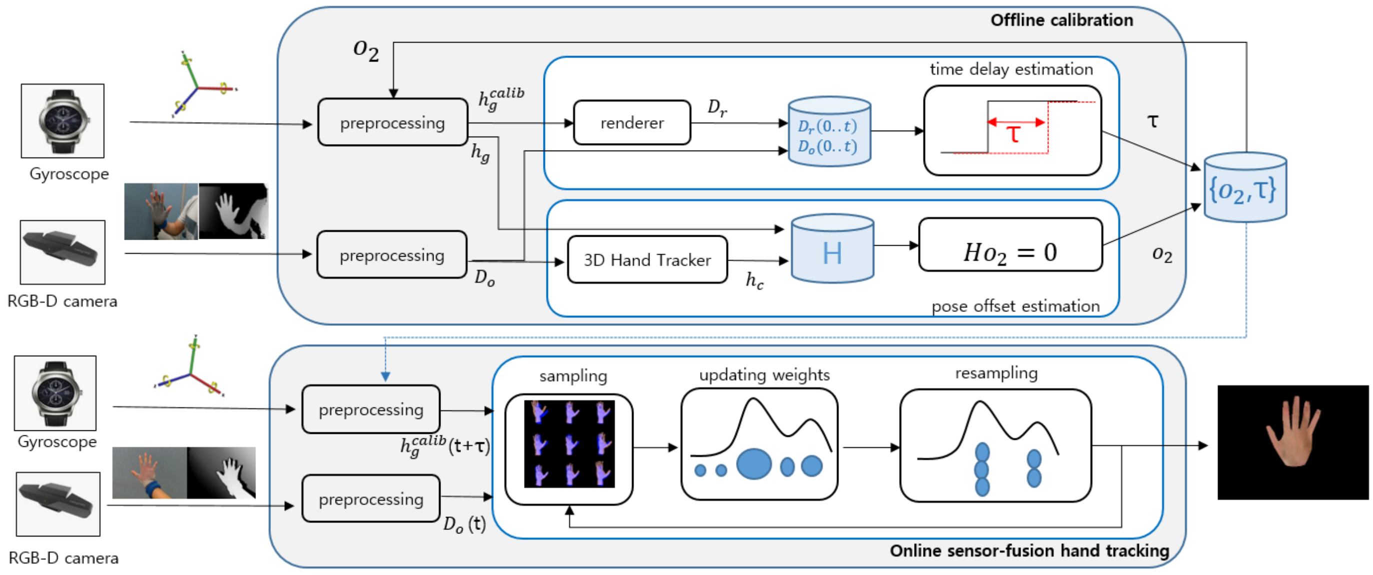


Fig. 2. Overview of the proposed method as described in Section 3. Input RGB-D images and angular velocity are preprocessed for offline calibration and online sensor-fusion hand tracking. For offline calibration, the pose offset o_2 and time delay τ are estimated (see Section 3.1). The calibrated information is fused to online sensor-fusion hand tracking (see Section 3.2).

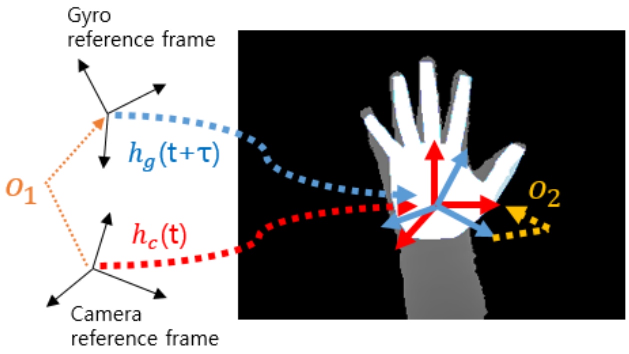


Fig. 3. Time and pose offset between gyroscope and camera.

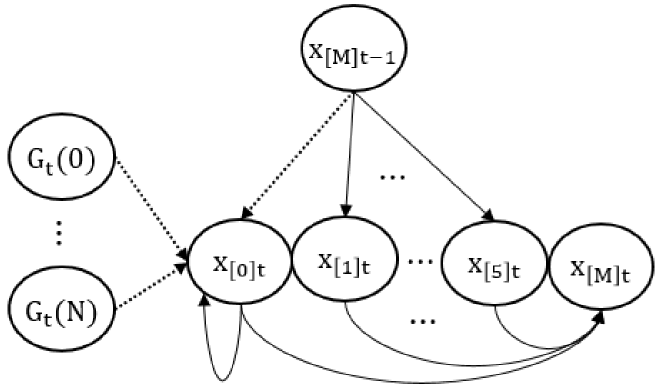


Fig. 4. Dynamic Bayesian network of the proposed model. We omitted observations of the depth image and the gyro's angular velocity for clarity. The dot arrow represents the slerp-based estimation for a palm.

2. To minimize the accumulation error of the gyroscope, we initialize the data pair ($\Delta h_g(t)$,$\Delta h_c(t)$) by calculating it from the point when H is constructed.

3. To minimize the time delay issue, the matrix H is constructed when the hand does not move.

Consequently, the calibrated gyro rotation h_g^{calib} with the estimated pose offset is calculated by $h_c(0)o_2^{-1}\Delta h_g(t)o_2$.

3.1.2 Time delay

To estimate the time delay τ between the gyroscope and the depth camera we seek for the time offset of the two signals that maximizes their correlation. In a preliminary experiment, we tried to estimate the time delay in a straightforward manner, that is by calculating the correlation between the hand rotation as estimated by the camera and by the gyroscope during fixed time. However, we observed that the estimated delay was not adequately accurate since the calibration requires highly accurate and fast-tracking performance in one frame unit. Therefore, we used the depth images from the orientations rather than directly comparing orientations. First of all, hand pose is initialized by the 3D hand tracker of Makris et al. [21] based on a depth image. Since only the hand rotation affects the variable of the gyro, the remaining parameters (the hand translation and the fingers' rotation) are fixed as the initial pose. Following that, we record the depth images $D_o(0,..,T)$ from the depth camera and the angular velocities $\vec{w}(0,..,T)$ of the gyroscope while the hand simply turns left and right. The depth image D_r is rendered by the rotation h_g^{calib} of the gyroscope calibrated from the pose offset. The optimal time delay is estimated by finding the minimal average of the pixel-by-pixel errors between depth images $D_o(0,..,T)$ and the corresponding rendered depth maps $D_r(h_g^{calib}(0,..,T))$ based on:

$$\tau^* = arg\min_{\tau} \sum_{t=0}^{T} |D_o(t) - D_r(h_g^{calib}(t+\tau))|/N, \qquad (11)$$

where N is the number of non-zero pixels in the depth image.

