Multi-Kernel Maximum Correntropy Kalman Filter for Orientation Estimation

Shilei Li, Lijing Li, Dawei Shi, Wulin Zou, Pu Duan, and Ling Shi, *Senior Member, IEEE*

***Abstract*—Inertial measurement units (IMUs), composed of gyroscopes, accelerometers, and magnetometers, have been widely used in the fields of human motion animation, rehabilitation, robotics, and aerospace. However, their performances degenerate remarkably with external acceleration and magnetic disturbance. To handle this issue, we employ a multi-kernel maximum correntropy Kalman filter (MKMCKF) to suppress the adversarial acceleration and magnetic disturbance and use Bayesian optimization (BO) to explore the optimal kernel bandwidths. We validate our algorithm in a set of experiments with different levels of disturbance. Results show that the proposed method is significantly better than the traditional error state Kalman filter (ESKF) and the gradient descent (GD) method, and its root mean square error (RMSE) is less than 0.4629° on the roll and pitch even under the worst testing case.**

***Index Terms*—Multi-kernel correntropy, optimization and optimal control, orientation estimation, sensor fusion.**

I. Introduction

MICRO-ELECTRO-MECHANICAL-SYSTEM based IMUs have attracted continuing research efforts for their advantages of low cost, portability, and being free of restrictions. They are found in most tablet computers, smart-phones and virtual reality (VR) headsets, and have been widely used in the fields of motion capture [1], robotics [2], [3], navigation [4], localization [5], and rehabilitation [6].

IMUs are usually composed of gyroscopes, accelerometers, and magnetometers where gyroscopes measure the angular velocity rate while accelerometers and magnetometers detect the gravity vector and the magnetic vector, and are used to correct the long-term gyroscope drift. They are also referred to as attitude and heading reference system (AHRS) or magnetic angular rate and gravity system (MARG), and many algorithms have been designed to solve the orientation estimation issue. Among them, the Kalman filter-based method is a popular one and has been widely used in the motion capture and robotics [7], [8]. Other methods, such as complementary filter [9], [10], gradient descend (GD) [11], [12], and sliding mode observer [13], are also developed with different characteristics and are used for different applications. Challenging problems for IMUs are that the accelerometer is usually contaminated by external accelerations and the magnetometer is easily distorted by ferromagnetic materials, like iron and magnet. Unfortunately, the acceleration disturbance and magnetic disturbance are difficult to be modeled accurately since we do not have *a priori* knowledge about their statistical distributions. Therefore, inevitably, the estimation accuracy deteriorates remarkably. To handle this problem, Li *et al.* [14] used a Gaussian-Markov model to estimate the external acceleration; Roetenberg *et al.* [15] employed an adaptive measurement covariance to eliminate the influence of the magnetic disturbance; Madgwick *et al.* [11] re-normalized the magnetic vector to minimize its influence on inclination portions (i.e., roll and pitch); Seel and Ruppin [16] used a constraint so that the inclination was only affected by accelerometers while the heading was only affected by magnetometers. Those methods, in general, can mitigate the bad effects of external accelerations and magnetic disturbance to some extent. However, their performances are usually unsatisfactory since the estimation accuracy without disturbance is sacrificed.

Maximum correntropy criterion (MCC) provides a potential solution for the aforementioned issue. Correntropy is an information-theoretic measure of the similarity between two arbitrary random variables [17], [18]. It is robust to heavy-tailed noises and has been widely used in regression [19], machine learning [17], Kalman filter (KF) [20] or its variants [21]–[23], and adaptive filtering [24]. Aravkin *et al.* [25] showed that a mixed-norm penalty function (combined ℓ_1 norm and ℓ_2 norm) can effectively suppress the disturbance. Moreover, Liu *et al.* [18] reported that the correntropy induced metric (CIM) divided space into three regions: when two samples were close, the CIM was similar to an ℓ_2 norm; when they went further apart, the CIM was approaching an ℓ_1 norm; when the distance between the two samples was very large, the CIM was approximated by an ℓ_0 norm. The scale of the norm can be controlled by the kernel bandwidth which gives great flexibility for the algorithm design when considering different types of noises (i.e., the Gaussian

Manuscript received December 21, 2021; accepted May 5, 2022. Date of publication May 23, 2022; date of current version May 31, 2022. This letter was recommended for publication by Associate Editor B. Noack and Editor E. Marchand upon evaluation of the reviewers' comments. The work by Shilei Li, Wulin Zou and Ling Shi was supported by the Hong Kong ITF Fund under Grant GHP/001/18SZ and the work of Dawei Shi was supported by the Natural Science Foundation of China under Grant 61973030. *(Corresponding author: Shilei Li.)*

Shilei Li and Wulin Zou are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong 999077, and also with the Control Department, Xeno Dynamics, Co., Ltd, Shenzhen 518055, China (e-mail: slidk@connect.ust.hk; wzouab@connect.ust.hk).

Lijing Li is with the School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China (e-mail: lilijing_29@163.com).

Dawei Shi is with the School of Automation, Beijing Institute of Technology, Haidian 10081, China (e-mail: daweishi@bit.edu.cn).

Pu Duan is with the Control Department, Xeno Dynamics, Co., Ltd, Shenzhen 518055, China (e-mail: duanpu@xeno.com).

Ling Shi is with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong 999077 (e-mail: eesling@ust.hk).

Digital Object Identifier 10.1109/LRA.2022.3176798

noises and the heavy-tailed noises). Using this property, in our previous work, Li *et al.* [26] designed a multi-kernel maximum correntropy Kalman filter (MKMCKF) which utilized different bandwidths for different channels so that the CIM can be adjusted among an ℓ_2 norm, ℓ_1 norm, and ℓ_0 norm flexibly in accordance with the instantaneous error characteristics. Specifically, when the error is relatively small, the corresponding CIM behaves like an ℓ_2 norm. In this case, the MKMCKF is nearly identical to the KF. On the contrary, when the system involves outliers or non-Gaussian noises, the error is expected to be bigger and the CIM behaves like an ℓ_1 norm or ℓ_0 norm. In this situation, the MKCMKF is very robust to outliers or disturbance. Therefore, this algorithm rejects disturbance effectively with the existence of disturbance and performs well without disturbance.

A remaining issue for the MKMCKF is the kernel bandwidths selection. Generally, we aim to choose a set of kernel bandwidths so that the MKMCKF is similar to the traditional KF with an ℓ_2 norm-based objective function when without disturbance and is similar to an ℓ_0 or ℓ_1 norm-based objective function with disturbance. However, it is difficult to tune the bandwidths manually, especially when the system's dimension is high or with constraints. To handle this issue, we employ the Bayesian optimization (BO) for kernel bandwidths selection which is an efficient tool for global minimization where the objective function can be black-box, non-convex, and expensive to be evaluated [27]–[29]. We first derive a novel algorithm named multi-kernel maximum correntropy Kalman filter for orientation estimation (MKMCKF-OE), then we explore the optimal bandwidths using BO on the training set. Finally, we validate the performance of the proposed algorithm on the testing set and compare it with the gradient descent (GD) [11], traditional error state Kalman filter (ESKF) [15], [30], improved gradient descent (IGD) [16], and versatile quaternion-based filter (VQF) [31]. The construction of the MKMCKF-OE is built upon our previous work [26]. Contributions of this letter are summarized as follows:

1) A novel MKMCKF-OE is designed for IMUs, which performs well both with and without disturbance.
2) The BO is utilized to find the optimal kernel bandwidths, which is highly efficient and convenient.
3) The performance of the proposed algorithm is validated on the testing set and is shown to be significantly better than the GD and ESKF, and is superior to the IGD and VQF on the yaw axis with magnetic disturbance.

