Joint Constraints Based Dynamic Calibration of IMU Position on Lower Limbs in IMU-MoCap

Qian Hu, Lingfeng Liu *, Feng Mei and Changxuan Yang

School of Information Engineering, East China Jiao Tong University, Shuanggang Dong Dajie 808, Nanchang 330013, China; huqian_ecjtu@163.com (Q.H.); mei_feng3@163.com (F.M.); changxuan_y@163.com (C.Y.)

* Correspondence: lingfeng.liu@ecjtu.edu.cn

Abstract: The position calibration of inertial measurement units (IMUs) is an important part of human motion capture, especially in wearable systems. In realistic applications, static calibration is quickly invalid during the motions for IMUs loosely mounted on the body. In this paper, we propose a dynamic position calibration algorithm for IMUs mounted on the waist, upper leg, lower leg, and foot based on joint constraints. To solve the problem of IMUs' position displacement, we introduce the Gauss–Newton (GN) method based on the Jacobian matrix, the dynamic weight particle swarm optimization (DWPSO), and the grey wolf optimizer (GWO) to realize IMUs' position calibration. Furthermore, we establish the coordinate system of human lower limbs to estimate each joint angle and use the fusion algorithm in the field of quaternions to improve the attitude calibration performance of a single IMU. The performances of these three algorithms are analyzed and evaluated by gait tests on the human body and comparisons with a high-precision IMU-Mocap reference device. The simulation results show that the three algorithms can effectively calibrate the IMU's position for human lower limbs. Additionally, when the degree of freedom (DOF) of a certain dimension is limited, the performances of the DWPSO and GWO may be better than GN, when the joint changes sufficiently, the performances of the three are close. The results confirm that the dynamic calibration algorithm based on joint constraints can effectively reduce the position offset errors of IMUs on upper or lower limbs in practical applications.

Keywords: IMU; MoCap; DWPSO; GWO; Gauss–Newton; joint constraints

1. Introduction

In recent years, inertial measurement units (IMUs) have attracted increasing interest in the field of human motion analysis. The wearable sensor motion capture system is less costly, more flexible, and more portable than optical camera-based motion capture devices [1,2]. By mounting IMUs on each limb of human bodies, the real-time tracking and motion data analysis of human postures can be realized. The IMU-based motion capture and analysis have shown substantial applications in athletic training, e.g., golf training, baseball training, dart-throwing training, etc. [3–6]. It also has promising application prospects in medical rehabilitation training. In [7], the IMU-Mocap system is applied to determine the level of autonomy for patients with Parkinsonism syndromes. To obtain the information of body motions, the installation position of the IMUs and the variation of joint space position should be accurately measured. The works [1,8] analyzed the influence of IMUs' positions or directions on the accuracy of motion evaluation, and it will further affect the variation of the joint angle. However, in practical applications, different types of clothing materials and muscle stretching during exercise will cause IMU position displacements. A direct consequence of IMU displacement is the difference of the derived joint position relative to the pre-calibration. Therefore, an effective IMU position calibration method is necessary.

Regarding IMU position displacement, there are currently two types of IMU position calibration, divided into static calibration, e.g., quiet standing, and dynamic calibration, e.g., knee flexion. For static calibration, the work [9] realized an IMU-to-body calibration based on preset static postures. In [10,11], the direction of the IMUs was estimated by a T-pose or N-pose to ensure that each frame in the IMUs aligns with the known direction in that posture. For dynamic calibration, the works [12,13] calibrated the IMUs bound to the upper leg and lower leg by using flexion/extension (FE) and abduction/adduction (AA) of the knee joint, so that the angle of the knee joint is equal to zero in the standing posture. The studies [14,15] proposed a simple calibration scheme, which does not need to specify the motion of limbs, and used human gait analysis to align the direction of IMUs to the body. However, the above methods do not estimate the positions of the sensors relative to the adjacent limbs, which is critical information for calculating the joint angle using IMUs, especially during fast rotations of the joint [16], and establish a motion chain model in high-speed motions [17]. The study [18] proposed a position estimation algorithm to estimate the position of IMUs relative to the limbs based on the least-squares optimization. This approach was further extended in [19,20] for gait analysis and the angles of the knee and ankle joint were estimated; it was then applied to the rehabilitation system of human limbs.

The limitations of the above calibration methods can be summarized as follows:

(1) The wearing position of the sensor needs to be fixed or special tools are required [9–11].

(2) The limbs are required to perform specific motions, but it is difficult for volunteers with damaged joints to complete them. Even volunteers with normal physical activities need to be guided by professionals [12,13].

(3) The direction of IMUs must be estimated in advance to complete the calibration, and it is easy to make mistakes, especially when using a magnetometer, which is vulnerable to the interference of magnetic field [14,15,21].

(4) When the joint rotates, it cannot be fully rotated in all specified directions, resulting in the decline of calibration accuracy [18,19].

Problem Statement

Considering the limitations of previous calibration algorithms, and based on the joint constraints proposed in [18], in this work, we aim to study the influence of different algorithms on the dynamic calibration performance of IMUs' position based on joint constraints. Furthermore, we aim to establish the human lower limb coordinate system and calculate the joint angle to study the influence of IMUs' position accuracy on human gait space-time parameters. The study [18] did not provide research on the difference of results caused by different joint motion types in the actual motion process. Due to the different variation ranges of the hinge joint and spherical joint, the degree of freedom (DOF) of the joint will change. When the joint rotation is insufficient, the result of the Gaussian–Newton (GN) algorithm based on the Jacobian matrix may be inaccurate. To solve the IMU position displacement and consider the influence of the change of joint DOF on the calibration algorithm, we introduce the dynamic weight particle swarm optimization (DWPSO) [22] and grey wolf optimizer (GWO) [23] to realize the position calibration of IMUs based on the joint constraints, and the calibration results of the two algorithms are compared with GN. The main contents of this work are as follows:

(1) The four IMUs are bound to the waist, upper leg, lower leg, and foot, respectively, for the gait experiment, and the data of accelerometers and gyroscopes of each IMU are collected. High-precision IMU-Mocap equipment is bound on the lower limbs for synchronous data acquisition with IMUs. High-precision motion capture equipment is only for reference. In addition, we place IMUs in two different positions, and three subjects are tested in two positions, including one female of height 165 cm, and two males of height 175 cm and 180 cm respectively.

(2) The collected data are substituted into the GN, DWPSO, and GWO for position estimation to obtain the position information of IMUs relative to the limbs.

(3) Using the calibrated IMU position information, we establish the human lower limb coordinate system to calculate the angles of the hip, knee, and ankle joint in each DOF. Additionally, to improve the performance of attitude calibration, the quaternion fusion algorithm is used to fuse the data of the accelerometer and gyroscope of single IMU.

(4) The performance of the three algorithms is evaluated by comparing them with the high-precision IMU-Mocap reference device.

The following of the paper is organized into 5 parts. In Section 2, the IMU position calibration model of the spherical joints and the hinge joints is introduced, respectively. In Section 3, analyzing the performance of the GN, and points out its limitations. Under the same constraints as the GN, the DWPSO and GWO are used to calibrate IMUs' positions. Section 4 establishes the coordinate system of human lower limbs, combines the position information of IMUs and the attitude of single IMU to calculate the joint angle of human lower limbs during walking. Section 5 introduces the experimental test device and test scheme and analyzes the test data by analyzing the angle variation of each joint angle to verify the performance of the three calibration algorithms. Finally, Section 6 summarizes the study.

2. IMU Position Calibration Principle

In this study, we focus on the calibration of IMUs' positions relative to the lower limb joints. According to the international society of biomechanics (ISB) standard [24] and the joints coordinate system defined in human anatomy [25], 3D rotation for the lower limbs joints can be defined as: (1) hip: flexion/extension (H_{FE}), abduction/adduction (H_{AA}), internal/external rotation (H_{IE}), (2) knee: flexion/ extension (K_{FE}), abduction/adduction (K_{AA}), internal/external rotation (K_{IE}), (3) ankle: flexion/extension (A_{FE}), abduction/adduction (A_{AA}), internal/external rotation (A_{IE}).

For the IMU position estimation, human lower limbs can be simplified as rigid segments connected by joints. Figure 1 presents a model of human left lower limbs; as the human body is symmetrical, it can also be applied to the right lower limbs. The four IMUs are denoted as S, $S \in \{A, B, C, D\}$ being mounted on the waist, upper leg, lower leg, and foot. The hip, knee, and ankle joints are denoted as J_i, $J_i \in \{J_H, J_K, J_A\}$. r_{J_K} is the rotation axis of the knee joint. O_g is the global coordinate system, which represents the coordinate system of the 3D space object. O_s, $S \in \{A, B, C, D\}$ is the sensor coordinate system, which takes IMUs' center as the coordinate origin.
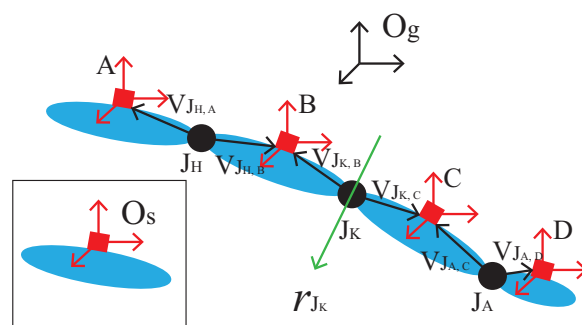
Figure 1. A model of a human left lower limb connected by joints.

The human lower limb joints can be classified into a spherical joint and hinge joint, where the hip and ankle are spherical joints and the knee joint is the hinge joint. At time step t ($t = 1 \ldots n$), the accelerations measured by the accelerometers are denoted as $a_S(t)$. The angular velocities measured by the gyroscope are denoted as $w_S(t)$, $S \in \{A, B, C, D\}$.

(1) The spherical joint

The spherical joint is analyzed on the hip joint as an example and is applicable for the ankle joint. Assuming A and B are connected through the joint J_H, the position of the

two IMUs relative to the joint can be determined from the sequence of measurements of A and B. Let $V_{J_H, S} = [x_{J_H, S}, y_{J_H, S}, z_{J_H, S}]^T, S \in \{A, B\}$ denote the vector pointing from the joints center to the origin of the two IMUs coordinate systems in Figure 1. Through the IMU position estimation algorithm proposed in [18], the spherical joint model is defined by Equation (1).

$$\|\boldsymbol{a}_A(t) - \boldsymbol{\Gamma}_A(t)\| - \|\boldsymbol{a}_B(t) - \boldsymbol{\Gamma}_B(t)\| = 0, \tag{1}$$

$$\boldsymbol{\Gamma}_S(t) = \boldsymbol{w}_S(t) \times (\boldsymbol{w}_S(t) \times V_{J_H, S}) + \boldsymbol{\alpha}_S(t) \times V_{J_H, S}, S \in \{A, B\}, \tag{2}$$

where $\|\cdot\|$ is the norm for vectors, $\times$ is a cross product. The angular acceleration calculated by the angular velocity is denoted as $\boldsymbol{\alpha}_S$, which is defined by [19].