3.2 Sensor-fusion hand tracking

We adapt Hierarchical Particle Filter (HPF) for sensor-fusion hand articulations tracking. The work of Makris et al. [21] firstly proposed the HPF for hand articulations tracking. We describe the HPF tracking framework of Fig. 4. We define the full state x_t of the hand pose at a time step t with the auxiliary models $x_{[0:5]t}$ to track distinct parts

Algorithm 1 HPF for Sensor-fusion Hand Tracking

```
Input: ${x^{(j)}_{[0:M]t-1}, w^{(j)}_{t-1}}^N_{j=1}, z_t$
Ouput: ${x^{(j)}_{[0:M]t}, w^{(j)}_t}^N_{j=1}$
  //Slerp-based hand pose estimation for the palm.
  for each particle j=1 to N do
    Sample $x^j_{[0]t}$ from Eq. 16.
    Update its weight $w^j_t$ using Eq. 22.
  end for
  Normalize the particle weights.
  Resample the particle set according to its weights.
  //Hierarchical hand pose estimation.
  for each model i=0 to M do
    for each particle j=1 to N do
      Sample $x^j_{[i]t}$ from Eq. 17 and Eq. 18.
      Update its weight $w^j_t$ using Eq. 22.
    end for
    Normalize the particle weights.
    Resample the particle set according to its weights.
  end for
```

of the hand and the main model $x_{[M]t}$, and the trajectory $G_t(0..N)$ of the gyro pose to guide 3D orientation of the hand. The framework follows the Bayesian approach for tracking the hand pose. We denote the full state sequence $x_{0:t} = \{x_0 ... x_t\}$ and the the set of all observations $z_{0:t} = \{d_0 ... d_t, \vec{\omega}_0 ... \vec{\omega}_t\}$ where d is the preprocessed depth image and $\vec{\omega}$ is an angular velocity from a gyroscope. Under the assumption by a single Markov chain, the posterior is formulated based on:

$$p(x_{0:t}|z_{1:t}) = p(x_{0:t-1}|z_{1:t-1}) \prod_i \frac{p(z_t|x_{[i]t})p(x_{[i]t}|Pa(x_{[i]t}))}{p(z_t|z_{1:t-1})}, \quad (12)$$

where $Pa(x_{[i]t})$ represents the parent nodes of $x_{[i]t}$. The HPF approximates the posterior by propagating hypotheses (particles) based on the importance sampling method [2]. The particles are generated from the proposal distribution based on:

$$q(x_{0:t}|z_{1:t}) = q(x_{0:t-1}|z_{1:t-1}) \prod_i q(x_{[i]t}|Pa(x_{[i]t})), \quad (13)$$

where $q(x)$ is proposal distribution of x. The particles are weighted by the importance weights based on:

$$w_t = \frac{p(x_{0:t}|z_{1:t})}{q(x_{0:t}|z_{1:t})} \quad (14)$$

$$\propto w_{t-1} \prod_i \frac{p(z_t|x_{[i]t})p(x_{[i]t}|Pa(x_{[i]t}))}{q(x_{[i]t}|Pa(x_{[i]t}), z_t)}. \quad (15)$$

The steps to estimate the posterior at time t are shown in Algorithm 1, given the weighted particles $\{x^{(j)}_{[0:M]t-1}, w^{(j)}_{t-1}\}^N_{j=1}$ from the previous time and the observations z_t at time t. The algorithm sequentially updates the states by sampling particles and updating the weights. We first sample particles for a palm pose with slerp-based sampling, which are evaluated by observation likelihood. After normalizing the weights of the particles, particle resampling is performed as in the works [2, 17]. Next, each auxiliary models are sequentially updated as in the work [21]. The final solution is obtained by the weighted average of the main model particles.

3.2.1 State evolution

The state evolution for the HPF is conducted based on the Bayesian network, similarly to the work [21]. However, relying on only previous main model is not suitable for our purposes because the assumption of the temporal continuity is broken for the fast hand motion. Therefore, the palm model $\mathrm{x}_{[0]t}$ initially samples the particles from the trajectory of the calibrated gyro poses $G_t(0..N)$ and the palm part of the previous main model $\mathrm{x}_{[M]t-1}$. In our approach, we obtain the multiple gyro poses in a camera scene since the sampling frequency of the gyroscope is faster than that of the camera. The particles of the palm model are evenly sampled based on:

$$x_{[0]t} = slerp(a(x_{[M]t-1}, 0), G_t(0), .., G_t(N)), \quad (16)$$

where $a(x_{[M]t-1}, 0)$ is the palm part of the previous main model and $slerp(\cdot)$ performs spherical interpolation of quaternion among the poses. Since the gyroscope measures only rotation, the translation parameters follow the previous main model. We sample again the particles for the palm model from the state updated by the slerp-based pose estimation. This refines the estimation when the hand simultaneously moves during rotation and the actual hand pose is away from the trajectory of the gyro pose. The other auxiliary models (fingers) are sampled from the main model at the previous time step. The sampling is formulated based on:

$$p(x_{[i]t}|Pa(x_{[i]t})) = \begin{cases} N(x_{[i]t}; x_{[i]t}, \Sigma_{[i]}) & \text{if } i = 0 \\ N(x_{[i]t}; a(x_{[M]t-1}, i), \Sigma_{[i]}) & \text{if } i = 1, .., 5 \end{cases} \quad (17)$$

where the operator $a(x_{[M]t-1}, i)$ produces the part of the state of the main model, which represents the i-th auxiliary model, and N(x;m,Σ) expresses the normal distribution over x with mean m and the predefined diagonal covariance matrix Σ.

The main model is sampled from the updated auxiliary models at the present time step based on:

$$p(x_{[M]t}|Pa(x_{[M]t})) = N(x_{[M]t}; x_{[0:M-1]t}, \Sigma_M), \quad (18)$$

where $x_{[0:M-1]t}$ are the auxiliary models updated by previous steps. This searches for refinement of the hand pose in a full dimensional space (26 DoFs), given the concatenation of the updated sub-states corresponding to the M auxiliary models in lower dimensional spaces.