The remainder of this letter is organized as follows. In Section II, we give some preliminaries. In Section III, we derive a novel algorithm MKMCKF-OE for IMUs. In Section IV, we explore the optimal kernel bandwidths by BO. In Section V, we verify the effectiveness of the proposed methods in experiments. In section VI, we draw a conclusion.

Notations: The transpose of a matrix A is denoted by A^T. The *a priori* estimate and the *a posteriori* estimate of state x is denoted by x^- and x^+, respectively. The symbol $\times$ is the vector product operation and $\mathrm{v}\times$ converts a vector v to a skew symmetric matrix. A vector with l dimensions is denoted by $\mathbb{R}^l$ and a matrix with m rows and n columns is denoted by $\mathbb{R}^{m\times n}$. The Gaussian distribution with mean μ and covariance Σ is denoted by $\mathcal{N}(\mu,\Sigma)$.

II. Preliminaries

In this section, we give a brief introduction about sensor models of IMUs, multi-kernel maximum correntropy Kalman filter (MKMCKF), and Bayesian optimization (BO).

A. Sensor Models

We give sensor models of gyroscopes, accelerometers, and magnetometers. Firstly, the gyroscope signal $y_{G,k}\in\mathbb{R}^3$ at time step k is composed of the angular velocity w_k, the offset b_k, and the white noise $v_{G,k}$. The offset is modeled as a Markov process with an additional Gaussian noise $v_{b,k}$. Then, we have

$$\begin{aligned} y_{G,k} &= w_k + b_k + v_{G,k} \\ b_k &= b_{k-1} + v_{b,k} \end{aligned} \tag{1}$$

where $v_{G,k}\sim\mathcal{N}(0,Q_{vG})$ and $v_{b,k}\sim\mathcal{N}(0,Q_{vb})$. The accelerometer signal $y_{A,k}\in\mathbb{R}^3$ is composed of the gravity vector in the sensor frame ${}^S g_{A,k}$, the external acceleration ${}^S a_k$ in the sensor frame, and an additional noise $v_{A,k}$. The external acceleration in the global frame ${}^G a_k$ is modeled as a decayed Markov process with coefficient η_a and the noise $v_{a,k}$. Thus, we obtain

$$\begin{aligned} y_{A,k} &= -{}^S g_{A,k} + {}^S a_k + v_{A,k} \\ {}^G a_k &= \eta_a({}^G a_{k-1}) + v_{a,k} \end{aligned} \tag{2}$$

where $0<\eta_a<1, v_{A,k}\sim\mathcal{N}(0,Q_{vA}), v_{a,k}\sim\mathcal{N}(0,Q_{va})$. The free acceleration in the global frame and the sensor frame can be transferred by a rotation matrix ${}^G_S R$ with

$${}^G a_k = {}^G_S R({}^S a_k). \tag{3}$$

Finally, the magnetometer signal $y_{M,k}\in\mathbb{R}^3$ is a summation of the magnetic vector ${}^S m_k$, the magnetic disturbance ${}^S d_k$ and the noise $v_{M,k}$. The magnetic disturbance in the global frame ${}^G d_k$ is modeled as a Markov process with a coefficient η_m and an additional noise $v_{d,k}$. Hence, one has

$$\begin{aligned} y_{M,k} &= {}^S m_k + {}^S d_k + v_{M,k} \\ {}^G d_k &= \eta_m({}^G d_{k-1}) + v_{d,k} \end{aligned} \tag{4}$$

where $0<\eta_m<1,\ v_{M,k}\sim\mathcal{N}(0,Q_{vM}),\ v_{d,k}\sim\mathcal{N}(0,Q_{vd})$. Similarly, the disturbance in the sensor frame and the global frame can be transferred by

$${}^G d_k = {}^G_S R({}^S d_k). \tag{5}$$

One can refer to [15], [30] for more information.

B. Multi-kernel Maximum Correntropy Kalman Filter

We consider the linear model with

$$\begin{aligned} \mathrm{x}_{k+1} &= \mathrm{A}\mathrm{x}_k + \mathrm{p}_k \\ \mathrm{y}_k &= \mathrm{C}\mathrm{x}_k + \mathrm{q}_k \end{aligned} \tag{6}$$

where $\mathrm{x}_k\in\mathbb{R}^n$, $\mathrm{y}_k\in\mathbb{R}^m$, A and C is state transition matrix and observation matrix, respectively. The noise $\mathrm{p}_k\sim\mathcal{N}(0,\mathrm{Q})$ and $\mathrm{q}_k\sim\mathcal{N}(0,\mathrm{R})$. (6) can be rewritten as

$$\begin{bmatrix} \mathrm{x}_k^- \\ \mathrm{y}_k \end{bmatrix} = \begin{bmatrix} \mathrm{I} \\ \mathrm{C} \end{bmatrix} \mathrm{x}_k + \zeta_k \tag{7}$$

with

$$\zeta_k = \begin{pmatrix} \mathrm{x}_k^- - \mathrm{x}_k \\ \mathrm{q}_k \end{pmatrix}$$

where x_k^- denotes the *a priori* estimate of x_k. The covariance of ζ_k has

$$E(\zeta_k \zeta_k^T) = \begin{pmatrix} \mathrm{P}_k^- & 0 \\ 0 & \mathrm{R} \end{pmatrix} = \begin{pmatrix} \mathrm{T}_p \mathrm{T}_p^T & 0 \\ 0 & \mathrm{T}_r \mathrm{T}_r^T \end{pmatrix} = \mathrm{T}_k \mathrm{T}_k^T \tag{8}$$

where P_k^- is the *a priori* error covariance at time step k, R is the measurement covariance, and T_p, T_r can be obtained by Cholesky decomposition. By left multiplying T_k^{-1} in both sides of (7), we obtain

$$\mathrm{D}_k = \mathrm{W}_k \mathrm{x}_k + \nu_k \tag{9}$$

where

$$\mathrm{D}_k = \mathrm{T}_k^{-1} \begin{pmatrix} \mathrm{x}_k^- \\ \mathrm{y}_k \end{pmatrix}, \mathrm{W}_k \quad = \mathrm{T}_k^{-1} \begin{pmatrix} \mathrm{I} \\ \mathrm{C} \end{pmatrix}, \nu_k = \mathrm{T}_k^{-1} \zeta_k \tag{10}$$