The IMU position displacement will cause the errors of Equation (1), and the equation is not equal on the left and right. The errors are defined by Equation (3).

$$e_{J_H}(t) = \|\boldsymbol{a}_A(t) - \boldsymbol{\Gamma}_A(t)\| - \|\boldsymbol{a}_B(t) - \boldsymbol{\Gamma}_B(t)\| \tag{3}$$

(2) The hinge joint

The knee joint is a hinge joint, and the model of the knee joint rotation axis estimation proposed in [18] is defined by Equation (4).

$$\|\boldsymbol{w}_B(t) \times \boldsymbol{r}_{J_K, B}\| - \|\boldsymbol{w}_C(t) \times \boldsymbol{r}_{J_K, C}\| = 0, \tag{4}$$

where $\boldsymbol{r}_{J_K, B}, \boldsymbol{r}_{J_K, C}$ are the coordinates of the unit vector parallel to the knee joint axis in the O_B and O_C.

Figure 2 show the coordinates of $\boldsymbol{r}_{J_K, B}$ and $\boldsymbol{r}_{J_K, C}$ in spherical coordinates, and converts it to the rectangular coordinate system in Equation (5)

$$\boldsymbol{r}_{J_K, S} = [\cos(\varphi_S)\cos(\theta_S),\ \cos(\varphi_S)\sin(\theta_S),\ \sin(\theta_S)], S \in \{B, C\}, \tag{5}$$

where $\varphi \in [0, \pi]$ is the pitch angle, $\theta \in [0, 2\pi]$ is the yaw angle. The IMU position displacement will cause Equation (4) to be unequal on the left and right. The errors are defined by Equation (6).

$$e_{J_K}(t) = \|\boldsymbol{w}_B(t) \times \boldsymbol{r}_{J_K, B}\| - \|\boldsymbol{w}_C(t) \times \boldsymbol{r}_{J_K, C}\| \tag{6}$$
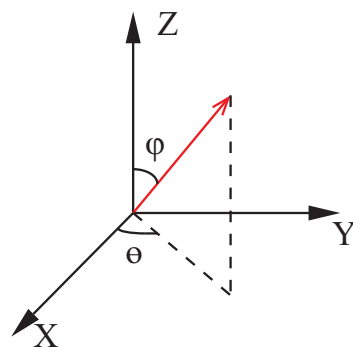

Figure 2. Spherical coordinates of $\boldsymbol{r}_{J_K, B}$ and $\boldsymbol{r}_{J_K, C}$.

3. Calibration Algorithm Design

3.1. Gauss–Newton Method for IMUs Position Calibration

By the analysis of joint constraints in Section 2, we use the Gaussian–Newton (GN) algorithm based on the Jacobian matrix to calculate Equations (3) and (6).

For Equation (3), the optimization problem is expressed by Equation (7).

$$\min_{x_{J_H}} \sum_{t=1}^{n} e_{J_H}^2(t),$$

$$x_{J_H} = [V_{J_H, A},\ V_{J_H, B}]^T,$$

$$e_{J_H}(t) = \|a_A(t) - \Gamma_A(t)\| - \|a_B(t) - \Gamma_B(t)\|,$$

(7)

where x_{J_H} is the vector containing IMUs' position parameters, and $x_{J_H, S}$, $y_{J_H, S}$, $z_{J_H, S}$, $S \in \{A, B\}$ are in the range $[-0.2,\ 0.2]$.

The iteration steps at time t are described as follows:

(1) Randomly generate initial values of $x_{J_H}^{\gamma}$, γ is the number of iterations.

(2) Calculate the deviation vector e_{J_H} using Equation (7).

(3) Calculate the Jacobian matrix $J = \frac{de_{J_H}}{dx_{J_H}}$ using Equation (8), and then calculate the generalized inverse matrix of J, which is $pinv(J)$.

$$J = \begin{bmatrix} \frac{\partial e_{J_H}(1)}{\partial V_{J_H, A}} & \frac{\partial e_{J_H}(1)}{\partial V_{J_H, B}} \\ \vdots & \vdots \\ \frac{\partial e_{J_H}(n)}{\partial V_{J_H, A}} & \frac{\partial e_{J_H}(n)}{\partial V_{J_H, B}} \end{bmatrix},$$

(8)

where

$$\frac{\partial e_{J_H}}{\partial V_{J_H, S}} = -\frac{(a_S - \Gamma_S)^T}{\|a_S - \Gamma_S\|} ([w_S]_\times [w_S]_\times + [\alpha_S]_\times), S \in \{A,\ B\}$$

(9)

the following symbols are introduced by Equation (10)

$$[w_S]_\times = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix},$$

$$[\alpha_S]_\times = \begin{bmatrix} 0 & -\alpha_z & \alpha_y \\ \alpha_z & 0 & -\alpha_x \\ -\alpha_y & \alpha_x & 0 \end{bmatrix},$$

(10)

where $w_S = [w_x,\ w_y,\ w_z]^T$, $\alpha_S = [\alpha_x,\ \alpha_y,\ \alpha_z]^T$.

(4) Update $x_{J_H}^{\gamma}$ by Equation (11) and return to (2).

$$x_{J_H}^{\gamma+1} = x_{J_H}^{\gamma} - pinv(J)e_{J_H},$$

(11)

For Equation (6), the optimization iteration is expressed by Equation (12).

$$\min_{x_{J_K}} \sum_{t=1}^{n} e_{J_K}^2(t),$$

$$x_{J_K} = [\varphi_B,\ \theta_B,\ \varphi_C,\ \theta_C]^T,$$

$$e_{J_K}(t) = \|w_B(t) \times r_{J_K, B}\| - \|w_C(t) \times r_{J_K, C}\|,$$

(12)

where x_{J_K} is the vector containing knee joint axis position parameters. The iteration steps at time t are described as follows:

(1) Randomly generate initial values of $x_{J_K}^{\gamma}$.

(2) Calculate $r_{J_K, S}$ using Equation (5)

(3) Calculate the deviation vector e_{J_K} using Equation (12).

(4) Calculate the Jacobian matrix $J = \frac{de_{J_K}}{dx_{J_K}}$ using Equation (13) and calculate the generalized inverse matrix of J is $pinv(J)$.

$$J = \begin{bmatrix} \frac{\partial e_{J_K}(1)}{\partial r_{J_K, B}} & \frac{\partial e_{J_K}(1)}{\partial r_{J_K, C}} \\ \vdots & \vdots \\ \frac{\partial e_{J_K}(n)}{\partial r_{J_K, B}} & \frac{\partial e_{J_K}(n)}{\partial r_{J_K, C}} \end{bmatrix}, \tag{13}$$

where

$$\frac{\partial(\|\boldsymbol{w}_S \times \boldsymbol{r}_{J_H, S}\|)}{\partial \boldsymbol{r}_{J_H, S}} = \frac{(\boldsymbol{w}_S \times \boldsymbol{r}_{J_H, S}) \times \boldsymbol{w}_S}{\|\boldsymbol{w}_S \times \boldsymbol{r}_{J_H, S}\|}, \ S \in \{B, \ C\} \tag{14}$$

(5) Update $\boldsymbol{x}_{J_K}^{\gamma}$ using Equation (15) and return to (2).

$$\boldsymbol{x}_{J_K}^{\gamma+1}(t) = \boldsymbol{x}_{J_K}^{\gamma}(t) - pinv(\boldsymbol{J})e_{J_K}(t) \tag{15}$$

According to the definition of the DH coordinate system in [26], the three DOF (3-DOF) joints of the hip and ankle can be divided into three hinge joints. Therefore, the position of the IMUs relative to the knee joint can be calculated using the spherical joint approach. The positions of B and C relative to the knee joint can be obtained by Equation (16).

$$\begin{aligned} \boldsymbol{V}_{J_K, B} &= \widetilde{\boldsymbol{V}}_{J_K, B} - \frac{1}{2}(\boldsymbol{r}_{J_K, B}^T \cdot \widetilde{\boldsymbol{V}}_{J_K, B} + \boldsymbol{r}_{J_K, C}^T \cdot \widetilde{\boldsymbol{V}}_{J_K, C})\boldsymbol{r}_{J_K, B}, \\ \boldsymbol{V}_{J_K, C} &= \widetilde{\boldsymbol{V}}_{J_K, C} - \frac{1}{2}(\boldsymbol{r}_{J_K, B}^T \cdot \widetilde{\boldsymbol{V}}_{J_K, B} + \boldsymbol{r}_{J_K, C}^T \cdot \widetilde{\boldsymbol{V}}_{J_K, C})\boldsymbol{r}_{J_K, C}, \end{aligned} \tag{16}$$

where $\widetilde{\boldsymbol{V}}_{J_K, B}$ and $\widetilde{\boldsymbol{V}}_{J_K, C}$ are the estimated by Equation (7).

By analyzing the algorithm, the limitations of the GN are as follows:

(1) In the process of using the GN, the Jacobi matrix theoretically needs to be positive definite; however, the calculation may not be of full rank. When people walk, the motion of the knee joint is mainly flexion and extension, i.e., there is a significant change in only one DOF. When in other DOF, such as internal/external rotation of the knee joint, $w_x = w_y = 0$ cause $\alpha_x = \alpha_y = 0$. According to the analysis of Equations (8)–(10), this will reduce the rank of the Jacobian matrix. For the hip or ankle joint, it is also not guaranteed that each motion produces rotation in all 3-DOF at the same time, which reduces the rank of $\boldsymbol{J}$. The matrix $\boldsymbol{J}$ may be singular and leads to non-convergence of the algorithm.

(2) In the process of practical calculation, the GN is iterated along with one or two of the matrix entries, which causes the high complexity of the $\boldsymbol{J}$. Therefore, each joint can only be calculated separately. If the motion data of the whole lower limb is processed at the same time, it will increase the complexity of the algorithm and make the iteration time longer and the performance of the GN will be compromised.