3.2.2 Observation likelihood

The sampled particles are evaluated based on the observation likelihood, which measures the fitting of the rendered hand model to the observations of the gyroscope and the depth camera. The input is the preprocessed depth image and the calibrated gyro pose. To calculate this likelihood, an objective function is defined as:

$$E = (1 - \alpha_g)E_d + \alpha_g E_g, \quad (19)$$

where E_d quantifies the depth discrepancy between visual observations and the rendered hand model, detailed in the work [21], and E_g is a gyro-regularization term. When the hand motion is fast, the depth image is blurred. In this case, adopting only E_d does not exhibit adequate performance due to the degraded depth image. Therefore, we additionally adopt the term E_g that serves to regularize the sampled particle x from the calibrated gyro pose. To calculate the term E_g, similarly to the work [19], we introduce two arbitrary unit length vectors because the quaternion q is the same as $-q$. The vectors rotated by the sampled particle and the calibrated gyro pose are compared in Euclidean space for similarity of the orientations based on:

$$E_g = \sum_{i=1}^{2} \frac{\beta \| p_r(\vec{u}_i) - h_g^{calib}(\vec{u}_i) \|_2^2}{2}, \quad (20)$$

where $\vec{u}_i$ is an arbitrary unit length vector and $p_r(\vec{u}_i)$ represents the vector rotated by the hand orientation corresponding to a sampled particle p_r and $h_g^{calib}(\vec{u}_i)$ is the vector rotated by the calibrated gyro orientation, and β is a empirically identified weight value.

The weight α_g is adapted according to the speed of hand motion. When the hand moves fast, the depth image is usually blurred. In this case, the depth term E_d is less effective than the gyro-regularization term E_g. Therefore, we empirically adapt the weight based on:

$$\alpha_g = \begin{cases} 0.2 + \frac{\Omega - 0.02}{0.1 - 0.02} \times 0.6, & \text{if } 0.02 < \Omega < 0.1 \\ 0.2, & \text{if } \Omega \leq 0.02 \\ 0.8, & \text{if } \Omega \geq 0.1 \end{cases} \tag{21}$$

where Ω is the sum of the quaternion-based distance starting the previous hand pose $a(x_{[M]t-1}, 0)$ to the end of the gyro pose $G_t(N)$ at the time step t, which reflects the rotational speed of the gyroscope at the time step t. The likelihood is then calculated by:

$$p(z|x) = exp\left(-\frac{E^2(z,x)}{2\sigma^2}\right). \tag{22}$$

4 Experiments

We performed extensive experiments to evaluate the performance of the proposed method and to compare it with state-of-the-art approaches.

Datasets: For the calibration experiments, we used a *synthetic dataset* compiled by rendering a hand model. Synthetic depth images were rendered by controlling the 26 DoFs of hand pose. The orientations corresponding to the synthetic gyroscope were obtained by spherical interpolation between the previous and the current pose of the rendered hand model. The number of samples depends on the sampling rate of the considered synthetic gyroscope. We assumed a sampling rate of 0.03 sec for the camera and of 0.01 sec for the synthetic gyroscope. For example, when a synthetic hand image is rendered at time t, the three orientations of the synthetic gyroscope are obtained at the time steps t, $t+0.01$ and $t+0.02$. Based on these data, we assessed quantitatively the calibration performance.

For the evaluation of tracking performance, we used two real datasets acquired by actual sensors. Specifically, the *real dataset* #1 was used for an ablation study and contains slow hand motions (100 frames) and fast hand motions with temporal discontinuities (865 frames), and very fast hand motions (188 frames). The *real dataset* #2 was used for comparison to state-of-the-art. This contains slow hand motions with temporal continuity and no motion blur (300 frames), and fast hand motions with temporal discontinuities (2070 frames), out of which 218 frames were classified as containing very fast motions. The classification was assisted by considering the magnitude of the angular velocities measured by the gyroscope. Frames are classified as very fast when the average L2-norm of angular velocities exceeds 6.66 rad/sec during the adjacent 3 frames. In this case, excessive motion blur is frequently caused. To the best of our knowledge, these are the first datasets of their kind and our study is the first to compare quantitatively/qualitatively hand-pose tracking performance in the case of motion blur.

We observed that an infrared image is not blurred despite fast hand motion and a reflective material exhibits high intensity in the IR image. We put reflective tape on the tip of the little finger, which reflects the error well when the hand is wrongly flipped. The 2D center position of the tape was estimated as ground truth based on a simple contour-based detection method.

Parameter settings: To estimate the time-delay and the pose-offset during calibration, α_g of Eq. 19 is zero since gyroscope and camera were not yet calibrated. Three gyro poses were used for slerp-based sampling and 64 particles were sampled in the state evolution. For the evaluation of the particles, β of Eq. 20 was set to 0.4.

Performance issues: Our method was evaluated on an Intel Core i7 4GHz with a single NVIDIA GTX1080-ti GPU. It ran on a 30Hz RGBD camera like Intel Realsense SR-300 and 100 Hz gyroscope of LG Watch Urbane W150 OEM. The angular velocity of the gyroscope is transferred to the hand tracking system through User Datagram Protocol (UDP) socket communication over WiFi. Regarding the computational complexity of the model, as we use a particle filter to track a hand model, analysis made by the works [3, 7, 39] could apply. Indicatively, our tracking method achieves real time performance (50Hz) under the above settings.

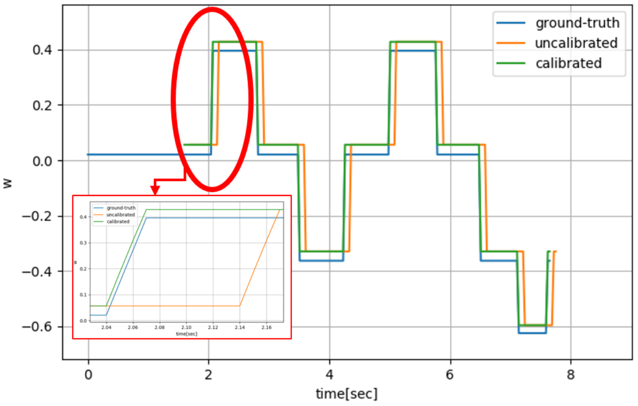


Fig. 5. Time delay experiment based on synthetic data. A time series of the real part of the quaternion representing hand orientation was estimated based on the proposed approach. The plot shows (a) the time series calibrated for the time offset based on the proposed method, (b) the relevant ground truth and (c) the measurements (prior to calibration) that correspond to a synthetic gyroscope with the predefined time delay. The calibrated time series that is estimated by our method exhibits the same timeline as the ground truth.