The noise term ν_k is white since $\mathrm{E}(\nu_k \nu_k^T) = \mathrm{E}[(\mathrm{T}_k^{-1}\zeta_k)(\mathrm{T}_k^{-1}\zeta_k)^T] = \mathrm{I}$. The objective of MKMCKF is to solve the following problem:

$$\begin{aligned} \mathrm{x}_k^+ &= \arg\max_{\mathrm{x}_k \in \mathscr{X}} J(\mathrm{x}_k) \\ &= \arg\max_{\mathrm{x}_k \in \mathscr{X}} \sum_{i=1}^{n+m} 2\sigma_i^2 G_{\sigma_i}(\mathrm{e}_{i,k}). \end{aligned} \tag{11}$$

with

$$G_{\sigma_i}(\mathrm{e}_{i,k}) = \exp\left(-\frac{\mathrm{e}_{i,k}^2}{2\sigma_i^2}\right) \tag{12}$$

where $\mathscr{X}$ is the domain of x_k, $\mathrm{e}_{i,k} = \mathrm{d}_{i,k} - \mathrm{w}_{i,k}\mathrm{x}_k$ denotes the error at time step k in the i-th channel, $\mathrm{d}_{i,k}$ is the i-th element of D_k, and $\mathrm{w}_{i,k}$ is the i-th row of W_k. (11) can be equivalently written in a CIM-based form with

$$\mathrm{x}_k^+ = \arg\min_{\mathrm{x}_k \in \mathscr{X}} \sum_{i=1}^{n+m} \left(\mathrm{CIM}(\mathrm{e}_{i,k})\right)^2. \tag{13}$$

where

$$\mathrm{CIM}(\mathrm{e}_{i,k}) = \left(2\sigma_i^2 - 2\sigma_i^2 G_{\sigma_i}(\mathrm{e}_{i,k})\right)^{1/2}.$$

Then, the norm of i-th channel can be adjusted among the ℓ_2, ℓ_1 and ℓ_0 norm based on the bandwidth σ_i. The corresponding MKMCKF is summarized in Appendix A1, and the details can be found in our previous work [26].

C. Bayesian Optimization

We use the BO to explore the optimal kernel bandwidths for the MKMCKF. Suppose we have a function $f : \mathscr{Z} \subset \mathbb{R}^l \to \mathbb{R}$ with $x \in \mathscr{Z}$. We try to find the minima point x^* on a domain $\mathscr{Z}$, i.e.,

$$x^* = \arg\min_{x \in \mathscr{Z}} f(x). \tag{14}$$

In the framework of BO, the distribution f is assumed to be Gaussian with $p(f) = \mathcal{GP}(f; \mu, K)$ where μ is the mean and K is the covariance. With N finite samples $\boldsymbol{x}_{1:N}$, the Gaussian Process (GP) model is a joint Gaussian $\mathbf{f}(\boldsymbol{x}_{1:N}) \sim \mathcal{N}(\mu(\boldsymbol{x}_{1:N}), \mathbf{K})$ where $\mathbf{K}_{i,j} = \kappa(x_i, x_j)$ and $\kappa(\cdot,\cdot)$ is the covariance function. For a new data x_{N+1}, we consider the joint distribution over f with both the old data and new data:

$$\begin{pmatrix} \mathbf{f}(\boldsymbol{x}_{1:N}) \\ f(x_{N+1}) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}(\boldsymbol{x}_{1:N}) \\ \mu(x_{N+1}) \end{pmatrix}, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^T & \kappa(x_{N+1}, x_{N+1}) \end{bmatrix}\right) \tag{15}$$

where $\mathbf{k} = \kappa(\boldsymbol{x}_{1:N}, x_{N+1})$. We denote the training set as $\mathcal{D}_N := \{\boldsymbol{x}_{1:N}, \mathbf{f}(\boldsymbol{x}_{1:N})\}$. Then, using the standard conditioning rules for Gaussian random variables, we have

$$f(x_{N+1}) \mid \mathcal{D}_N, x_{N+1} \sim \mathcal{N}\left(\hat{\mu}(x_{N+1}), \hat{\Sigma}^2(x_{N+1})\right) \tag{16}$$

with

$$\begin{aligned} \hat{\mu}(x_{N+1}) &= \mu(x_{N+1}) + \mathbf{k}^T \mathbf{K}^{-1}(\mathbf{f}(\boldsymbol{x}_{1:N}) - \boldsymbol{\mu}(\boldsymbol{x}_{1:N})) \\ \hat{\Sigma}^2(x_{N+1}) &= \kappa(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) - \mathbf{k}^T \mathbf{K}^{-1} \mathbf{k}. \end{aligned} \tag{17}$$

To use a GP, one has to specify the mean function and the covariance function. The mean function is usually set to be zero and the covariance function can be set as Gaussian kernel with $\kappa(x, x') = \exp(\frac{1}{2}(x - x')^T M^{-1}(x - x'))$. The kernel parameters M can be obtained by empirical Bayesian methods [32]; see [33], [34] for more information.

Expected improvement (EI) is a common acquisition function to determine which point should be evaluated at the next iteration. Suppose that f^* is the minimum value of f observed so far. Essentially, EI tries to find a candidate point x that improves upon f^* the most with a utility function

$$u(x) = \max_{x \in \mathscr{Z}}(0, f^* - f(x)). \tag{18}$$

The acquisition function can be written as

$$a_{EI}(x) = \hat{\Sigma}(x)\left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)}\Phi\left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)}\right) + \phi\left(\frac{f^* - \hat{\mu}(x)}{\hat{\Sigma}(x)}\right)\right) \tag{19}$$

where Φ is the cumulative distribution function and ϕ is the standard normal probability density function; see [27], [35] for details. Naturally, $a_{EI}(x)$ balances between the exploitation and the exploration. The next candidate point x_{nc} can be selected as

$$x_{nc} = \arg\max_{x \in \mathscr{Z}} a_{EI}(x). \tag{20}$$

In the applications with the inequity constraint $c(x) \leq \lambda$, we adopt the method in [27] which writes the constrained utility function as

$$u_c(x) = \Delta(x) \max_{x \in \mathscr{Z}}(0, f^* - f(x)) = \Delta(x) u(x). \tag{21}$$

where $\Delta(x) = \{0, 1\}$ is a feasibility indicator and $\Delta(x) = 1$ if $c(x) \leq \lambda$, and $\Delta(x) = 0$ otherwise. Then, the constrained acquisition function can be written as

$$a_{CEI}(x) = Pr(c(x) \leq \lambda) \cdot a_{EI}(x) \tag{22}$$

with

$$Pr(c(x) \leq \lambda) = \int_{-\infty}^{\lambda} p(c(x) \mid x, \mathcal{D}_N) \mathrm{d}c(x) \tag{23}$$

The next candidate point can be obtained by

$$x_{nc} = \arg\max_{\sigma \in \mathscr{Z}} a_{CEI}(x). \tag{24}$$

III. Proposed Algorithm

In this section, we derive a novel MKMCKF-OE algorithm using the maximum correntropy criterion. Then, we tune the hyper-parameters (i.e., kernel bandwidths) with BO to minimize the orientation error.