3.2. Dynamic Weight Particle Swarm Optimization for IMUs Position Calibration

Both the population algorithm and genetic algorithm simulate the adaptability of the individual population on the basis of natural characteristics and use some transformation rules to solve the optimal solution through the search space. However, individual variation will occur in the process of the genetic algorithm, which cannot completely solve the constraints in the optimization problem [27]. Therefore, we chose the population optimization algorithm to realize the position calibration of IMUs.

Under the same constraints as [18], we introduce dynamic weight particle swarm optimization (DWPSO) to calibrate the position of IMUs, where the dynamic weight is added to the traditional PSO. Unlike the GN, the DWPSO does not need to consider the complexity of the Jacobian matrix calculation.

At time t, let N is the number of particles in the population, and ε is the ε-th particle of all particles. The parameter vector containing the IMUs' positions is expressed by Equation (17). The vector containing knee joint axis position parameters is expressed by Equation (18).

$$x_{J_H, \varepsilon} = [V_{J_H, A, \varepsilon}, \ V_{J_H, B, \varepsilon}]^T ,\tag{17}$$

$$x_{J_K, \varepsilon} = [\varphi_{B, \varepsilon}, \ \theta_{B, \varepsilon}, \ \varphi_{C, \varepsilon}, \ \theta_{C, \varepsilon}]^T,\tag{18}$$

For Equations (3) and (6), the optimization is expressed by Equations (19)–(21).

$$\eta_\varepsilon = \operatorname{argmin}(e_{J_i}(x^\gamma_{J_i, \varepsilon})),\tag{19}$$

where η_ε is individual extreme as optimal value of x_{J_i} found by the particle ε in the iteration, $J_i \in \{J_H, \ J_K\}$. γ is the number of iterations.

$$p = \operatorname{argmin}(e_{J_i}(\eta_\varepsilon)),\tag{20}$$

where p is the global extremum, i.e., all particles find the optimal value of x_{J_i} in the iterative.

$$F_{J_H, \varepsilon} = \begin{bmatrix} Fx_{J_H, A, \varepsilon} & Fx_{J_H, B, \varepsilon} \\ Fy_{J_H, A, \varepsilon} & Fy_{J_H, B, \varepsilon} \\ Fz_{J_H, A, \varepsilon} & Fz_{J_H, B, \varepsilon} \end{bmatrix},$$

$$F_{J_K, \varepsilon} = [F\varphi_{B, \varepsilon}, \ F\theta_{B, \varepsilon}, \ F\varphi_{C, \varepsilon}, \ F\theta_{C, \varepsilon}]^T,\tag{21}$$

where F_{J_H}, F_{J_K} are the update speed of particles. The algorithm steps at time t are as follows:

(1) Initialization $x_{J_i, \varepsilon}$ and $V_{J_i, \varepsilon}$. $x_{J_H, S}$, $y_{J_H, S}$, $z_{J_H, S}$, $S \in \{A, \ B\}$ are following uniform distribution in range $[-0.2, 0.2]$. φ_B, φ_C are following uniform distribution in range $[0, \ \pi]$. θ_B, θ_C are following uniform distribution in range $[0, \ 2\pi]$.

(2) Substitute $x_{J_H, \varepsilon}$ into Equation (3) to obtain e_{J_H}. $x_{J_K, \varepsilon}$ into Equation (6) to obtain e_{J_K}.

(3) According to Equations (19) and (20) to calculate η_ε, p.

(4) Update $x_{J_i, \varepsilon}$ and $V_{J_i, \varepsilon}$ as follows:

$$F^{\gamma+1}_{J_i, \varepsilon} = \mu F^\gamma_{J_i, \varepsilon} + \Omega_1 \partial_1 (\eta_\varepsilon - x^\gamma_{J_i, \varepsilon}) + \Omega_2 \partial_2 (p - x^\gamma_{J_i, \varepsilon}),\tag{22}$$

$$x^{\gamma+1}_{J_i, \varepsilon} = x^\gamma_{J_i, \varepsilon} + F^{\gamma+1}_{J_i, \varepsilon},\tag{23}$$

where $F^{\gamma+1}_{J_i, \varepsilon}$ and $x^{\gamma+1}_{J_i, \varepsilon}$ are the values after the γ-th update. ∂_1 and ∂_2 are following uniform distribution in range $[0, 1]$. μ is the inertia factor. Ω_1 and Ω_2 are acceleration constants. This work takes $\Omega_1 = 2$, $\Omega_2 = 2$.

(5) Repeat steps 2~4 until e_{J_i} convergence.

In step 4 of the algorithm, μ will affect the performance of the algorithm. μ is too large or too small will affect the convergence of the error. Therefore, we introduce a PSO based on dynamic weight. μ decreases exponentially in range $[\mu_{\min}, \ \mu_{\max}]$ as γ increases. Additionally, to avoid falling into a local optimum when μ decreases, random jumps are introduced by Equation (24).

$$\widetilde{\mu} = \mu_{\max} \left(\frac{\mu_{\max}}{\mu_{\min}} \right)^{1 + \frac{1}{10\partial} \frac{\gamma}{\gamma_{\max}}},\tag{24}$$

where $\widetilde{\mu}$ is dynamic weight. The maximum number of iterations is denoted as $\gamma_{\max}$. Let $\mu_{\min} = 0.2$, $\mu_{\max} = 0.8$. Figure 3 shows the convergence curve of e_{J_H} when using fixed weight and the dynamic weight, and the e_{J_H} is defined by Equation (3). When the N is small, the performance of dynamic weight is better than the fixed weight. It is suitable for some scenes that require high timeliness. When the N is large, the performance of dynamic weight is similar to fixed weight.
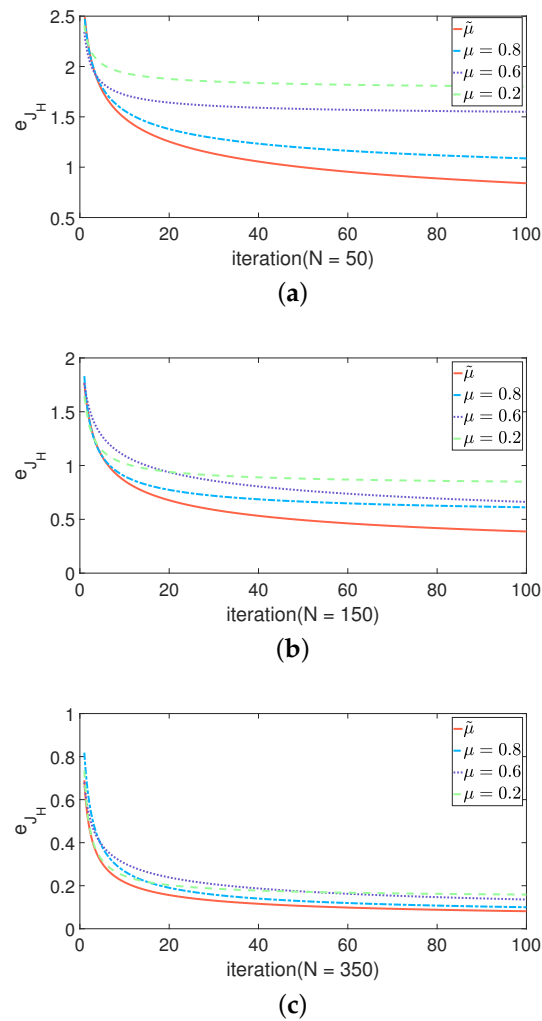
Figure 3. Convergence curves of dynamic weight and fixed weight with different numbers of particles. (**a**) $N = 50$; (**b**) $N = 150$; (**c**) $N = 350$.

3.3. Grey Wolf Optimizer for IMUs Position Calibration

To verify the influence of different population algorithms on IMUs' position calibration performance, we also introduce the grey wolf optimizer (GWO) [23] for position calibration. The GWO is a new optimization algorithm inspired by the hunting and social hierarchical behavior of gray wolves. It randomly generates a set of solutions to form an initial gray wolf group, and then iteratively selects the best three wolves in the population, similar to the optimal solution in the PSO.

The parameter vector containing the IMUs position is expressed by Equation (25). The vector containing knee joint axis position parameters is expressed by Equation (26).

$$x_{J_H,p} = \left[V_{J_H,\,A},\ V_{J_H,\,B} \right]^T,\tag{25}$$

$$x_{J_K,p} = \left[\varphi_B,\ \theta_B,\ \varphi_C,\ \theta_C \right]^T,\tag{26}$$

where p is a different individual in the population and $x_{J_i,\,p}$, $J_i \in \{J_H,\ J_K\}$ is the corresponding position of the individual.

In the GWO iteration, the position of each gray wolf represents a feasible solution in the solution space. Let the wolf with the best position be p_α; the second-best is p_β; the third-best is p_δ. The remaining wolves are p_ε, i.e., the wolf pack individuals other than the best three wolves. In each iteration, the three best gray wolves in the current population are retained, and then the positions of other search agents are updated according to their

position information. p_α, p_β, p_δ and p_ε are constantly updated and iterated until the optimal solution is found. The process of finding the optimal solution is the process of a gray wolf hunting prey.

(1) Encircling Prey

$$
\begin{aligned}
f_{p_\alpha} &= h_1 \cdot x^\gamma_{J_i,\, p_\alpha} - x^\gamma_{J_i,\, p_\varepsilon}, \\
f_{p_\beta} &= h_2 \cdot x^\gamma_{J_i,\, p_\beta} - x^\gamma_{J_i,\, p_\varepsilon}, \\
f_{p_\delta} &= h_3 \cdot x^\gamma_{J_i,\, p_\delta} - x^\gamma_{J_i,\, p_\varepsilon}, \\
h &= 2 \cdot r_1
\end{aligned}
\tag{27}
$$

where γ is the number of iterations. The positions of p_α, p_β, p_δ, p_ε in γ-th iteration are denoted by $x^\gamma_{J_i,\, p_\alpha}$, $x^\gamma_{J_i,\, p_\beta}$, $x^\gamma_{J_i,\, p_\delta}$, $x^\gamma_{J_i,\, p_\varepsilon}$, respectively. The position vectors of p_ε relative to p_α, p_β, p_δ are denoted as f_{p_α}, f_{p_β}, f_{p_δ}. The random vector following uniform distribution in range $[0, 1]$ is denoted as r_1. $h \in \{h_1,\ h_2,\ h_3\}$ is the random weight and decreases nonlinearly in iterations. From the initial iteration to the final iteration, it provides global search in the decision space. When the algorithm falls into local optimization and it is not easy to jump out, the randomness of h plays an important role in avoiding local optimization. Equation (27) shows that, after moving, the p_ε will move around the target gray wolves p_α, p_β, p_δ, and its orientation is determined by the size of each dimension and the h.