Trial	1	2	3	4	5	6	7
Time delay $[sec]$	0.03	0.01	0.03	0.04	0.02	0.04	0.04

Table 1. The consistency of the estimated time delay based on real data.

4.1 Evaluation of calibration

4.1.1 Time delay

In a first experiment, we evaluated our approach for estimating the time delay between the depth camera and the gyroscope. First, we verified our approach based on the *synthetic dataset*. Fig. 5 shows the estimated time delay based on synthetic data. We set ground-truth time delay as 0.1 sec. We plotted only the real part 'w' of the quaternion representing the hand orientation because the other imaginary components (x,y,z) showed a similar pattern. As shown in Fig. 5, the estimated time delay was the same to the ground truth (0.1 sec).

We also evaluated the consistency of the estimation of our approach based on real data. Unlike the evaluation based on the synthetic data, in real data ground truth is not available. Table 1 shows the consistency of the estimated time delay based on the real data. Unlike the synthetic experiment, we estimated slightly different time delay in every experiment. The optimal τ that minimizes Eq. 11 was estimated from 0.01 sec to 0.04 sec. The deviation is likely to come from the error of hand tracking, pose offset, and the noise of the sensors. For the fast motion frames, the errors in the estimated delays did not affect the overall performance of our hand tracking approach. However, for the frames of very fast motion, the case of time delay (0.04 sec) showed better results (see Table 2).

4.1.2 Pose offset

We evaluated the pose offset estimated by solving Eq. 10. Similarly to the time-delay experimental setting, the ground-truth pose offset was created from the synthetic hand model. In our configuration, the two synthetic offsets were created as $(-0.22, -0.46, -0.21, -0.82)$ and $(-0.07, -0.96, -0.21, -0.09)$ in Eq. 2 (the ordering of the quaternion is (w,x,y,z)). Since our formulation does not consider the offset o_1, we evaluated only the offset o_2. Fig. 6(a) shows the ratio d_2/d_3 where $D = \{d_0, d_1, d_2, d_3\}$ is sorted in descending order as diagonal elements of the SVD, as solving Eq. 10, and the offset error according to the frame. We observe that the estimated pose offset was closer to the ground-truth pose when the ratio d_2/d_3 increased. Finally, the estimated offset converged to the ground-truth offset.

Time delay $[sec]$	0.00	0.01	0.02	0.03	0.04
Mean±*std on* (1)	5.3±4.1	5.4±4.0	5.3±4.1	5.2±4.2	5.3±4.6
Mean±*std on* (2)	10.2±8.8	10.2±8.7	10.5±9.3	9.5±7.4	9.2±6.6

Table 2. Mean and standard deviation (std) with a fingertip pixel error on (1) fast motion and (2) very fast motion according to time delay.

Trial	1	2	3	4	5	6	7
w	-0.74	0.74	-0.73	-0.73	-0.73	0.75	0.73
x	-0.66	0.65	-0.67	-0.67	-0.66	0.64	0.67
y	0.07	-0.07	0.05	0.06	0.07	-0.07	-0.06
z	-0.14	0.14	-0.13	-0.13	-0.15	0.14	0.13

Table 3. Consistency of the estimated pose offset (normalized quaternion) based on real data. Note that the quaternion q represents the same orientation as the one of the $-q$.

Based on the result of the synthetic dataset, we conducted the experiment of the pose offset with a real dataset. Unlike the synthetic dataset, for this type of test no ground-truth information on the offset is available. We replaced the pose-offset error as the sum of the pose errors gathered for the calibration, which is calculated as the sum of the quaternion distance between $\Delta h_c(0..t)$ and the corresponding $o_2^{-1}\Delta h_g(0..t)o_2$ in Eq. 6 for the calibration of the pose offset. Fig. 6(b) shows that the estimated pose offset converged to an orientation as the sum of the orientation errors decreased. Table 3 shows the consistency of the estimated pose offset in the real data during seven trials. The quaternion estimations were similar in all trials.

4.2 Ablation study

We quantitatively/qualitatively evaluated our method with an ablative study of its main components on the *real dataset* #1. Specifically, we compared the proposed method with the baseline HPF approach [21], with gyro-regularization, slerp-sampling, and their combination. Fig. 7 shows the effect of the components of our algorithm by plotting the percentage of frames in which the 2D position error of the tip of the little finger is below a certain threshold. First of all, adding gyro-regularization increased the accuracy of the base algorithm. Since the likelihood includes the gyro-regularization term, the likelihood of the sampled particles is affected by the calibrated gyro pose. However, the problem is that sampling relies entirely on the previous frame. Only adopting the component of gyro-regularization was not enough to achieve adequate performance for fast hand motion.

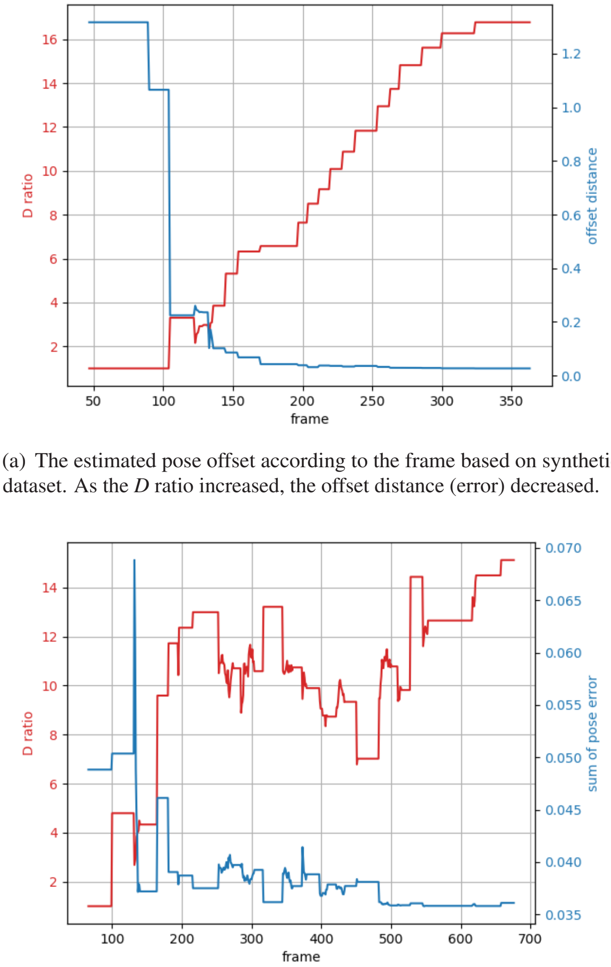
Slerp-sampling exhibited better accuracy than adopting gyro-regularization. By performing slerp-sampling, the particles are sampled among the previous orientation and the gyro trajectory. Therefore, the sampling may include the true hand pose through the trajectory of the gyro pose despite fast hand motion. However, fast hand motion generates image artifacts. Since adopting only the component of the slerp-sampling does not include gyro-regularization in the likelihood, the particles have high likelihood when fully fitting to the blurred depth.