A. MKMCKF-OE

We use the error state rather than the nominal state to construct the *error model:*

$$x_{\varepsilon,k+1} = \Phi_k x_{\varepsilon,k} + w_k \tag{25}$$

with

$$x_{\varepsilon,k} = \begin{pmatrix} \theta_{\varepsilon,k} \\ b_{\varepsilon,k} \\ {}^S a_{\varepsilon,k} \\ {}^S d_{\varepsilon,k} \end{pmatrix}, w_k = \begin{pmatrix} w_{\theta,k} \\ w_{b,k} \\ w_{a,k} \\ w_{d,k} \end{pmatrix} \tag{26}$$

where the subscript ε denotes the difference of the estimated state and the nominal state, the error state $x_{\varepsilon,k}$ includes the error orientation $\theta_{\varepsilon,k} \in \mathbb{R}^3$ (Euler angle), the error gyroscope offset $b_{\varepsilon,k} \in \mathbb{R}^3$, the error external acceleration in the sensor frame ${}^S a_{\varepsilon,k} \in \mathbb{R}^3$, and the error magnetic disturbance in the sensor frame ${}^S d_{\varepsilon,k} \in \mathbb{R}^3$. The noise w_k is assumed to be Gaussian with covariance Q. Notably, the error state transition matrix $\Phi_k \in \mathbb{R}^{12\times 12}$ is a zero matrix since the *a priori* estimate of $x^-_{\varepsilon,k}$ does not depend on the previous *a posteriori* estimate of $x^+_{\varepsilon,k-1}$ (the knowledge about previous errors has been incorporated in the current nominal state.). A more detailed description can be found in [15]; one can also refer to [30] for a full treatment of this model.

The measurement signals are related with the error state through

$$\begin{aligned} z_{\varepsilon,k} &= \begin{pmatrix} {}^S\hat{g}^-_{A,k} - {}^S\hat{g}^-_{G,k} \\ {}^S\hat{m}^-_{M,k} - {}^S\hat{m}^-_{G,k} \end{pmatrix} + v_k \\ &= H_k x_{\varepsilon,k} + v_k \end{aligned} \tag{27}$$

with

$$\begin{aligned} H_k &= \begin{bmatrix} -{}^S\widehat{g^-_{G,k}}\times, \delta t({}^S\widehat{g^-_{G,k}}\times), I_{3\times3}, 0_{3\times3} \\ -{}^S\widehat{m^-_{G,k}}\times, \delta t({}^S\widehat{m^-_{G,k}}\times), 0_{3\times3}, -I_{3\times3} \end{bmatrix} \\ {}^S g^-_{A,k} &= -y_{A,k} + {}^S\hat{a}^-_k,\ {}^S g^-_{G,k} = \hat{R}^-_k({}^G g_k) \\ {}^S m^-_{M,k} &= y_{M,k} - {}^S\hat{d}^-_k,\ {}^S m^-_{G,k} = \hat{R}^-_k({}^G m_k) \end{aligned} \tag{28}$$

where ${}^S\hat{g}^-_{A,k}$ and ${}^S\hat{g}^-_{G,k}$ denotes the *a priori* estimate of gravity vector obtained by the accelerometer model and by the coordinate transformation, ${}^S\hat{m}^-_{M,k}$ and ${}^S m^-_{G,k}$ denotes the *a priori* estimate of geomagnetic vector obtained by the magnetometer model and by the coordinate transformation, ${}^S\hat{a}^-_k$ and ${}^S\hat{d}^-_k$ is the *a priori* estimate of external acceleration and the *a priori* estimate of magnetic disturbance, ${}^G g_k$ and ${}^G m_k$ is the constant gravity vector and the geomagnetic vector, $\hat{R}^-_k$ is the *a priori* estimate of orientation matrix. The measurement noise is v_k with covariance Z_k, δt is the sampling rate, and P_k is the corresponding error covariance matrix. One can refer to [15], [30] for more information.

Algorithm 1: MKMCKF-OE.

Initialization:
Choose kernel bandwidth vector $\sigma_p \in \mathbb{R}^{12}$, $\sigma_r \in \mathbb{R}^6$.
State Prediction:
$\hat{b}^-_k = \hat{b}^+_{k-1}$
${}^S\hat{a}^-_k = \eta_a({}^S\hat{a}^+_{k-1})$ ▷derived from (2)
${}^S\hat{d}^-_k = \eta_m({}^S\hat{d}^+_{k-1})$ ▷derived from (4)
$\hat{\omega}^-_k = y_{G,k} - \hat{b}^-_k$
$\hat{q}^-_k = \hat{q}^+_{k-1}\Delta q(\hat{\omega}^-_k \delta t)$
Error State Propagation:
$P^-_k = f(P^+_{k-1}, Q)$, details in A2
Obtain T_p and T_r by $T_p T_p^T = P^-_k$ and $T_r T_r^T = Z_k$
Initialize $x^-_{\varepsilon,k} = x^+_{\varepsilon,k,0} = 0$
while $\frac{x^+_{\varepsilon,k,t} - x^+_{\varepsilon,k,t-1}}{x^+_{\varepsilon,k,t-1}} \le \varepsilon$ or $t = 1$ **do** ▷t starts from 1
$x^+_{\varepsilon,k,t} = x^-_{\varepsilon,k} + \tilde{K}_k z_{\varepsilon,k}$ ▷$z_{\varepsilon,k}$ obtained by (27)
$\tilde{K}_k = \tilde{P}^-_k H_k^T S_k^{-1}$ ▷H_k is shown in (28)
$S_k = H_k \tilde{P}^-_k H_k^T + \tilde{Z}_k$
$\tilde{P}^-_k = T_p \tilde{M}_p^{-1} T_p^T$
$\tilde{Z}^-_k = T_r \tilde{M}_r^{-1} T_r^T$
$\tilde{M}_p = diag(G_{\sigma_p}(e_p))$
$\tilde{M}_r = diag(G_{\sigma_r}(e_r))$
$e_p = T_p^{-1} x^-_{\varepsilon,k} - T_p^{-1} x^+_{\varepsilon,k,t-1}$
$e_r = T_r^{-1}(z_{\varepsilon,k} - H_k x^+_{\varepsilon,k,t-1})$
end while
State Update:
Obtain $\hat{\theta}^+_{\varepsilon,k}$, ${}^S\hat{a}^+_{\varepsilon,k}$ $\hat{b}^+_{\varepsilon,k}$, ${}^S\hat{d}^+_{\varepsilon,k}$ from $x^+_{\varepsilon,k,t}$ ▷based on (26)
$\hat{q}^+_k = \hat{q}^-_k \Delta q(-\hat{\theta}^+_{\varepsilon,k})$
$\hat{b}^+_k = \hat{b}^-_k - \hat{b}^+_{\varepsilon,k}$
${}^S\hat{a}^+_k = {}^S\hat{a}^-_k - {}^S\hat{a}^+_{\varepsilon,k}$
${}^S\hat{d}^+_k = {}^S\hat{d}^-_k - {}^S\hat{d}^+_{\varepsilon,k}$ ▷if $M_J = 0$ in (33)
update ${}^G m_k$ using (34), (35) ▷if $M_J = 0$ in (33)
$P^+_k = (I - \tilde{K}_k H_k) P^-_k (I - \tilde{K}_k H_k)^T + \tilde{K}_k Z_k \tilde{K}_k^T$

The nominal state can be written as $x_k = [\theta_k^T, b_k^T, ({}^S a_k)^T, ({}^S d_k)^T]^T$ which includes the orientation (Euler angle), the gyroscope offset, the external acceleration, and the magnetic disturbance. The orientation $\theta_k \in \mathbb{R}^3$ can be written as a rotation matrix $R_k \in \mathbb{R}^{3\times3}$ or a quaternion $q_k \in \mathbb{R}^4$. Obviously, the modeling of the external acceleration and the magnetic disturbance in (2) and (4) is not accurate. The unmodeled part can be seen as non-Gaussian noises which deteriorate the estimate accuracy significantly. To handle this issue, we employ MKMCKF for orientation estimation and construct a novel algorithm MKMCKF-OE. The difference between our algorithm and the ESKF [15], [30] is that we employ Gaussian kernels to increase its robustness to disturbance. The detailed algorithm is summarized in Algorithm 1.