(2) Hunting

$$
\begin{aligned}
x_{J_i,\, p_1} &= x^\gamma_{J_i,\, p_\alpha} - k_1 \cdot f_{p_\alpha}, \\
x_{J_i,\, p_2} &= x^\gamma_{J_i,\, p_\beta} - k_2 \cdot f_{p_\beta}, \\
x_{J_i,\, p_3} &= x^\gamma_{J_i,\, p_\delta} - k_3 \cdot f_{p_\delta}, \\
k &= 2 \cdot \psi \cdot r_2 - \psi,
\end{aligned}
\tag{28}
$$

$$
x^{\gamma+1}_{J_i,\, p_\varepsilon} = \frac{x_{J_i,\, p_1} + x_{J_i,\, p_2} + x_{J_i,\, p_3}}{3},
\tag{29}
$$

where $k \in \{k_1,\ k_2,\ k_3\}$ is the random weight. The random vector following uniform distribution in range $[0, 1]$ is denoted as r_2. ψ is the convergence factor, which decreases linearly from 2 to 0 with the number of iterations. Combining Equations (27) and (28) shows that the p_ε moves its position by observing the positions of p_α, p_β, p_δ, and denoted as $x_{J_i,\, p_1}$, $x_{J_i,\, p_2}$, $x_{J_i,\, p_3}$, respectively. Then, use Equation (29) to determine the moving direction of the prey and update its position, i.e., $x^{\gamma+1}_{J_i,\, p_\varepsilon}$ is the updated position of the p_ε. Through a continuous iterative search, the optimal solution is found. Additionally, Equation (29) shows that the target position of the p_ε is the centroid of the area enclosed by the three positions obtained by observing the p_α, p_β, p_δ.

(3) Attacking Prey

During the iteration, when ψ decreases linearly from 2 to 0, its corresponding k also changes in range $[-\psi,\ \psi]$. When the value of k is in the range, the next position of the gray wolf can be anywhere between its current position and the prey position. When $|k| < 1$, wolves attack their prey. When $|k| > 1$, the gray wolf separates from its prey and continues to look for more suitable prey.

The algorithm steps at time t are as follows:

(1) Initialization h, k, $x_{J_i,p}$. $x_{J_H,\, S}$, $y_{J_H,\, S}$, $z_{J_H,\, S}$, $S \in \{A,\ B\}$ are following uniform distribution in range $[-0.2, 0.2]$. φ_B, φ_C are following uniform distribution in range $[0,\ \pi]$. θ_B, θ_C are following uniform distribution in range $[0,\ 2\pi]$.

(2) Calculate the individual fitness of the population by substituting $x_{J_H,p}$ into Equation (3) and $x_{J_K,p}$ into Equation (6). Select the three individuals with the smallest error as p_α, p_β, p_δ.

(3) Calculate the position vectors of p_ε relative to p_α, p_β, p_δ in γ-th iteration, respectively by Equation (27). Update the position of p_ε by Equations (28) and (29).

(4) If the maximum number of iterations is reached, go to step 6. Otherwise, go to step 5.

(5) Reorder to determine the position of the gray wolf, and go to step 2.

(6) Output current optimal solution by Equation (29).

4. Calculation of Human Lower Limbs Joint Angles

To calculate the joint angle, the coordinate system of each limb needs to be constructed first. The previous method of establishing the limb coordinate system is to let the subject stand in a standard standing posture. The limitations of this method have been analyzed in Section 1. According to the calibration algorithm in Section 3, we can obtain the installation position and direction information of human lower limb sensors and use this information to establish the coordinate system attached to a limb.

4.1. Establish the Coordinate System Attached to a Limb

In human kinematic analysis, it is crucial to determine the spatial relationship between two adjacent limbs. The establishment of the spatial relationship between two limbs depends on the coordinate frame fixed on each limb, i.e., the coordinate system attached to a limb. The commonly used method in domestic and overseas is to establish the coordinate frame on the axis of the proximal or posterior joints of each limb [28]. In this work, according to the standards of the international society of biomechanics (ISB) [24], we establish the coordinate system attached to a limb on the proximal joint, and all the coordinate systems are the right-hand Cartesian coordinates. As shown in Figure 4, the limbs of the pelvis, upper leg, lower leg, and foot of the left leg are denoted as the rod L_1, L_2, L_3, and L_4, respectively, and the coordinate system attached to each limb is denoted as $\{L_1\}$, $\{L_2\}$, $\{L_3\}$, $\{L_4\}$. In most human motions, the pelvis generally makes only translational motion without rotation; therefore, in the case of only calculating the joint angles, the L_1 rod can be considered as a fixed rod.
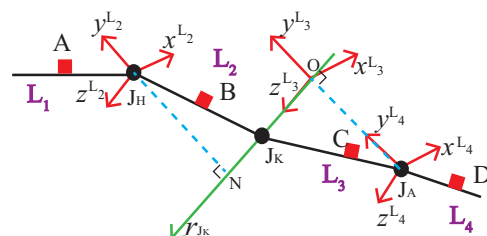
Figure 4. Structural diagram of human lower limb.

As shown in Figure 5a, $\{L_2\}$ is established at J_H. Make the normal line of r_{J_K} through J_H, and the foot point is N. The y-axes direction of the attached coordinate system is from N to J_H. The z-axes parallel to the knee joint rotation axis and the direction is left leg to right leg. The x-axes are perpendicular to the y- and z-axes.

In Figure 5b, the vertical line of the knee joint is made through the point J_A, and the vertical point is O, $\{L_3\}$ is established at O. The y-axes direction is from J_A to O. The z-axes coincides with the knee joint axis, and the direction is the left leg to right leg. The x-axes are perpendicular to the y and z-axes, $\{L_4\}$ is established at J_A. At the initial stage, $\{L_1\}$ and $\{L_2\}$ coincide, and $\{L_3\}$ and $\{L_4\}$ are parallel.
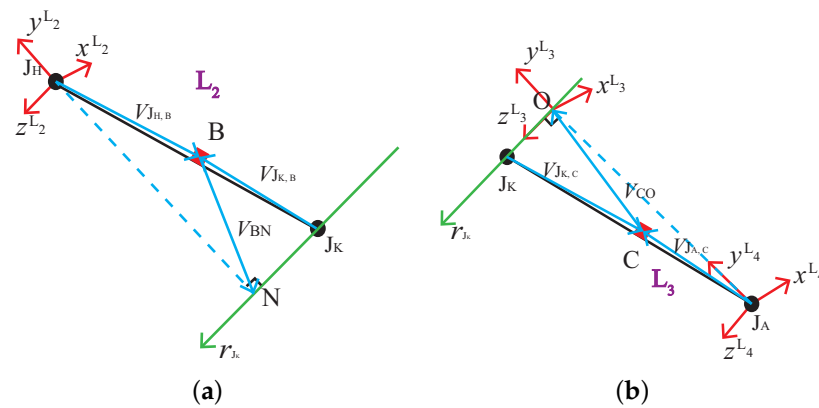
Figure 5. Structure diagram of limbs and position vector of each joint. (**a**) Limb L_2; (**b**) Limb L_3.

The attitude transformation between two adjacent sensors is defined by Equation (30).

$$
\begin{aligned}
R_B^A(t) &= (R_A^{O_g}(t))^T R_B^{O_g}(t), \\
R_C^B(t) &= (R_B^{O_g}(t))^T R_C^{O_g}(t), \\
R_D^C(t) &= (R_C^{O_g}(t))^T R_D^{O_g}(t),
\end{aligned}
\tag{30}
$$

where R_B^A is the attitude transformation matrix between A and B. The attitude transformation matrix between B and C is denoted as R_C^B. The attitude transformation matrix between C and D is denoted as R_D^C. The attitude transformation matrix of each IMU relative to the global coordinate system at time t is denoted as $R_S^{O_g}$, $S \in \{A,\ B,\ C,\ D\}$. It is worth noting that only the sensor coordinate system O_s is time-dependent, resulting in $R_S^{O_g}$ changing with time and O_g is not time-dependent. We analyze the fusion process in detail in Section 4.3. When the human body has not started motion capture and stands still, the attitude transformations between IMUs are denoted by $R_B^A(0)$, $R_C^B(0)$, $R_D^C(0)$.

According to the analysis in Section 3, the $V_{J_H,\ B}$, $V_{J_K,\ B}$, $V_{J_K,\ C}$, $V_{J_A,\ C}$, $V_{J_A,\ D}$, $r_{J_K,\ B}$, and $r_{J_K,\ C}$ can be obtained. As shown in Figure 5a,b, V_{BN} and V_{CO} need to be calculated to obtain the coordinate system attached to each limb. Establishing the equation of the knee joint axis by Equation (31).

$$
\frac{x + x_2}{r_x} = \frac{y + y_2}{r_y} = \frac{z + z_2}{r_z},
\tag{31}
$$

where $[x_2,\ y_2,\ z_2]^T$ denote the coordinates of $V_{J_H,\ B}$ and $[x_3,\ y_3,\ z_3]^T$ denote the coordinates of $V_{J_K,\ B}$. $[r_x,\ r_y,\ r_z]^T$ denote the coordinates of $r_{J_K,\ B}$. $[x,\ y,\ z]^T$ is any point on a line, and point N is on this line. Assume that the coordinates of V_{BN} are $[a,\ b,\ c]^T$, the equation of knee joint axis by Equation (32).

$$
\frac{a + x_2}{r_x} = \frac{b + y_2}{r_y} = \frac{c + z_2}{r_z}
\tag{32}
$$

The normal line of r_{J_K} can be expressed by Equation (33).

$$
r_x(a + x_3) + r_y(b + y_3) + r_z(a + z_3) = 0
\tag{33}
$$

According to Equations (32) and (33), V_{BN} can be calculated, and $\{L_2\}$ is calculated by Equation (34).

$$z^{L_2} = r_{J_K, B}$$
$$y^{L_2} = \frac{V_{J_H, B} + V_{BN}}{\|V_{J_H, B} + V_{BN}\|} \tag{34}$$
$$x^{L_2} = y^{L_2} \times z^{L_2}$$

The attitude transformation matrix from $\{L_2\}$ to the coordinate system of sensor B is denoted by Equation (35).

$$R_B^{L_2} = \left[x^{L_2}, \ y^{L_2}, \ z^{L_2} \right]^T \tag{35}$$

Similarly, we can obtain V_{CO}, and the attitude transformation matrix from $\{L_3\}$ to the coordinate system of sensor C is denoted by Equation (36).

$$R_C^{L_3} = \left[x^{L_3}, \ y^{L_3}, \ z^{L_3} \right]^T \tag{36}$$

At initial stage, $\{L_1\}$ and $\{L_2\}$ coincide, and $\{L_3\}$ and $\{L_4\}$ are parallel, i.e.,

$$R_{L_2}^{L_1} = I, \tag{37}$$
$$R_D^{L_4} = R_C^{L_3} R_D^C,$$

where I is a identity matrix of 3 by 3, $R_{L_2}^{L_1}$ is a rotation matrix between $\{L_1\}$ and $\{L_2\}$, and $R_D^{L_4}$ is attitude transformation matrix from $\{L_4\}$ to the coordinate system of sensor D.