Finally, the proposed combination of both gyro regularization and slerp sampling highly increased accuracy within range of the error [5..30] pixels.

Fig. 8 shows three consecutive frames of fast hand motion. Note that motion blur was generated by fast hand motion, as shown in the second row of Fig. 8. The base algorithm and the one with gyro-regularization failed to track the fast hand motion. Although the strategy of slerp-sampling seemed to track the fast hand motion, it tried to fit the model to blurred depth image. Consequently, the proposed exhibited adequate accuracy despite fast hand motion.

4.3 Comparison to state-of-the-art

We quantitatively/qualitatively compared the proposed method to other hand tracking methods (PSO [27], PSO with gyro [31], Tkach et al. [46], Chen et al. [4]) in both exocentric and egocentric camera view. We adapted the work [31] by replacing the CNN estimates with the calibrated gyro pose and reproduced PSO [27]. Figure 9 shows the quantitative result on *real dataset* #2, which measured the 2D pixel error of the little fingertip between the ground truth and the estimated position from each algorithm. Sample qualitative results are shown in Fig. 11.



(a) The estimated pose offset according to the frame based on synthetic dataset. As the D ratio increased, the offset distance (error) decreased.

(b) The estimated pose offset according to the frame based on real dataset. As the D ratio increased, the sum of the pose errors decreased.

Fig. 6. The result of pose-offset experiment. According to the frames, the pose-offset converged to an orientation.

Slow motion: As shown in Fig. 9, most algorithms exhibited adequate accuracy in the case of slowly moving hands where the assumption of temporal continuity was valid.

Fast motion: The performance of all algorithms decreased in the case of fast motions. In particular, the accuracy of the works [27,46] dropped considerably because they rely heavily on the estimation for the previous frame. The method [31] showed a somewhat better accuracy than the works [27,46]. Search space adaptation and gyro-regularization had a good effect when hand motion is fast. The accuracy of the fingers' pose was not stable. Especially, there were many finger tracking losses during fast hand motion. The discriminative approach of Chen et al. [4] was not much affected by fast hand motion because it estimates hand pose from learned CNN models. Although the 2D localization accuracy of the tip of the little finger was better than ours, it showed unstable result in some poses not generalized from the dataset [50] (see the first row in Fig. 11(a)).

Very fast motion: Excessively fast motions introduce considerable motion blur. Tracking methods such as the works [27,46] fail to track

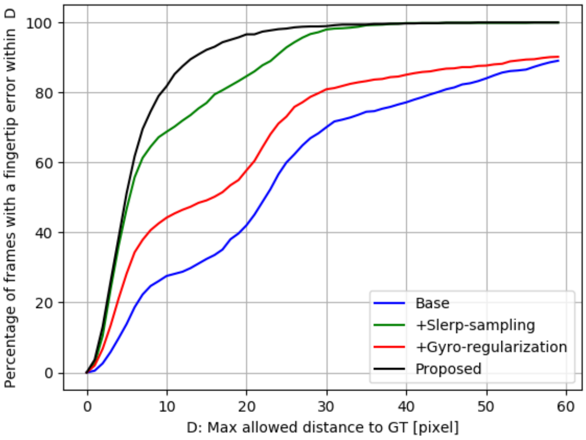


Fig. 7. Quantitative result of ablative study to evaluate the main components of our algorithm.

the hand pose. Specifically, the tracking result of Tkach et al. [46] is likely to be reinitialized well when all fingertips are visible. However, the motion blur makes the finger part highly noisy in the image and increases uncertainty. Although the gyro information is exploited by the method of Park et al. [31], its accuracy drops. Moreover, in this case, the accuracy of Chen et al. [4] was less than the proposed method because the trained dataset does not contain images contaminated by motion blur. The proposed method showed accurate and robust performance and was not much affected by very fast hand motion.

We also qualitatively compared the proposed method in another sequence obtained from an egocentric viewpoint which contains 1600 frames with fast rotation including motion blur. In this view, there are many self-occlusions, which make the problem more difficult with fast hand motion. Unlike the previous experiment, the comparison is shown qualitatively because fingers are frequently invisible in the egocentric view. Figure 11(b) shows the tracking results. In this more challenging sequence, our algorithm outperformed other methods [4, 27, 30, 46].

4.4 Possible applications

Our 3D hand tracking method can be a valuable component for supporting various applications such as 3D manipulation of virtual objects, system control, mapping to 3D avatar's hand, etc. Our method can be particularly useful in supporting hand-based interaction in scenarios where hands move fast. This allows the users to interact with virtual objects without constraints on hand motion speed. As a simple example (see Fig. 10), we show an application with the HMD-attached camera setting as in Fig. 1 where a user manipulates rapidly virtual objects in a virtual environment which was implemented in the Unity3D game engine. In order to address both camera and hand motion in the unrestricted case, we need an independent mechanism that tracks the camera relative to its environment. Although there are several methods (both visual and non-visual) for such camera tracking, we did not employ one as we considered it beyond the scope of this research. Nevertheless, our method tracks successfully the relative pose between the camera and the gyroscope-worn hand under the assumption that the camera does not move much and the hand remains within the camera's FOV. This assumption is quite realistic when a user manipulates virtual objects. More relevant results are available in the supplementary video.