A classifier is utilized to detect magnetic jamming. Moreover, a geomagnetic vector reconstruction strategy is adopted to decrease the influence of the magnetometer readings on the pitch and the roll. The details are shown in Appendix A3.

B. Kernel Bandwidths Optimization

In Algorithm 1, we have to choose kernel bandwidth vectors σ_p and σ_r. Based on Theorem 1 in [26], the multi-kernel maximum correntropy Kalman filter is identical to the traditional KF if we set all kernel bandwidths $\sigma_i \to \infty$. Since KF is optimal for a linear system with Gaussian noises, we can use a unified big kernel bandwidth, i.e., $\sigma_{\inf}$, for Gaussian channels. As for non-Gaussian channels, we tune them by BO. In IMUs, the non-Gaussian noises exit in the external acceleration and magnetic disturbance channels due to the first-order dynamic assumption in equations (2) and (4), while noises in other channels can be regarded as Gaussian. Therefore, we have

$$\begin{aligned}\sigma_p &= (\sigma_{\inf} I_{1\times 3}, \sigma_{\inf} I_{1\times 3}, \sigma_1 I_{1\times 3}, \sigma_2 I_{1\times 3})^T \\ \sigma_r &= (\sigma_{\inf} I_{1\times 3}, \sigma_{\inf} I_{1\times 3})^T \end{aligned} \tag{29}$$

where $I_{1\times 3}$ is the identity vector, $\sigma_{\inf}$ is a big constant ($\sigma_{\inf} = 10^8$ in experiments), and σ_1 and σ_2 are two parameters to be optimized. The objective function of BO is to minimize the RMSE on all axes with a constraint on the yaw axis:

$$\begin{aligned} &\arg\min_{\sigma_1,\sigma_2\in\mathscr{Z}} J_{BO} \\ &= \arg\min_{\sigma_1,\sigma_2\in\mathscr{Z}} \frac{1}{3m}\sum_{i=1}^{3}\sum_{j=1}^{m}\underbrace{\frac{1}{n}\sqrt{\sum_{k=1}^{n}(\hat{\theta}_{i,j,k}(\sigma_1,\sigma_2)-\theta_{i,j,k})^2}}_{RMSE} \\ &s.t. \frac{1}{m}\sum_{j=1}^{m}\frac{1}{n}\sqrt{\sum_{k=1}^{n}(\hat{\theta}_{3,j,k}(\sigma_1,\sigma_2)-\theta_{3,j,k})^2} \le \lambda \end{aligned} \tag{30}$$

where m is the number of training sets, $\mathscr{Z}$ is the domain of kernel bandwidths, $\hat{\theta}_{i,j,k}(\sigma_1,\sigma_2)$ represents the estimated angle under i axis (roll, pitch and yaw) at j-th experiment and time step k, $\theta_{i,j,k}$ represents the corresponding ground truth angle (Euler angle). The constant λ gives an upper RMSE bound on the yaw axis ($i = 3$). In total, we conduct m experiments to involve different situations, i.e., with or without acceleration disturbance, with or without magnetic disturbance, and different types of movement (linear motion or rotation).

IV. Experiments

In this section, we first obtain the optimal kernel bandwidths through BO. Then, we validate the performance of the MKMCKF-OE in the testing set and compare it with the GD [11], ESKF [15], [30], IGD [16], and VQF [31]. The implementations of the GD, ESKF, and MKMCKF-OE are available at [36].

A. Data Collection

We use Xsens MTI-670 to collect the raw data. The sampling frequency for accelerometers and gyroscopes is 400 Hz while the value for magnetometers is 100 Hz. The MKMCKF-OE is executed at 100 Hz. The experimental setup is shown in Fig. 1(a) and Fig. 1(b) which corresponds to the linear motion and the rotation, respectively. In Fig. 1(a), the acceleration disturbance A_d is generated by moving the Xsens along the rail manually, while the magnetic disturbance M_d is generated by the approaching

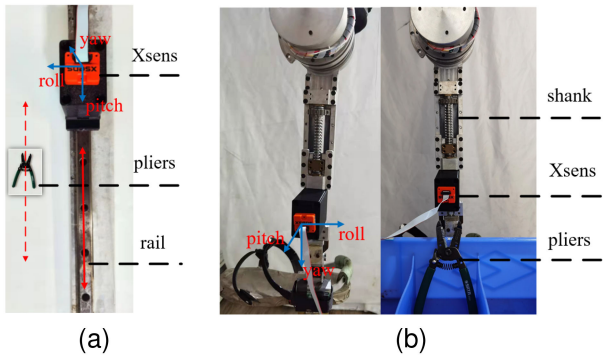

Fig. 1. The experimental setup and the IMU coordinate system. The movement can be divided into linear motion in Fig. 1(a) and rotation in Fig. 1(b). The acceleration disturbance A_d is generated by the movement of Xsens itself and the magnetic disturbance M_d is generated by the pliers.

TABLE I
Experimental Description

Index	Mode	Description	Usage
1	linear motion	A_d,M_d	training
2	rotation	$f = 0.1Hz$	training
3	rotation	$f = 0.5Hz$	training
4	rotation	$f = 0.1Hz$, M_d	training
5	rotation	$f = 0.5Hz$, M_d	training
6	linear motion	A_d,M_d	testing
7	rotation	$f = 0.2Hz$	testing
8	rotation	$f = 0.3Hz$	testing
9	rotation	$f = 0.4Hz$	testing
10	rotation	$f = 0.2Hz$, M_d	testing
11	rotation	$f = 0.3Hz$, M_d	testing
12	rotation	$f = 0.4Hz$, M_d	testing

of the ferromagnetic pliers. In Fig. 1(b), the Xsens is fixed to the shank of an exoskeleton robot where its leg swings to imitate the gait of a human. The gait frequency is denoted by f and the acceleration disturbance grows with f. On the right side of Fig. 1(b), the magnetic disturbance M_d is added when the pliers are close to the Xsens.

The ground truth orientation for the linear motion is set to be the initial mean angle when the Xsens is static without disturbance. The ground truth pitch angle for rotation is obtained by encoders of the exoskeleton robot, while the roll angle and the yaw angle are obtained by the initial mean angles of the corresponding axes when the Xsens is static since there is no rotation at those axes. In total, we conduct 12 experiments which contain two experiments for linear motion and 10 experiments for rotation. A detailed description of the experiments is summarized in Table I.