4.2. Joint Angles Calculation

According to the ISB standard [24], each joint angle of the lower limbs is the motion of the lower limbs relative to the adjacent upper limb, i.e., the upper leg is relative to the pelvis, the lower leg is relative to the upper leg and the foot is relative to the lower leg. In the 3-DOF of the joint angles, flexion/extension is β, which is the angle of rotation about the z-axes. Abduction/adduction is ϕ, which is the angle of rotation about the x-axes. Internal/external rotation is δ, which is the angle of rotation about the y-axes, according to the Z-X-Y Euler angular rotation order to calculate the joint angles. At t time, the rotation matrix of limb L_i relative to L_{i-1} ($i = 2, \ 3, \ 4$) can be obtained by Equation (38).

$$R_{L_2}^{L_1}(t) = R_B^{L_2} R_B^{O_g}(t)$$
$$R_{L_3}^{L_2}(t) = R_B^{L_2}(R_B^{O_g}(t))^T R_C^B(0) R_C^{O_g}(t)(R_C^{L_3})^T, \tag{38}$$
$$R_{L_4}^{L_3}(t) = R_C^{L_3}(R_C^{O_g}(t))^T R_D^C(0) R_D^{O_g}(t)(R_D^{L_4})^T$$

where $R_{L_i}^{L_{i-1}}$ can be expressed by Equation (39).

$$R_{L_i}^{L_{i-1}} = \begin{bmatrix} -s\delta s\beta s\phi + c\delta c\phi & -s\delta c\beta & s\delta s\beta s\phi + c\delta s\phi \\ c\delta s\beta s\phi + s\delta c\phi & c\delta s\beta & -c\delta s\beta c\phi + s\delta s\phi \\ -c\beta s\phi & s\beta & c\beta c\phi \end{bmatrix}$$
$$= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \tag{39}$$

where c is cos and s is sin. Euler angle can be calculated by Equations (40)–(42).

when $\cos \beta \neq 0$:

$$\beta = \text{atan2} \left(r_{32}, \sqrt{r_{12}^2 + r_{22}^2} \right)$$
$$\delta = \text{atan2} \left(-\frac{r_{12}}{\cos \beta}, \frac{r_{22}}{\cos \beta} \right),$$
$$\phi = \text{atan2} \left(-\frac{r_{31}}{\cos \beta}, \frac{r_{33}}{\cos \beta} \right) \tag{40}$$

when $\beta = 90°$:

$$\beta = 90°$$
$$\delta = 0°$$
$$\phi = \text{atan2} \left(r_{13}, -r_{23} \right) \tag{41}$$

when $\beta = -90°$:

$$\beta = -90°$$
$$\delta = 0°$$
$$\phi = \text{atan2} \left(r_{13}, r_{23} \right) \tag{42}$$

4.3. Single IMU Attitude Fusion

To improve the accuracy of attitude acquisition by single IMU, we need to fuse the attitude rotation matrix in Equation (30). The quaternion-based attitude fusion algorithm can effectively combine the error characteristics of gyroscope and accelerometer, and improve the accuracy of attitude calculation [29]. The expression of a quaternion is defined by Equation (43).

$$q = q_0 + q_1 i + q_2 j + q_3 k, \tag{43}$$

where i, j, k is an imaginary unit, q_0, q_1, q_2, q_3 is a real number, and each quaternion is a linear combination of 1, i, j and k.

(1) Quaternion initialization

At time t, the quaternion of attitude change is $q_t = [q_0, q_1, q_2, q_3]^T$, the attitude calculation error is $\xi_t = [\xi_x, \xi_y, \xi_z]^T$. At the initial stage, q_0 and ξ_0 are defined by Equation (44).

$$q_0 = [1, \ 0, \ 0, \ 0]^T,$$
$$\xi_0 = [0, \ 0, \ 0]^T, \tag{44}$$

(2) Correction of angular velocity error

Based on the definition of cosine matrix and Euler angles in [30], the gravity vector of the global coordinate system can be rotated to the sensor coordinate system by Equation (45).

$$g_t^{O_s} = R_{O_s}^{O_g}(t) \cdot g_t^{O_g}$$
$$= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_3 q_0) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \tag{45}$$

where the rotation matrix of O_s relative to O_g by quaternion is denoted as $R_{O_s}^{O_g}(t)$. The acceleration of gravity is denoted as $g_t^{O_g}$. The acceleration of gravity after rotation from O_g to O_s is denoted as $g_t^{O_s} = \left[g_x^{O_s}, g_y^{O_s}, g_z^{O_s} \right]^T$. Equation (45) is simplified by Equation (46).

$$g_x^{O_s} = 2(q_1 q_3 - q_0 q_2),$$
$$g_y^{O_s} = 2(q_2 q_3 + q_0 q_1), \tag{46}$$
$$g_z^{O_s} = q_0^2 - q_1^2 - q_2^2 + q_3^2.$$

In the process of IMU attitude rotation, the gravity vector measured by the accelerometer is $\boldsymbol{a}_t^{O_s} = [a_x^{O_s}, a_y^{O_s}, a_z^{O_s}]^T$, and the gravity vector calculated by the attitude integrated by the gyroscope is $\boldsymbol{g}_t^{O_s} = \left[g_x^{O_s}, g_y^{O_s}, g_z^{O_s}\right]^T$. The error vector between them is $\boldsymbol{\xi}_t = \left[\xi_x, \xi_y, \xi_z\right]^T$, which is the error between the attitude integrated by the gyroscope and the attitude measured by the accelerometer. It can be expressed by cross product, and ξ_t is defined by Equation (47).

$$\xi_x = a_y^{O_s} \cdot g_z^{O_s} - a_z^{O_s} \cdot a_y^{O_s},$$
$$\xi_y = a_z^{O_s} \cdot g_x^{O_s} - a_x^{O_s} \cdot a_z^{O_s}, \tag{47}$$
$$\xi_z = a_x^{O_s} \cdot g_y^{O_s} - a_y^{O_s} \cdot a_x^{O_s}.$$

(3) Data fusion

The cross product error is adjusted by proportional-integral (PI) controller [31] to correct the bias of the gyroscope. By adjusting the two parameters λ_p and λ_i, the speed of the accelerometer to correct the integral attitude of the gyroscope can be controlled, where, λ_p is the proportional adjustment coefficient, which is used to control the speed of the error converges to the accelerometer. Once there is a deviation in the system, the proportional adjustment will immediately produce an adjustment effect to reduce the error. λ_i is the integral adjustment coefficient, which is used to control the convergence speed of gyro bias, so as to eliminate the steady-state error and improve the accuracy of the system.

(1) At time t, integrate the cross product error by Equation (48).

$$\xi_x^+ = \xi_x^- + \xi_x \cdot \lambda_i \cdot \frac{1}{2}\Delta t,$$
$$\xi_y^+ = \xi_y^- + \xi_y \cdot \lambda_i \cdot \frac{1}{2}\Delta t, \tag{48}$$
$$\xi_z^+ = \xi_z^- + \xi_z \cdot \lambda_i \cdot \frac{1}{2}\Delta t,$$

(2) The fused gyro measurements are defined by Equation (49).

$$w_x^{O_s^+} = w_x^{O_s^-} + \lambda_p \cdot \xi_x + \xi_x^+,$$
$$w_y^{O_s^+} = w_y^{O_s^-} + \lambda_p \cdot \xi_y + \xi_x^+, \tag{49}$$
$$w_z^{O_s^+} = w_z^{O_s^-} + \lambda_p \cdot \xi_z + \xi_x^+,$$

where $\lambda_p = 2\Delta t$. The sampling frequency is denoted as Δt. The angular velocity measured by gyroscopes in IMUs coordinate system are denoted as $\boldsymbol{w}_t^{O_s} = \left[w_x^{O_s}, w_y^{O_s}, w_z^{O_s}\right]^T$. "-" is prior-estimted, and "+" is post-estimted. Since the parameter value of PI controller needs to be dynamically adjusted according to different experimental requirements, in this work, the value of λ_i cannot be too large. As shown in Figure 6, after several experimental parameter adjustments, when λ_i is greater than $0.2\Delta t$, the ξ defined by Equation (48) increases gradually. Therefore, any value less than 0.2 is acceptable, and we set $\lambda_i = 0.1\Delta t$.
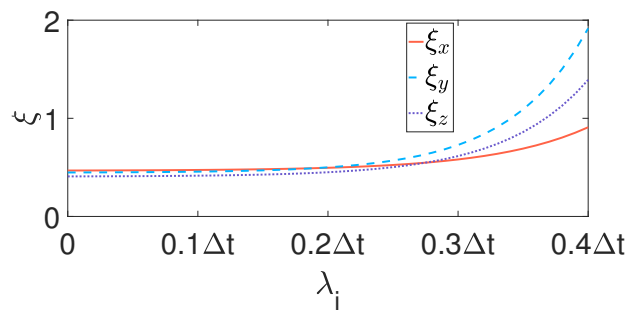
Figure 6. When the λ_i takes different values, the convergence curve of ξ.

(4) Update quaternion

$$
\begin{bmatrix} q_0(t+\triangle t) \\ q_1(t+\triangle t) \\ q_2(t+\triangle t) \\ q_3(t+\triangle t) \end{bmatrix} = \begin{bmatrix} q_0(t) \\ q_1(t) \\ q_2(t) \\ q_3(t) \end{bmatrix} \begin{bmatrix} 1 & -\frac{w_x^{O_s}}{2}\Delta t & -\frac{w_y^{O_s}}{2}\Delta t & -\frac{w_z^{O_s}}{2}\Delta t \\ \frac{w_x^{O_s}}{2}\Delta t & 1 & \frac{w_z^{O_s}}{2}\Delta t & -\frac{w_y^{O_s}}{2}\Delta t \\ \frac{w_y^{O_s}}{2}\Delta t & -\frac{w_z^{O_s}}{2}\Delta t & 1 & \frac{w_x^{O_s}}{2}\Delta t \\ \frac{w_z^{O_s}}{2}\Delta t & \frac{w_y^{O_s}}{2}\Delta t & -\frac{w_x^{O_s}}{2}\Delta t & 1 \end{bmatrix}
\tag{50}
$$

(5) Convert the updated quaternions into matrix forms by $R_{O_s}^{O_g}$ of Equation (45), and the attitude transformation matrix of each IMU relative to the global coordinate system can be obtained.