Our approach can be also used to support gesture-based interaction in an AR/VR scenario in the form of sign language understanding. Motion blur occurs very often in sign language movement tracking, especially in the transition phases between different hand signs. The successful tracking of the hands in such situations may prove very beneficial towards a more successful gesture interpretation. Moreover, it can be also used to transfer correctly the motion of a user's hand into some remote place in the context of a tele-presence or tele-operation application. The investigation of such possibilities and applications is within our plans for future work.

5 Summary

We proposed a sensor-fusion method to track the articulations of a hand in the presence of excessive motion blur. To do this, we firstly calibrated the time delay and the pose offset between a depth camera and a hand-worn gyroscope. The tracking problem was formulated as a hierarchical particle filter exploiting the fusion of gyroscopic and camera-based depth information. Specifically, we proposed slerp-based sampling and gyro-regularization within the HPF framework. In the course of the extensive evaluations we performed, our method exhibited accurate and robust performance despite fast hand motions. The proposed method is the first to achieve a solution to hand articulations tracking based on sensor-fusion method in the presence of excessive motion blur.

Acknowledgments

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7066316)

References

[1] A. A. Argyros and M. I. A. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In *ECCV*, 2004.

[2] 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*, 50(2):174–188, Feb 2002.

[3] M. Boliundefined, P. M. Djuriundefined, and S. Hong. Resampling algorithms for particle filters: A computational complexity perspective. *EURASIP J. Adv. Signal Process*, 2004:2267–2277, Jan. 2004. doi: 10.1155/S1110865704405149

[4] X. Chen, G. Wang, H. Guo, and C. Zhang. Pose guided structured region ensemble network for cascaded hand pose estimation. *Neurocomputing*, 2019. doi: 10.1016/j.neucom.2018.06.097

[5] S. Cho and S. Lee. Convergence analysis of map based blur kernel estimation. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4818–4826, 2016.

[6] C. Choi, S. Kim, and K. Ramani. Learning hand articulations by hallucinating heat distribution. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3123–3132, Oct 2017. doi: 10.1109/ICCV.2017.337

[7] F. E. Daum and J. Huang. Mysterious computational complexity of particle filters. In *Signal and Data Processing of Small Targets*, vol. 4728, pp. 418 – 426. SPIE, 2002. doi: 10.1117/12.478522

[8] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. *Comput. Vis. Image Underst.*, 108(1-2):52–73, Oct. 2007. doi: 10.1016/j.cviu.2006.10.012

[9] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[10] L. Ge, Y. Cai, J. Weng, and J. Yuan. Hand pointnet: 3d hand pose estimation using point sets. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8417–8426, 2018.

[11] A. Gilbert, M. Trumble, C. Malleson, A. Hilton, and J. Collomosse. Fusing visual and inertial sensors with semantics for 3d human pose estimation. *International Journal of Computer Vision*, 127(4):381–397, Apr 2019. doi: 10.1007/s11263-018-1118-y

[12] T. Helten, M. Müller, H. Seidel, and C. Theobalt. Real-time body tracking with one depth camera and inertial sensors. In *2013 IEEE International Conference on Computer Vision*, pp. 1105–1112, Dec 2013. doi: 10.1109/ICCV.2013.141

[13] T. Hoegg, D. Lefloch, and A. Kolb. Real-time motion artifact compensation for pmd-tof images. In *Time-of-Flight and Depth Imaging*, 2013.

[14] Y. Jang, S. Noh, H. J. Chang, T.-K. Kim, and W. Woo. 3d finger cape: Clicking action and position estimation under self-occlusions in egocentric viewpoint. *IEEE Transactions on Visualization and Computer Graphics*, 21:501–510, 2015.

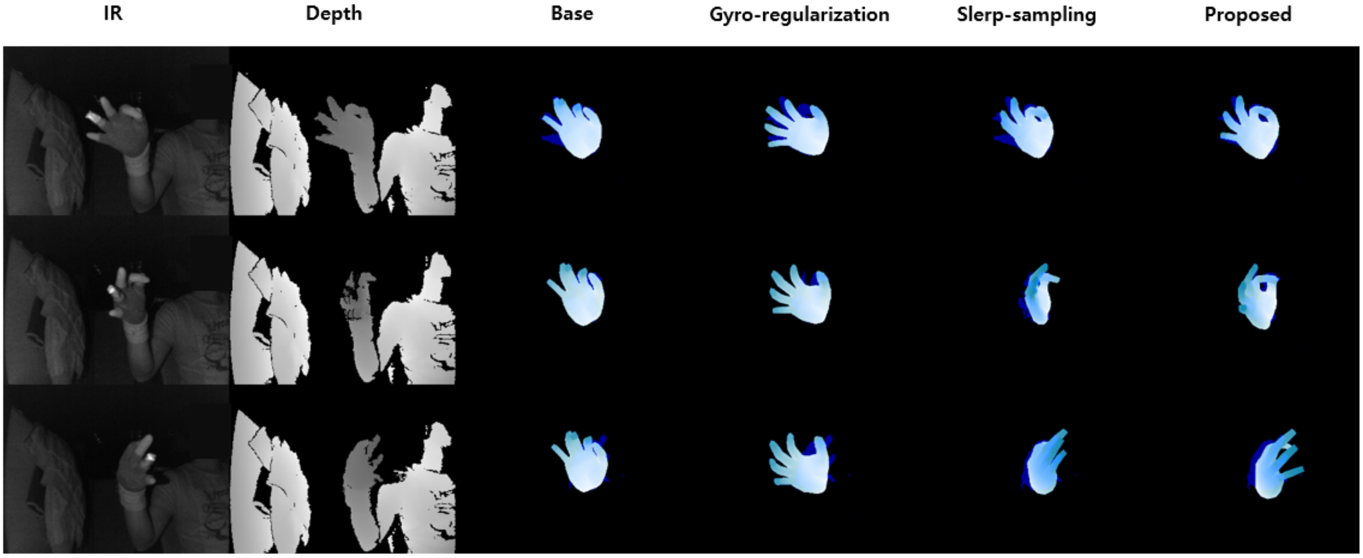


Fig. 8. Qualitative result of ablative study to evaluate the main components of our algorithm. The rows show consecutive frames during fast hand motion.