B. Kernel Bandwidths Optimization

A total of 5 experiments are involved in the training set which are experiments 1–5 in Table I. Among them, experiment 5 has the fastest dynamics with both obvious acceleration disturbance and magnetic disturbance. Since the magnetic disturbance affects the yaw angle the most, our objective is to minimize the

TABLE II
RMSE OF DIFFERENT ALGORITHMS ON THE ROLL AND PITCH

Index	Roll (deg)					Pitch (deg)				
	MKMC	GD	ESKF	IGD	VQF	MKMC	GD	ESKF	IGD	VQF
6	0.3683	1.7992	1.4010	0.7596	0.8669	0.4257	0.4549	0.9942	0.2891	0.5410
7	0.1381	0.4985	0.5447	0.1104	0.0996	0.2833	1.3211	0.8961	0.5396	0.4900
8	0.2199	0.4077	0.5463	0.1867	0.2204	0.3811	1.4246	0.9958	0.6671	0.3849
9	0.1827	0.5400	0.4274	0.4394	0.4002	0.4629	1.7052	1.3178	1.4895	0.5611
10	0.1133	1.5521	2.1813	0.0744	0.1039	0.4228	1.9076	3.3010	0.4441	0.2543
11	0.2093	1.2445	1.9321	0.1305	0.1467	0.3022	1.6302	2.9933	0.4670	0.3114
12	0.1759	1.2293	2.0358	0.1809	0.2110	0.2420	1.7925	3.2558	0.7003	0.3308

TABLE III
RMSE OF DIFFERENT ALGORITHMS ON THE YAW

Index	Yaw (deg)				
	MKMC	GD	ESKF	IGD	VQF
6	0.3581	4.4078	7.8615	5.1642	2.9495
7	0.2726	1.9509	1.3227	1.3768	0.8566
8	0.2604	1.6842	1.0041	1.2250	0.5759
9	0.4098	1.7093	1.0888	1.0490	0.3104
10	1.4697	19.7168	20.5097	17.1180	11.7320
11	1.4911	20.8336	21.4068	18.0540	12.9183
12	1.1269	24.4531	24.9301	22.2111	17.2807

overall RMSE with a constraint of the yaw error, i.e., $m = 5$ and $\lambda = 2.0\,\mathrm{deg}$ in (30). The kernel bandwidth domain is set to be $\mathscr{Z} = [0.1, 10]$, which is capable of most applications. The initial candidate points are randomly selected and the initial point number is 2. The maximum iteration number for BO is 100. The optimization procedure is conducted on MATLAB 2019b based on equations (14)–(24). The optimal kernel bandwidths are $\sigma_1^* = 1.6188$ and $\sigma_2^* = 0.4234$.

C. Results

We validate the performance of MKMCKF-OE on the testing set (experiments 6–12) and compared it with the GD [11], ESKF [15], IGD [16], and VQF [31]. The corresponding RMSE performance is summarized in Tables II and III (therein, we use the MKMC to denote the MKMCKF-OE). The detailed experimental description can be found in Table I. One can see that the overall performance of the MKMCKF-OE is better than the others, especially on the yaw axis with obvious magnetic disturbance in experiments 6, 10, 11, and 12.

The Euclidean norm of accelerometer readings and magnetometer readings can reflect the level of A_d and M_d where their values should be equal to the acceleration of gravity and the local geomagnetic field strength when without disturbance. In experiment 8, the accelerometer norm and the magnetometer norm are shown in Fig. 2. One can see that the IMU encounters big acceleration disturbance while small magnetic disturbance. The corresponding orientation errors are depicted in Fig. 3. One can see that the overall performance of MKMCKF-OE is superior to the GD, ESKF, IGD, and VQF, especially along the yaw and pitch axes.

We also visualize the norm of accelerometer readings and magnetometer readings in experiment 11 where the time is

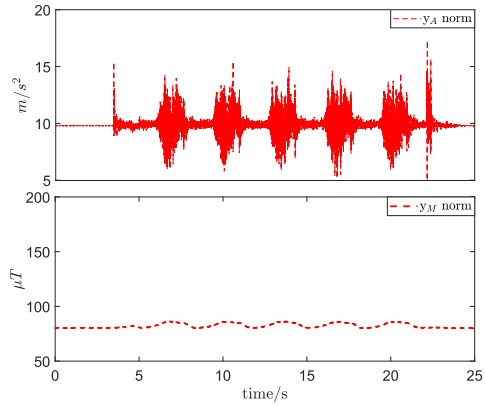

Fig. 2. Norms of the accelerometer readings and the magnetometer readings in experiment 8.

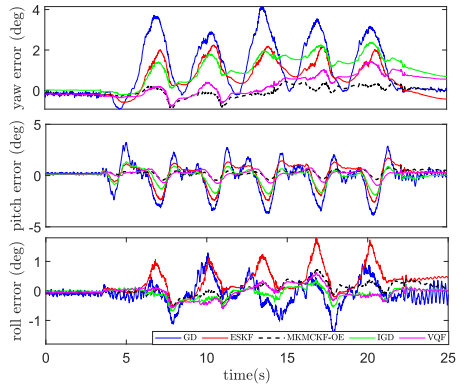

Fig. 3. Estimation errors of the GD, ESKF, MKMCKF-OE, IGD, and VQF in experiment 8.

divided into regions of no disturbance, static with M_d, rotation with M_d and A_d (see Fig. 4). The corresponding error performance is shown in Fig. 5. One can see that a very big yaw error is caused by the magnetic disturbance in the blue region by the GD, ESKF, IGD, and VQF, while it is avoided by the MKMCKF-OE. In the yellow region where the IMU rotates with both M_d and A_d, the performance of the MKMCKF-OE is again significantly better than the others, which reveals that our algorithm is very robust to both acceleration disturbance and magnetic disturbance. The IGD and VQF achieve good performance for the inclination estimation (the roll and pitch).

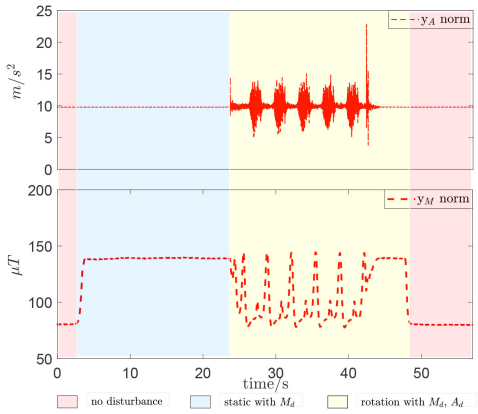

Fig. 4. Norms of the accelerometer readings and the magnetometer readings in experiment 11. The constant magnetic disturbance is added in the blue region. Then, the IMU rotates with both A_d and M_d.

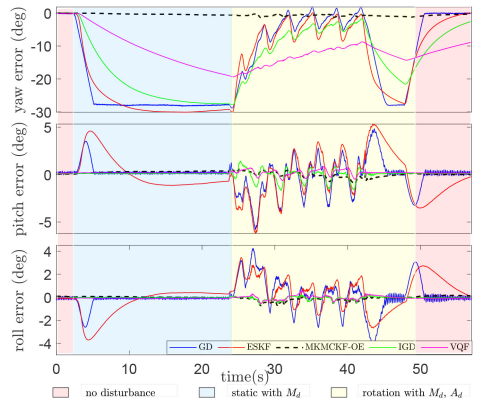

Fig. 5. Estimation errors of the GD, ESKF, MKMCKF-OE, IGD, and VQF in experiment 11.

However, their performances on the yaw are not satisfactory when there is magnetic disturbance.

V. Conclusion

In this letter, we design a new algorithm MKMCKF-OE for orientation estimation which is robust to acceleration disturbance and magnetic disturbance. The kernel bandwidths are obtained by BO which is both efficient and convenient. The effectiveness of the proposed method is validated by a set of experiments and compared with the benchmark methods. Experiments show that our proposed method outperforms the GD and the ESKF significantly, and is superior to the IGD and VQF when there is magnetic disturbance. The RMSE of MKMCKF-OE is less than 1.4911° in the yaw and is smaller than 0.4629° in the roll and pitch even under the worst testing case.