5. Experimental Analysis

5.1. Measurement Equipment

In Figure 7, four IMUs (Yost Labs, USA) with red marked points were bound to the limb with medical tape. Each IMU included a tri-axial accelerometer in the range ± 16 g and a tri-axial gyroscope in range ± 2000 deg/s. The four IMUs are bound to the waist, upper leg, lower leg, and foot, respectively, and the IMUs are connected by a 3.5 mm retractable spring cable. All IMUs record data synchronously during motion capture, and the sampling frequency is 100 Hz, which is transmitted to the computer wirelessly. The white marked point is a wireless motion capture device named Perception Neuron Pro (Noitom, CN), which is used as a reference system and worn on the limbs simultaneously with IMUs for synchronous capture. The motion capture device is only used to verify the performance of the calibration algorithm.

| (a) | (b) |
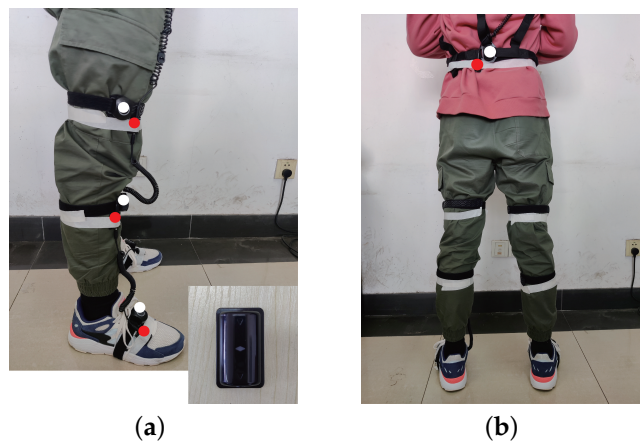
Figure 7. The motion capture test equipment. (**a**) he IMUs on the leg; (**b**) the IMUs on the waist.

Considering that different gait characteristics of different people may affect the experimental results and avoid the randomness of the experimental results, in the data acquisition

phase, the data of the accelerometer and gyroscope are acquired by three subjects, including one female at a height of 165 cm (subject 1) and two males at heights of 175 cm (subject 2) and 180 cm (subject 3), respectively. Among them, the data of motion capture device are only used for reference, which is different from the data collected by IMUs and they do not affect each other. As shown in Figure 8, the IMUs are mounted at position 1 and position 2, respectively, for two experiments, and the data of the accelerometer and gyroscope are collected at two different positions, respectively. To study the randomness and accuracy of the calibration algorithm, the binding positions of IMUs do not coincide with the motion capture device, i.e., the positions of IMUs is displaced. Before the formal motion capture, the subjects need to stand still for 10 s to obtain the attitude rotation matrix between adjacent IMUs in the initial state. During motion capture, the subjects walked for 15 s. To avoid the deviation of the walking mode from normal, the subjects are not informed that the walking data would be used for calibration. The data collected by IMUs were substituted into the calibration algorithm to calculate the angle changes of the hip, knee, and ankle during walking and compared with the reference value of motion capture equipment.

| (a) | (b) | (c) | (d) |
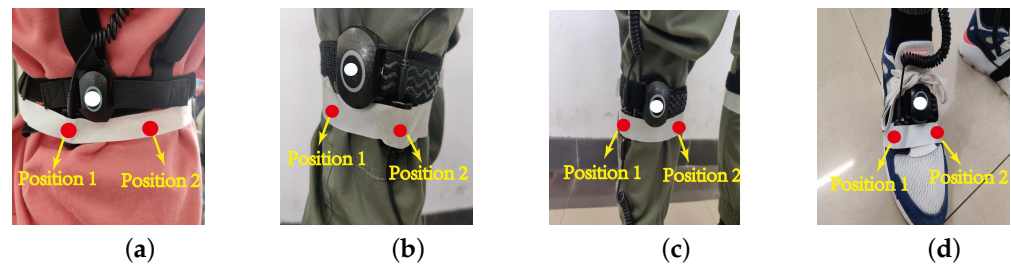
Figure 8. The IMUs are mounted in position 1 and position 2 for the experiment, respectively. (**a**) The waist; (**b**) the upper leg; (**c**) the lower leg; (**d**) the foot.

5.2. Data Analysis

To analyze the accuracy of the algorithm, we compared the deviation between the three algorithms and the reference value. The angle calculated by the GN, DWPSO, and GWO are estimated values, and the angle calculated by motion capture equipment are reference values. The Root Mean Squared Error (RMSE) between the estimate values and the reference value for each DOF is calculated by Equation (51).

$$RMSE = \sqrt{\frac{1}{t} \sum_{m=1}^{t} \left(\boldsymbol{H}_{IMU,m} - \boldsymbol{H}_{MCS,m} \right)^2}, \tag{51}$$

where $\boldsymbol{H}_{IMU}$ is the estimation values from IMUs, $\boldsymbol{H}_{MCS}$ is the angle reference value calculated by motion capture equipment named Perception Neuron Pro.

5.3. Results and Analysis

Figures 9–11 show the RMSE comparison of three algorithms when IMUs on three subjects were bound in two positions. It shows that the three algorithms achieve position calibration at two positions, respectively. No matter where it is, the RMSE of the DWPSO is the lowest of the three algorithms, which is closer to the reference value. It is worth mentioning that the reference value will change when IMUs change the position. When the IMUs were placed in the second position, the accuracy of the three algorithms were ranked the same. When the GWO is used for position calibration, the initial population is easy to be unevenly distributed and lacks global communication, resulting in the final solution being easy to fall into local optimization. In the DWPSO algorithm, we introduce dynamic weight to control the speed of the initial population and improve the accuracy of the algorithm. Therefore, the calibration performance of the GWO is lower than DWPSO. However, the introduction of dynamic weight increases the complexity of the PSO algorithm and reduces the efficiency of DWPSO.
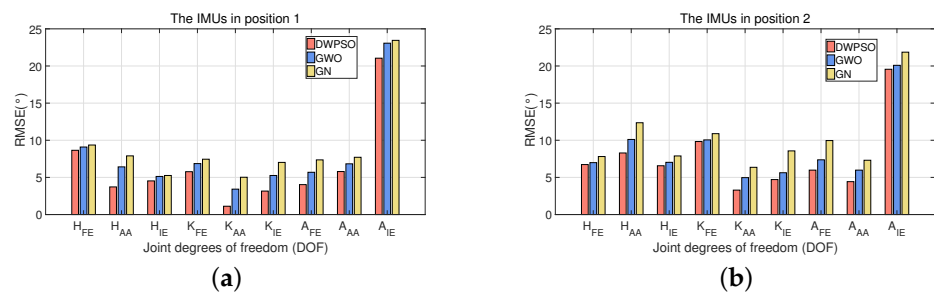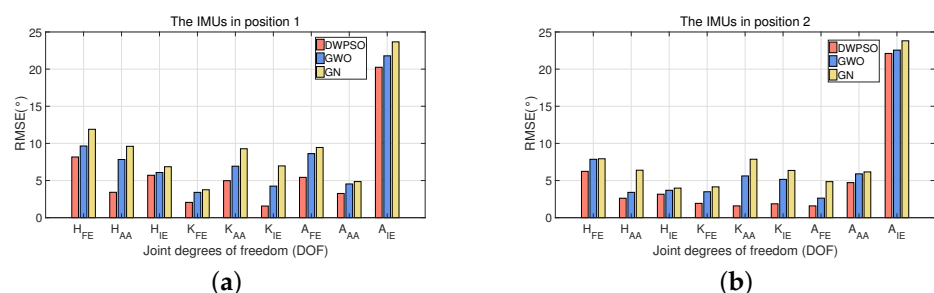
Figure 9. The RMSE (°) comparison of three algorithms when IMUs on subject 1 were bound in two positions. (**a**) The IMUs in position 1; (**b**) the IMUs in position 2.
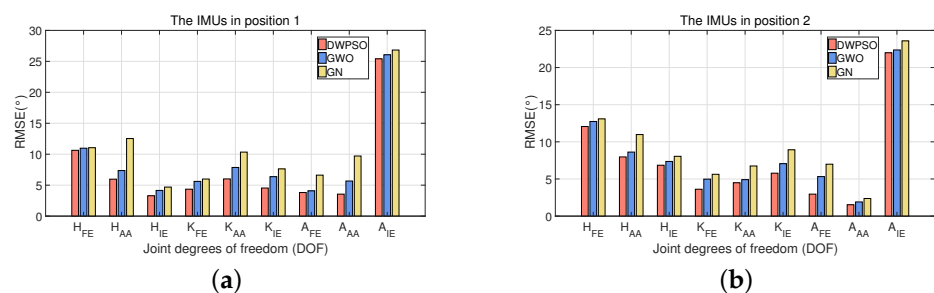
Figure 10. The RMSE (°) comparison of three algorithms when IMUs on subject 2 were bound in two positions. (**a**) The IMUs in position 1; (**b**) the IMUs in position 2.

Figure 11. The RMSE (°) comparison of three algorithms when IMUs on subject 3 were bound in two positions. (**a**) The IMUs in position 1; (**b**) the IMUs in position 2.

Table 1 shows the average and standard deviation (SD) of 15 computation times of three algorithms, and all algorithms are completed on the same computer. As shown in Table 1, the GWO uses the shortest average computation times, followed by the DWPSO, and the GN takes the longest. When a high calibration accuracy and fast algorithm efficiency are required, the GWO can be used for calibration. However, the SD value of the GWO is the highest, indicating that the algorithm is less stable than DWPSO and GN, which may reduce the efficiency. The DWPSO algorithm is relatively stable, and the optimization performance is better than the other two algorithms. When there is no requirement for speed, the DWPSO may be the best choice.