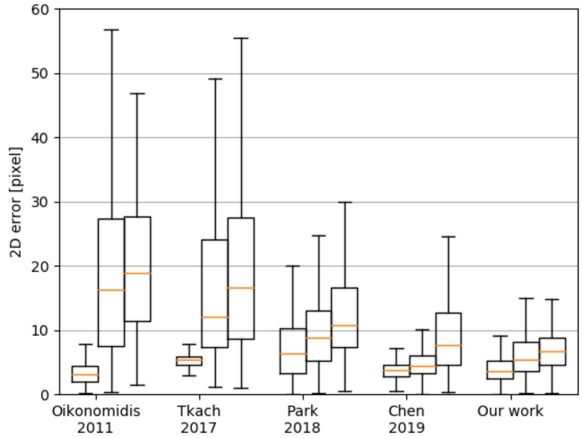


Fig. 9. Quantitative comparison based on 2D position of the little fingertip in IR image. Three cases/bars per algorithm: (left) slow motion (center) fast motion (right) very fast motion.

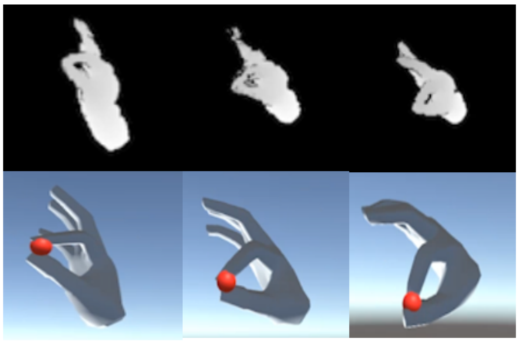


Fig. 10. Manipulation of a virtual object based on our hand tracking system. The red sphere is translated by fast hand rotation in the presence of motion blur: (first row) A blurred depth image. (second row) Interaction result corresponding to the depth image.

[15] H. Kim and W. Woo. Smartwatch-assisted robust 6-dof hand tracker for object manipulation in hmd-based augmented reality. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 251–252, March 2016. doi: 10.1109/3DUI.2016.7460065

[16] T. H. Kim, S. Nah, and K. M. Lee. Dynamic video deblurring using a locally adaptive blur model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2374–2387, Oct 2018.

[17] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[18] S. Lee. Time-of-flight depth camera motion blur detection and deblurring. *IEEE Signal Processing Letters*, 21(6):663–666, June 2014.

[19] Y. Lee, K. Wampler, G. Bernstein, J. Popović, and Z. Popović. Motion fields for interactive character locomotion. *ACM Trans. Graph.*, 29(6):138:1–138:8, Dec. 2010. doi: 10.1145/1882261.1866160

[20] A. Makris. Model-based 3D Hand Tracking with on-line Hand Shape Adaptation. *Bmvc*, pp. 1–12, 2015. doi: 10.5244/C.29.77

[21] A. Makris, N. Kyriazis, and A. A. Argyros. Hierarchical particle filtering for 3d hand tracking. *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 8–17, 2015.

[22] C. Malleson, A. Gilbert, M. Trumble, J. Collomosse, A. Hilton, and M. Volino. Real-time full-body motion capture from video and imus. In *2017 International Conference on 3D Vision (3DV)*, pp. 449–457, Oct 2017. doi: 10.1109/3DV.2017.00058

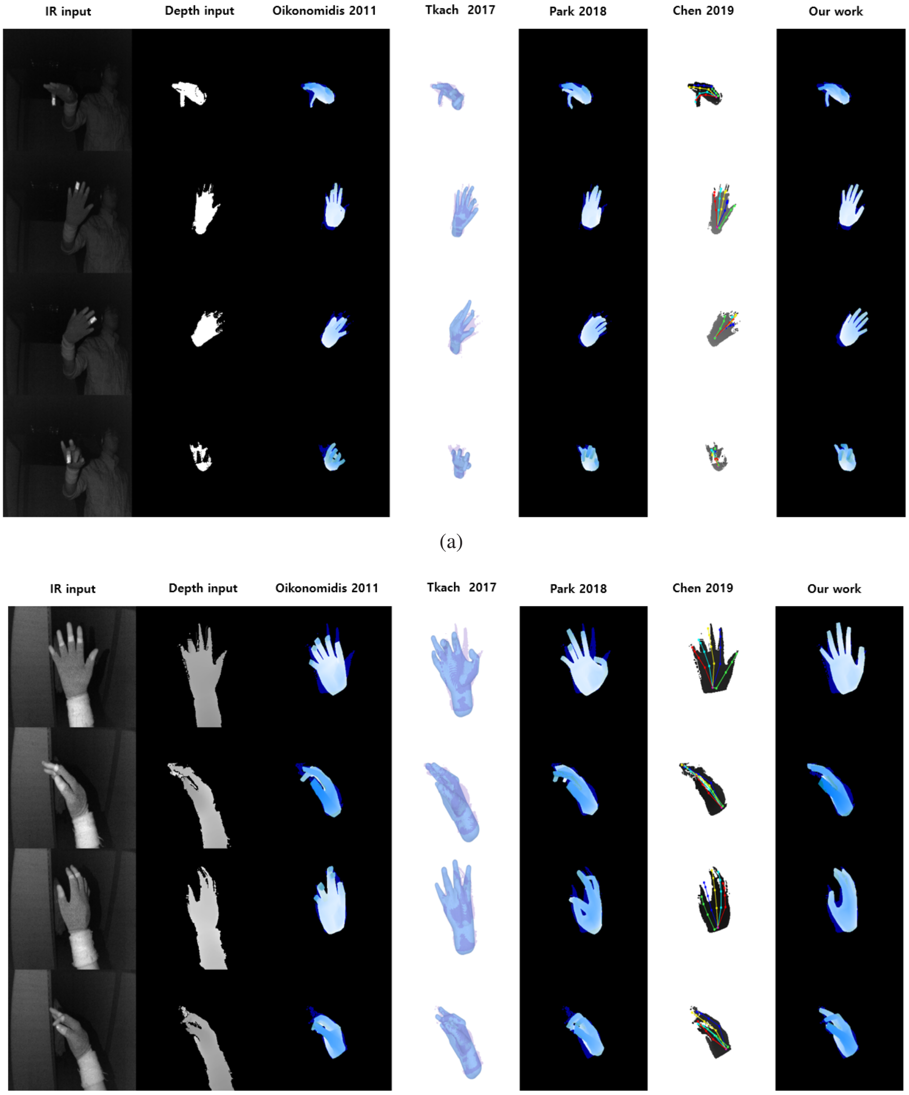
[23] S. Melax, L. Keselman, and S. Orsten. Dynamics based 3D skeletal hand tracking. *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games - I3D '13*, p. 184, 2013. doi: 10.1145/2448196.2448232

[24] G. Moon, J. Yong Chang, and K. Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[25] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[26] F. Mueller, M. Davis, F. Bernard, O. Sotnychenko, M. Verschoor, M. A. Otaduy, D. Casas, and C. Theobalt. Real-time Pose and Shape Reconstruction of Two Interacting Hands With a Single Depth Camera. *ACM Transactions on Graphics (TOG)*, 38(4), 2019.