Appendix

A. MKMCKF Algorithm

The detail algorithm of the MKMCKF is shown in Algorithm 2.

Algorithm 2: MKMCKF.

Step 1: Initialization
Select kernel vector σ_p, σ_r, and a threshold ε.
Step 2: Main Algorithm
$\hat{\mathrm{x}}_k^- = \mathrm{A}\hat{\mathrm{x}}_{k-1}^+$
$\mathrm{P}_k^- = \mathrm{A}\mathrm{P}_{k-1}^+\mathrm{A}^T + \mathrm{Q}$
Obtain T_p with $\mathrm{P}_k^- = \mathrm{T}_p\mathrm{T}_p^T$
Obtain T_r with $\mathrm{R} = \mathrm{T}_r\mathrm{T}_r^T$
$\hat{\mathrm{x}}_{k,0}^+ = \hat{\mathrm{x}}_k^-$
while $\frac{\|\hat{\mathrm{x}}_{k,t}^+ - \hat{\mathrm{x}}_{k,t-1}^+\|}{\|\hat{\mathrm{x}}_{k,t-1}^+\|} > \varepsilon$ **do**
  $\hat{\mathrm{x}}_{k,t}^+ = \hat{\mathrm{x}}_k^- + \tilde{\mathrm{K}}_{k,t}(\mathrm{y}_k - \mathrm{C}\hat{\mathrm{x}}_k^-)$ ▷t starts from 1
  $\tilde{\mathrm{K}}_{k,t} = \tilde{\mathrm{P}}_k^-\mathrm{C}^T(\mathrm{C}\tilde{\mathrm{P}}_k^-\mathrm{C}^T + \tilde{\mathrm{R}}_k)^{-1}$
  $\tilde{\mathrm{P}}_k^- = \mathrm{T}_p\tilde{\mathrm{M}}_p^{-1}\mathrm{T}_p^T$
  $\tilde{\mathrm{R}}_k = \mathrm{T}_r\tilde{\mathrm{M}}_r^{-1}\mathrm{T}_r^T$
  $\tilde{\mathrm{M}}_p = \mathrm{diag}(\mathrm{G}_{\sigma_p}(\mathrm{e}_p))$
  $\tilde{\mathrm{M}}_r = \mathrm{diag}(\mathrm{G}_{\sigma_r}(\mathrm{e}_r))$
  $\mathrm{e}_p = \mathrm{T}_p^{-1}\hat{\mathrm{x}}_k^- - \mathrm{T}_p^{-1}\hat{\mathrm{x}}_{k,t-1}^+$
  $\mathrm{e}_r = \mathrm{T}_r^{-1}\mathrm{y}_k - \mathrm{T}_r^{-1}\mathrm{C}\hat{\mathrm{x}}_{k,t-1}^+$
  $t = t + 1$
end while
$\mathrm{P}_k^+ = (\mathrm{I} - \tilde{\mathrm{K}}_k\mathrm{C})\mathrm{P}_k^-(\mathrm{I} - \tilde{\mathrm{K}}_k\mathrm{C})^T + \tilde{\mathrm{K}}_k\mathrm{R}\tilde{\mathrm{K}}_k^T$

B. Error Covariance Matrix Updates

As reported in [30, p. 22–24], the *a priori* estimate of the error covariance can be written as

$$P_k^- = \begin{bmatrix} P_k^-(\theta_\varepsilon,\theta_\varepsilon), P_k^-(\theta_\varepsilon,b_\varepsilon), P_k^-(\theta_\varepsilon,a_\varepsilon), P_k^-(\theta_\varepsilon,d_\varepsilon) \\ P_k^-(b_\varepsilon,\theta_\varepsilon), P_k^-(b_\varepsilon,b_\varepsilon), P_k^-(b_\varepsilon,a_\varepsilon), P_k^-(b_\varepsilon,d_\varepsilon) \\ P_k^-(a_\varepsilon,\theta_\varepsilon), P_k^-(a_\varepsilon,b_\varepsilon), P_k^-(a_\varepsilon,a_\varepsilon), P_k^-(a_\varepsilon,d_\varepsilon \\ P_k^-(d_\varepsilon,\theta_\varepsilon), P_k^-(d_\varepsilon,b_\varepsilon), P_k^-(d_\varepsilon,a_\varepsilon), P_k^-(d_\varepsilon,d_\varepsilon) \end{bmatrix} \tag{31}$$

with

$$\begin{cases} P_k^-(\theta_\varepsilon,\theta_\varepsilon) \\ = P_{k-1}^+(\theta_\varepsilon,\theta_\varepsilon) + \delta t^2\left[P_{k-1}^+(b_{\varepsilon,k},b_{\varepsilon,k}) + Q_{vb} + Q_{vG}\right] \\ P_k^-(\theta_\varepsilon,b_\varepsilon) = -\delta t[P_{k-1}^+(\theta_\varepsilon,b_\varepsilon) + Q_{vb}] \\ P_k^-(\theta_\varepsilon,a_\varepsilon) = P_k^-(\theta_\varepsilon,d_\varepsilon) = 0_{3\times3} \\ P_k^-(b_\varepsilon,\theta_\varepsilon) = P_k^-(\theta_\varepsilon,b_\varepsilon)^T \\ P_k^-(b_\varepsilon,b_\varepsilon) = P_{k-1}^+(b_\varepsilon,b_\varepsilon) + Q_{vb} \\ P_k^-(b_\varepsilon,a_\varepsilon) = P_k^-(b_\varepsilon,d_\varepsilon) = 0_{3\times3} \\ P_k^-(a_\varepsilon,\theta_\varepsilon) = P_k^-(a_\varepsilon,b_\varepsilon) = P_k^-(a_\varepsilon,d_\varepsilon) = 0_{3\times3} \\ P_k^-(a_\varepsilon,a_\varepsilon) = \eta_a^2 P_{k-1}^+(a_\varepsilon,a_\varepsilon) + Q_{va} \\ P_k^-(d_\varepsilon,\theta_\varepsilon) = P_k^-(d_\varepsilon,b_\varepsilon) = P_k^-(d_\varepsilon,a_\varepsilon) = 0_{3\times3} \\ P_k^-(d_\varepsilon,d_\varepsilon) = \eta_m^2 P_{k-1}^+(d_\varepsilon,d_\varepsilon) + Q_{vd} \end{cases} \tag{32}$$

where Q_{vb}, Q_{vG}, Q_{va}, Q_{vd} are the covariance of the gyroscope offset, the gyroscope signals, the linear accelerations and the magnetic disturbance. One can refer to [15], [30] for a detailed description.

C. Magnetic Jamming and Inclination Angle Tracking

To detect magnetic jamming, we apply the following threshold-based classifier:

$$M_J = \begin{cases} 1, \|^S\hat{d}_{\varepsilon,k}^+\|^2 > \tau^2 \\ 0, \text{otherwise} \end{cases} \tag{33}$$

where ${}^{S}\hat{d}^{+}_{\varepsilon,k}$ is the error magnetic disturbance, τ is a threshold, $\|\cdot\|$ is the Euclidean norm.