Table 1. Average and standard deviation (SD) of 15 computation times of the DWPSO, GWO, and GN.

| Algorithm Type	Average (s)	SD
DWPSO	1076.1	2.01
GWO	576.3	3.76
GN	1556.4	2.98

Combined with the analyses in Table 2 and Figures 9–11, although the heights and sexes of the subjects are different, the variation range of the results of each subject is roughly the same, and the performance of the calibration algorithm is also the same. This is because the three calibration algorithms are carried out under the same joint constraints and the joint constraints of each subject are the same, which will not be affected by the different gait characteristics of the subjects. Therefore, subject 1 is selected as the sample for analysis. Figure 12 shows the variation of the joint angle of IMUs in position 1 for 5 s. It shows that the angle variation waveform of each joint is consistent with the reference value, only the up and down translation is produced in terms of amplitude. It indicates that the offset error of IMUs position is fixed and will not change over time.

Table 2. The RMSE (°) comparison of three algorithms when IMUs on three subjects were bound in position 1.

|	Subject 1			Subject 2			Subject 3		
	DWPSO	GWO	GN	DWPSO	GWO	GN	DWPSO	GWO	GN
H_{FE}	8.65	9.09	9.36	10.63	10.97	11.05	8.17	9.65	11.90
H_{AA}	3.72	6.42	7.90	5.97	7.36	12.53	3.42	7.83	9.61
H_{IE}	4.53	5.13	5.26	3.29	4.15	4.69	5.71	6.08	6.86
K_{FE}	5.77	6.86	7.45	4.35	5.61	5.99	2.06	3.41	3.76
K_{AA}	1.12	3.42	5.02	6.01	7.86	10.34	4.98	6.93	9.28
K_{IE}	3.16	5.26	7.02	4.54	6.37	7.63	1.57	4.25	6.97
A_{FE}	4.03	5.69	7.36	3.81	4.08	6.62	5.43	8.62	9.45
A_{AA}	5.76	6.83	7.71	3.55	5.67	9.71	3.26	4.54	4.86
A_{IE}	21.05	23.07	23.45	25.41	26.06	26.83	20.25	21.79	23.67

Table 2 shows the test results of the RMSE when IMUs on three subjects were bound in position 1. In H_{FE}, H_{IE}, K_{FE} and A_{IE}, the performances of the three algorithms are close to each other. In H_{AA}, K_{AA}, K_{IE}, A_{FE} and A_{AA}, the calibration performances of the DWPSO and GWO are better than GN. A possible explanation is that under this DOF, the variation of the joint is not significant, which will affect the calculation of the Jacobian matrix and the accuracy of the calibration. The DWPSO and GWO do not consider the Jacobian matrix, and their accuracy is significantly higher than GN. Additionally, the results in Figure 12 show that when the joint angle is around 0°, the values of the DWPSO and GWO are closer to the reference value than GN, e.g., K_{AA} or A_{FE}. This is because the DOF is not the main activity of the joint, and will also affect the performance of the GN algorithm.

To more intuitively evaluate the consistency between the results of the three calibration algorithms and the reference, we selected the angle values of H_{AA} and K_{IE} in Figure 12 as samples and plot the Bland–Altman diagram for analysis. As shown in Figure 13, the x-axes are the average of each individual between the reference value and estimated value, the y-axes are the difference of each individual between the reference and the estimated. The two red lines in the figure are the upper and lower limits of the 95% consistency interval, the purple dotted line indicates that the average value of the difference is 0, and the green line is the average value of the difference between the reference value and the estimated value in each individual. The closer the green line is to the purple dotted line, the higher the consistency between the reference value and the estimated value. As shown in Figure 13a, in H_{AA}, the average difference value of the DWPSO is the closest to 0, and the consistency with the reference value is the highest, the GWO is the second, and the GN is the lowest. As shown in Figure 13b, the consistency analysis of K_{IE} is also the highest in the DWPSO. These results are consistent with the curve results in Figure 12. Additionally, most of the results in Figure 13 are within the confidence interval, which explains why the waveforms of the estimated value and the reference value are similar in Figure 12.
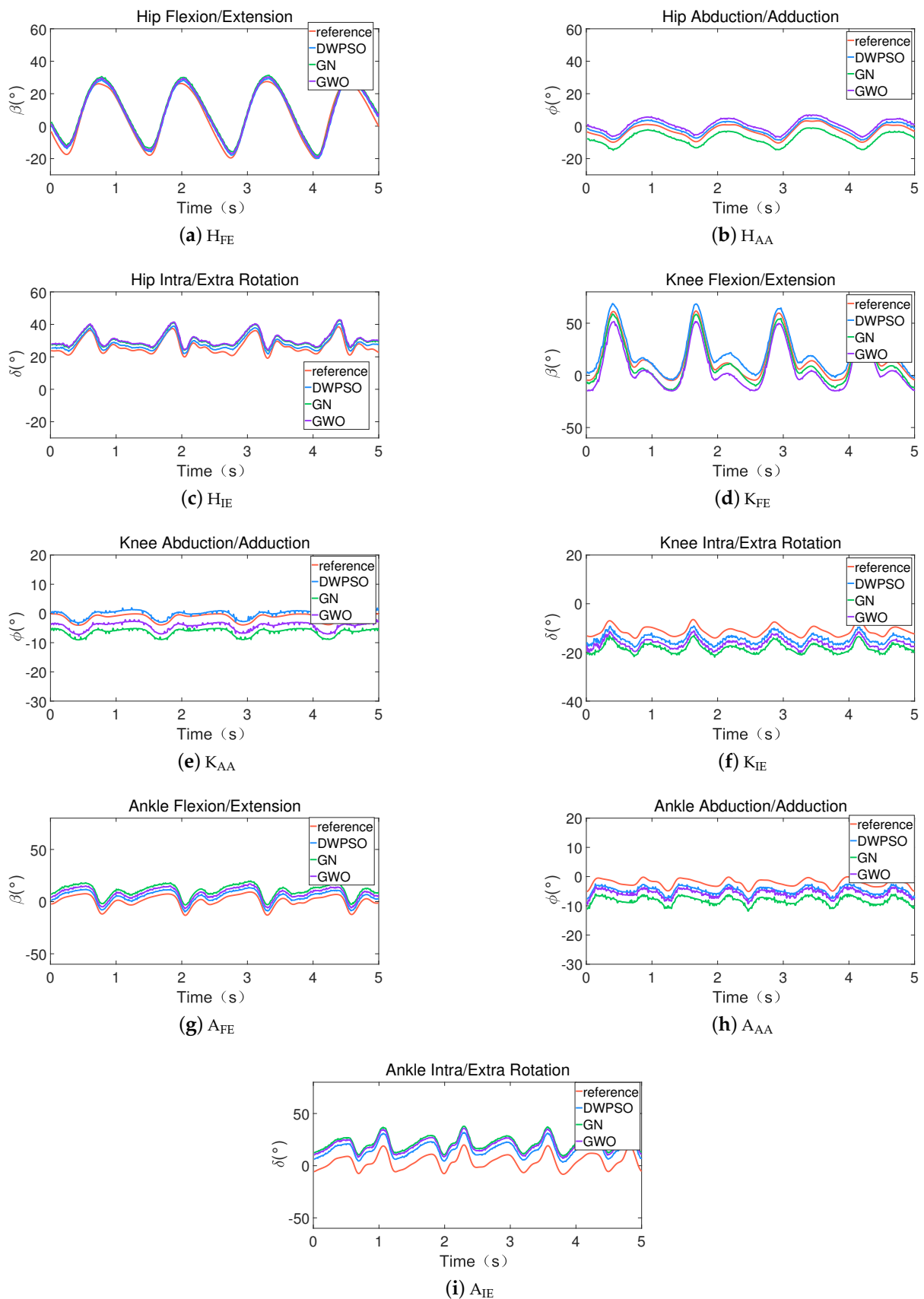
Figure 12. The variation of the joint DOF when IMUs on subject 1 were bound in position 1. (**a**–**c**) The hip joint; (**d**–**f**) the knee joint; (**g**–**i**) the ankle joint.

(a)

(b)
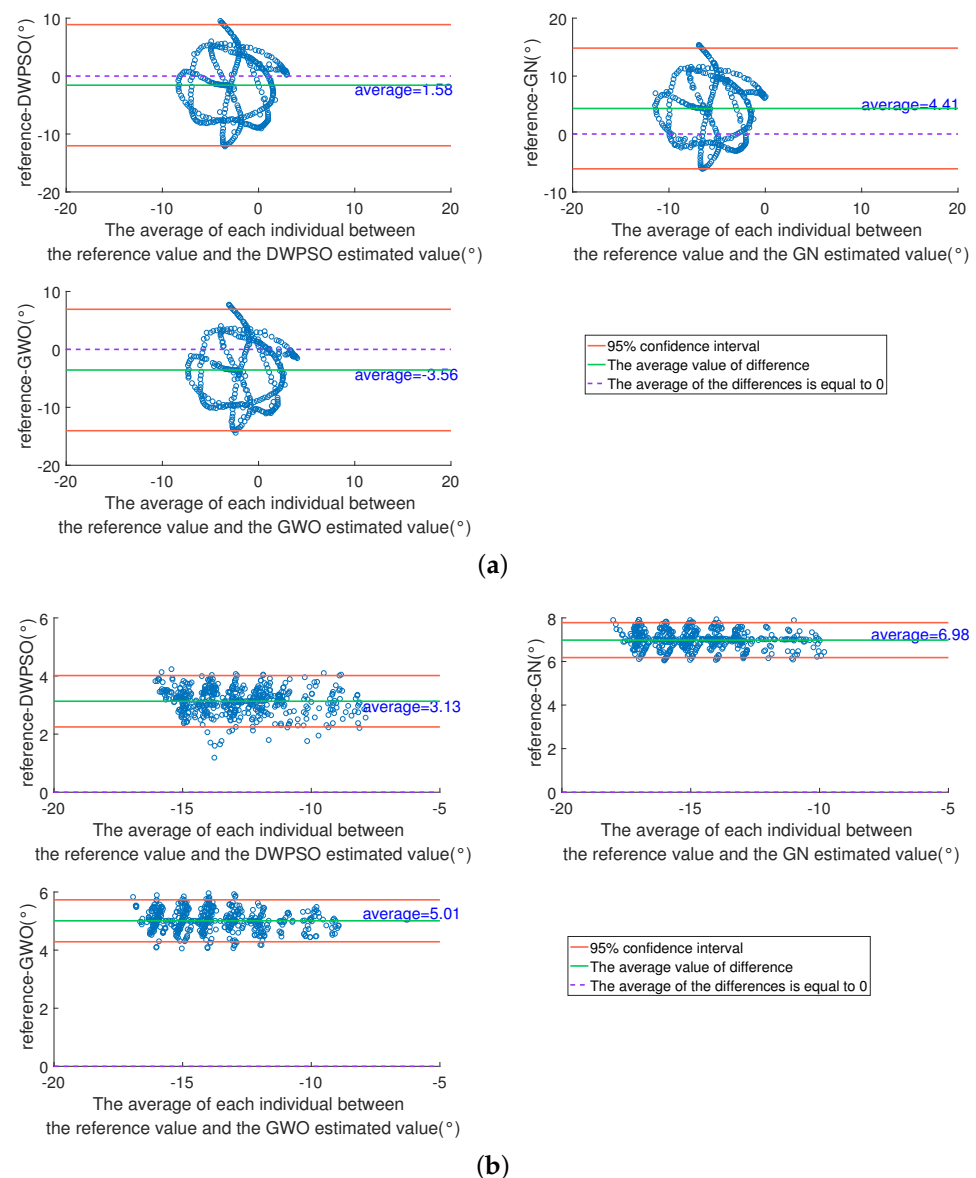
Figure 13. The Bland-Altman consistency analysis of estimated and reference values of three algorithms when IMUs in position 1. (**a**) H_{AA}; (**b**) K_{IE}.