[27] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient model-based 3D tracking of hand articulations using Kinect. *Procedings of the British*



(a)

(b)

Fig. 11. Qualitative comparison to state-of-the-art methods (a) in exocentric view and (b) egocentric view. This shows the tracking results in the presence of motion blur when the hand rotates quickly.

Machine Vision Conference 2011, pp. 101.1–101.11, 2011. doi: 10.5244/C.25.101

[28] I. Oikonomidis, M. I. A. Lourakis, and A. A. Argyros. Evolutionary quasi-random search for hand articulations tracking. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3422–3429, June 2014. doi: 10.1109/CVPR.2014.437

[29] P. Panteleris, I. Oikonomidis, and A. A. Argyros. Using a single rgb frame for real time 3d hand pose estimation in the wild. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 436–445, 2017.

[30] G. Park, A. A. Argyros, and W. Woo. Efficient 3d hand tracking in articulation subspaces for the manipulation of virtual objects. In *CGI*, 2016.

[31] G. Park and W. Woo. Hybrid 3d hand articulations tracking guided by classification and search space adaptation. *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 57–69, 2018.

[32] G. Poier, K. Roditakis, S. Schulter, M. Damien, H. Bischof, and A. Antonis. Hybrid one-shot 3d hand pose estimation by exploiting uncertainties. In *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 182.1–182.14. ., 2015. doi: 10.5244/C.29.182

[33] G. Pons-Moll, A. Baak, T. Helten, M. Müller, H. Seidel, and B. Rosenhahn. Multisensor-fusion for 3d full-body human motion capture. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 663–670, June 2010. doi: 10.1109/CVPR.2010.5540153

[34] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pp. 1106–1113. IEEE Computer Society, Washington, DC, USA, 2014. doi: 10.1109/CVPR.2014.145

[35] W. Ren, J. shan Pan, X. Cao, and M.-H. Yang. Video deblurring via semantic segmentation and pixel-wise non-linear kernel. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1086–1094, 2017.

[36] K. Roditakis, A. Makris, and A. A. Argyros. Generative 3d hand tracking with spatially constrained pose sampling. In *BMVC*, 2017.

[37] M. Schmidt and B. Jähne. Efficient and robust reduction of motion artifacts for 3d time-of-flight cameras. In *2011 International Conference on 3D Imaging (IC3D)*, pp. 1–8, Dec 2011.

[38] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, D. Freedman, P. Kohli, E. Krupka, A. Fitzgibbon, and S. Izadi. Accurate, robust, and flexible real-time hand tracking. In *Proceedings of the 33rd Annual ACM Conference on Human*

Factors in Computing Systems, CHI '15, pp. 3633–3642. ACM, New York, NY, USA, 2015. doi: 10.1145/2702123.2702179

[39] B. G. Sileshi, C. Ferrer, and J. Oliver. Particle filters and resampling techniques: Importance in computational complexity analysis. In *2013 Conference on Design and Architectures for Signal and Image Processing*, pp. 319–325, Oct 2013.

[40] X. Sun, Y. Wei, Shuang Liang, X. Tang, and J. Sun. Cascaded hand pose regression. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 824–832, June 2015. doi: 10.1109/CVPR.2015.7298683

[41] A. Tagliasacchi, M. Schröder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly. Robust articulated-icp for real-time hand tracking. *Comput. Graph. Forum*, 34:101–114, 2015.

[42] D. Tang, H. J. Chang, A. Tejani, and T. Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3786–3793, June 2014. doi: 10.1109/CVPR.2014.490

[43] D. Tang, J. Taylor, P. Kohli, C. Keskin, T. Kim, and J. Shotton. Opening the black box: Hierarchical sampling optimization for estimating human hand pose. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3325–3333, Dec 2015. doi: 10.1109/ICCV.2015.380

[44] D. Tang, T. Yu, and T. Kim. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *2013 IEEE International Conference on Computer Vision*, pp. 3224–3231, Dec 2013. doi: 10.1109/ICCV.2013.400

[45] J. Taylor, V. Tankovich, D. Tang, C. Keskin, D. Kim, P. Davidson, A. Kowdle, and S. Izadi. Articulated distance fields for ultra-fast tracking of hands interacting. *ACM Trans. Graph.*, 36(6):244:1–244:12, Nov. 2017. doi: 10.1145/3130800.3130853

[46] A. Tkach, A. Tagliasacchi, E. Remelli, M. Pauly, and A. Fitzgibbon. Online generative model personalization for hand tracking. *ACM Trans. Graph.*, 36(6):243:1–243:11, Nov. 2017. doi: 10.1145/3130800.3130830

[47] T. v. Marcard, G. Pons-Moll, and B. Rosenhahn. Human pose estimation from video and imus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1533–1547, Aug 2016. doi: 10.1109/TPAMI.2016.2522398

[48] T. Vodopivec, V. Lepetit, and P. Peer. Fine hand segmentation using convolutional neural networks. *ArXiv*, abs/1608.07454, 2016.

[49] C. Wan, T. Probst, L. V. Gool, and A. Yao. Dense 3d regression for hand pose estimation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5147–5156, 2018.

[50] S. Yuan, Q. Ye, B. Stenger, S. Jain, and T.-K. Kim. Bighand2.2m benchmark: Hand pose dataset and state of the art analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[51] Z. Zheng, T. Yu, H. Li, K. Guo, Q. Dai, L. Fang, and Y. Liu. Hybridfusion: Real-time performance capture using a single depth sensor and sparse imus. In *The European Conference on Computer Vision (ECCV)*, September 2018.