Then, if there is magnetic jamming with $M_J = 1$, it implies that the magnetometer signals are not reliable. In this case, the error state should be updated only based on the accelerometer readings and the gyroscope readings (see [30, p.19] for details). Moreover, the update of ${}^{S}\hat{d}^{+}_{\varepsilon,k}$ and ${}^{G}m_k$ (Line 29 and 30) in algorithm 1 should be ignored. Otherwise, we should reconstruct ${}^{G}m_k$ to minimize the influence of magnetic disturbance on the pitch and the roll as follows

$$\varphi_{mag,k} = \tan^{-1}\left(\frac{{}^{G}m_{z,k-1} - {}^{G}\hat{d}^{+}_{\varepsilon z,k}}{{}^{G}m_{x,k-1} - {}^{G}\hat{d}^{+}_{\varepsilon x,k}}\right) \tag{34}$$

where $\varphi_{mag,k}$ is the inclination angle at time step k, and the subscript x and z denotes the first and the third element of the corresponding vector, respectively. Then, the geomagnetic vector is updated as

$${}^{G}m_k = \begin{pmatrix} M_{sth}\cos(\varphi_{mag,k}) \\ 0 \\ M_{sth}\sin(\varphi_{mag,k}) \end{pmatrix}. \tag{35}$$

where M_{sth} is the local geomagnetic field strength. A full treatment of this model can be found in [30, p.19].

REFERENCES

[1] I. Prayudi and D. Kim, "Design and implementation of IMU-based human arm motion capture system," in *Proc. IEEE Int. Conf. Mechatronics Automat.*, 2012, pp. 670–675.

[2] W. Huo, S. Mohammed, Y. Amirat, and K. Kong, "Fast gait mode detection and assistive torque control of an exoskeletal robotic orthosis for walking assistance," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1035–1052, Aug. 2018.

[3] Y. Ding, I. Galiana, C. Siviy, F. A. Panizzolo, and C. Walsh, "IMU-based iterative control for hip extension assistance with a soft exosuit," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 3501–3508.

[4] J. Wendel, O. Meister, C. Schlaile, and G. F. Trommer, "An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter," *Aerosp. Sci. Technol.*, vol. 10, no. 6, pp. 527–533, 2006.

[5] M. Hao, K. Chen, and C. Fu, "Smoother-based 3-D foot trajectory estimation using inertial sensors," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 12, pp. 3534–3542, Dec. 2019.

[6] F. Wittmann *et al.*, "Assessment-driven arm therapy at home using an IMU-based virtual reality system," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, 2015, pp. 707–712.

[7] A. Sabatini, "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 7, pp. 1346–1356, Jul. 2006.

[8] R. V. Vitali, R. S. McGinnis, and N. C. Perkins, "Robust error-state Kalman filter for estimating IMU orientation," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3561–3569, Feb. 2021.

[9] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Trans. Autom. Control*, vol. 53, no. 5, pp. 1203–1218, Jun. 2008.

[10] K. Kanjanapas, Y. Wang, W. Zhang, L. Whittingham, and M. Tomizuka, "A human motion capture system based on inertial sensing and a complementary filter," in *Proc. Dynamic Syst. Control Conf.*, 2013, Art. no. V003T40A004.

[11] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, 2011, pp. 1–7.

[12] M. Admiraal, S. Wilson, and R. Vaidyanathan, "Improved formulation of the IMU and MARG orientation gradient descent algorithm for motion tracking in human-machine interfaces," in *Proc. IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, 2017, pp. 403–410.

[13] A. El Hadri and A. Benallegue, "Sliding mode observer to estimate both the attitude and the gyro-bias by using low-cost sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 2867–2872.

[14] L. Xiang and L. Qun, "External acceleration elimination for complementary attitude filter," in *Proc. IEEE Int. Conf. Inf. Automat.*, 2017, pp. 208–212.

[15] D. Roetenberg, H. Luinge, C. Baten, and P. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 13, no. 3, pp. 395–405, Sep. 2005.

[16] T. Seel and S. Ruppin, "Eliminating the effect of magnetic disturbances on the inclination estimates of inertial sensors," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8798–8803, 2017.

[17] B. Chen, X. Wang, N. Lu, S. Wang, J. Cao, and J. Qin, "Mixture correntropy for robust learning," *Pattern Recognit.*, vol. 79, pp. 318–327, 2018.

[18] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: Properties and applications in non-Gaussian signal processing," *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5286–5298, Nov. 2007.

[19] L. Bako, "Robustness analysis of a maximum correntropy framework for linear regression," *Automatica*, vol. 87, pp. 218–225, 2018.

[20] B. Chen, X. Liu, H. Zhao, and J. C. Principe, "Maximum correntropy Kalman filter," *Automatica*, vol. 76, pp. 70–77, 2017.

[21] X. Liu, Z. Ren, H. Lyu, Z. Jiang, P. Ren, and B. Chen, "Linear and nonlinear regression-based maximum correntropy extended Kalman filtering," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 5, pp. 3093–3102, May 2021.

[22] X. Liu, H. Qu, J. Zhao, and P. Yue, "Maximum correntropy square-root cubature Kalman filter with application to SINS/GPS integrated systems," *ISA Trans.*, vol. 80, pp. 195–202, 2018.

[23] X. Liu, B. Chen, B. Xu, Z. Wu, and P. Honeine, "Maximum correntropy unscented filter," *Int. J. Syst. Sci.*, vol. 48, no. 8, pp. 1607–1615, 2017.

[24] A. Singh and J. C. Principe, "Using correntropy as a cost function in linear adaptive filters," in *Proc. Int. Joint Conf. Neural Netw.*, 2009, pp. 2950–2955.

[25] A. Aravkin, J. V. Burke, L. Ljung, A. Lozano, and G. Pillonetto, "Generalized Kalman smoothing: Modeling and algorithms," *Automatica*, vol. 86, pp. 63–86, 2017.

[26] S. Li, D. Shi, W. Zou, and L. Shi, "Multi-kernel maximum correntropy Kalman filter," *IEEE Control Syst. Lett.*, vol. 6, pp. 1490–1495, 2022.

[27] J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham, "Bayesian optimization with inequality constraints," in *Proc. Int. Conf. Mach. Learn.*, 2014, vol. 2014, pp. 937–945.

[28] K. Kawaguchi, L. P. Kaelbling, and T. Lozano-Pérez, "Bayesian optimization with exponential convergence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, vol. 28, pp. 2809–2817.

[29] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, vol. 25, pp. 2951–2959.

[30] M. S. Mark Pedley and Z. Baranski, "Freescale sensor fusion Kalman filter," *GitHub Repository*, pp. 24–26, 2014. [Online]. Available: https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion/tree/master/docs

[31] D. Laidig and T. Seel, "VQF: Highly accurate IMU orientation estimation with bias estimation and magnetic disturbance rejection," 2022, *arXiv:2203.17024*.

[32] K. P. Murphy, *Machine Learning: A. Probabilistic Perspective*, Cambridge, MA, USA: MIT Press, 2012.

[33] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, Cambridge, MA, USA: MIT Press, 2005.

[34] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv:1807.02811*.

[35] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," Tech. Rep. UBC TR-2009-023, 2009, *arXiv:1012.2599*.

[36] S. Li, "Code and data for MKMCKF-OE," *GitHub Repository*, 2022. [Online]. Available: https://github.com/lsl-zsj/MKMCKF-OE