Through the above analysis, it shows that in the IMUs position calibration of three joints of human lower limbs, the three algorithms have achieved good calibration results, and the calibration accuracy of the population algorithm is better than GN. When the joint changes sufficiently in a certain DOF, the results of the three algorithms are close. When the joint changes are insufficient, the calibration accuracy of the population algorithm is obviously better than GN. For two different population algorithms the DWPSO and GWO, different choices can be made according to practical applications.

6. Conclusions

In this work, we introduce the DWPSO, GWO, and GN algorithms to realize the dynamic calibration of IMUs' positions based on human lower limb joint constraints. The performance of the algorithm is evaluated by gait experiments. The results show that the three algorithms have achieved IMU position calibration and are suitable for estimating the angles of the hip, knee, and ankle of humans during free walking. The simulation results show that the DWPSO has the best calibration performance, followed by the GWO and GN. When the joint rotation is sufficient or the joint is in the main motion, the performances of

the three algorithms are close. When the joint rotation is insufficient, the performances of the DWPSO and the GWO are significantly better than the GN.

At present, our work has achieved an IMU position calibration of human lower limbs. However, when applied for a whole-body calibration, a large amount of data may cause the decline of the searchability of the DWPSO and GWO. In future work, we need to conduct further experiments.

Another route of future work is that when the offset error of IMUs position drifts slowly over time in the short term, an accelerometer and gyroscope can be combined to estimate the joint axis of the knee joint, and further improve the position calibration accuracy.

References

1. Muro-De-La-Herran, A.; Garcia-Zapirain, B.; Mendez-Zorrilla, A. Gait analysis methods: An overview of wearable and non-wearable systems, highlighting clinical applications. *Sensors* **2014**, *14*, 3362–3394. [CrossRef] [PubMed]
2. Tao, W.; Tao, L.; Zheng, R.; Feng, H. Gait Analysis Using Wearable Sensors. *Sensors* **2012**, *12*, 2255–2283. [CrossRef] [PubMed]
3. Blake, A.; Grundy, C. Evaluation of motion capture systems for Golf Swings: Optical vs. gyroscopic. In Proceedings of the ITI 2008—30th International Conference on Information Technology Interfaces, Cavtat, Croatia, 23–26 June 2008; pp. 409–414.
4. Ghasemzadeh, H.; Jafari, R. Coordination Analysis of Human Movements with Body Sensor Networks: A Signal Processing Model to Evaluate Baseball Swings. *IEEE Sens. J.* **2011**, *11*, 603–610. [CrossRef]
5. Walsh, M.; Barton, J.; O'Flynn, B.; O'Mathuna, C.; Tyndyk, M. Capturing the overarm throw in darts employing wireless inertial measurement. In Proceedings of the SENSORS, 2011 IEEE, Limerick, Ireland, 28–31 October 2011; pp. 1441–1444.
6. Valentina, C.; Elena, B.; Fantozz, S.; Vannozzi, G. Trends Supporting the In-Field Use of Wearable Inertial Sensors for Sport Performance Evaluation: A Systematic Review. *Sensors* **2018**, *18*, 873.
7. Salarian, A.; Russmann, H.; Vingerhoets, F.; Dehollain, C.; Aminian, K. Gait assessment in Parkinson's disease: Toward an ambulatory system for long-term monitoring. *IEEE Trans. Biomed. Eng.* **2004**, *51*, 1434–1443. [CrossRef] [PubMed]
8. Anwary, A.; Yu, H.; Vassallo, M. An Automatic Gait Feature Extraction Method for Identifying Gait Asymmetry Using Wearable Sensors. *Sensors* **2018**, *18*, 676. [CrossRef] [PubMed]
9. Palermo, E.; Rossi, S.; Marini, F.; Cappa, P. Experimental evaluation of accuracy and repeatability of a novel body-to-sensor calibration procedure for inertial sensor-based gait analysis. *Measurement* **2014**, *52*, 145–155. [CrossRef]
10. Roetenberg, D.; Luinge, H.; Slycke, P. Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors. *Xsens Motion Technol. BV* **2009**, *3*, 1–9.
11. Robert-Lachaine, X.; Mecheri, H.; Larue, C.; Plamondon, A. Validation of inertial measurement units with an optoelectronic system for whole-body motion analysis. *Med. Biol. Eng. Comput.* **2016**, *55*, 1–11. [CrossRef] [PubMed]
12. Favre, J.; Aissaoui, R.; Jolles, B.M.; Guise, J.; Aminian, K. Functional calibration procedure for 3D knee joint angle description using inertial sensors. *J. Biomech.* **2009**, *42*, 2330–2335. [CrossRef] [PubMed]
13. Cutti, A.G.; Ferrari, A.; Garofalo, P.; Raggi, M.; Cappello, A.; Ferrari, A. 'Outwalk': A protocol for clinical gait analysis based on inertial and magnetic sensors. *Med. Biol. Eng. Comput.* **2010**, *48*, 17. [CrossRef] [PubMed]
14. Nazarahari, M.; Rouhani, H. Semi-Automatic Sensor-to-Body Calibration of Inertial Sensors on Lower Limb Using Gait Recording. *IEEE Sens. J.* **2019**, *19*, 12465–12474. [CrossRef]
15. Elias, L.S.V.; Elias, A.; Rocon, E.; Freire, T.; Frizera, A. An IMU-to-Body Alignment Method Applied to Human Gait Analysis. *Sensors* **2016**, *16*, 2090. [CrossRef] [PubMed]

16. Cheng, P.; Oelmann, B. Joint-Angle Measurement Using Accelerometers and Gyroscopes-A Survey. *IEEE Trans. Instrum. Meas.* **2010**, *59*, 404–414. [CrossRef]
17. Miezal, M.; Bleser, G.; Schmitz, N.; Stricker, D. A Generic Approach to Inertial Tracking of Arbitrary Kinematic Chains. In Proceedings of the 8th International Conference on Body Area Networks, Boston, MA, USA, 30 September–2 October 2013; Number 4, pp. 189–192.
18. Seel, T.; Schauer, T.; Raisch, J. Joint axis and position estimation from inertial measurement data by exploiting kinematic constraints. In Proceedings of the 2012 IEEE International Conference on Control Applications, Dubrovnik, Croatia, 3–5 October 2012; pp. 45–49.
19. Seel, T.; Raisch, J.; Schauer, T. IMU-Based Joint Angle Measurement for Gait Analysis. *Sensors* **2014**, *14*, 6891–6909. [CrossRef] [PubMed]
20. Nerino, R.; Contin, L.; Gonçalves da Silva Pinto, W.J.; Massazza, G.; Actis, M.; Capacchione, P.; Chimienti, A.; Pettiti, G. A BSN Based Service for Post-Surgical Knee Rehabilitation at Home. In Proceedings of the 8th International Conference on Body Area Networks; ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Boston, MA, USA, 30 September–2 October 2013; pp. 401–407.
21. O'Donovan, K.J.; Kamnik, R.; O'Keeffe, D.T.; Lyons, G.M. An inertial and magnetic sensor based technique for joint angle measurement. *J. Biomech.* **2007**, *40*, 2604–2611. [CrossRef] [PubMed]
22. Shi, Y.; Eberhart, R. A modified particle swarm optimizer. In Proceedings of the 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA, 4–9 May 1998; pp. 69–73.
23. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
24. Wu, G.; Siegler, S.; Allard, P.; Kirtley, C.; Leardini, A.; Rosenbaum, D.; Whittle, M.; DâĂŹLima, D.D.; Cristofolini, L.; Witte, H.; et al. ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion—Part I: Ankle, hip, and spine. *J. Biomech.* **2002**, *35*, 543–548. [CrossRef]
25. Cappozzo, A.; Catani, F.; Leardini, A.; Benedetti, M.; Della Croce, U. Position and orientation in space of bones during movement: Experimental artefacts. *Clin. Biomech.* **1996**, *11*, 90–100. [CrossRef]
26. Craig, J.J. *Introduction to Robotics: Mechanics and Control*; Pearson: New York City, NY, USA, 1986.
27. Gao, Y.; Song, D. A New Improved Genetic Algorithms and its Property Analysis. In Proceedings of the 2009 Third International Conference on Genetic and Evolutionary Computing, Guilin, China, 14–17 October 2009; pp. 73–76.
28. Paul, R.P. *Robot Manipulators: Mathematics, Programming and Control*; The MIT Press: Cambridge, MA, USA, 1981.
29. Zhou, Y.; Zhang, H.; Zhang, T.; Wang, L.; Wei, R.; Luan, M.; Liu, H.; Shi, J. A fusion attitude determination method based on quaternion for MEMS gyro/accelerometer/magnetometer. In Proceedings of the 2013 25th Chinese Control and Decision Conference (CCDC), Guiyang, China, 25–27 May 2013; pp. 3228–3232.
30. Premerlani, W.; Bizard, P. *Direction Cosine Matrix IMU: Theory*; Diy Drone: USA, 2009; pp. 13–15; Available online: https://www.researchgate.net/publication/265755808_Direction_Cosine_Matrix_IMU_Theory (accessed on 19 September 2021).
31. Goodwin, G.C.; Graebe, S.F.; Salgado, M.E. *Control System Design*; Prentice Hall New: Hoboken, NJ, USA, 2001; Volume 240.