A Temperature Compensation Method for a Six-Axis Force/Torque Sensor Utilizing Ensemble hWOA-LSSVM Based on Improved Trimmed Bagging

Xuhao Li 1,2, Lifu Gao 2,3, Huibin Cao 2,*, Yuxiang Sun 2, Man Jiang 2 and Yue Zhang 1,2

Temperature Compensation Method for a Six-Axis Force/Torque Sensor Utilizing Ensemble hWOA-LSSVM Based on Improved Trimmed Bagging. *Correspondence: hbcao@iim.ac.cn; Tel.: +86-0551-65592409

Abstract: The performance of a six-axis force/torque sensor (F/T sensor) severely decreased when working in an extreme environment due to its sensitivity to ambient temperature. This paper puts forward an ensemble temperature compensation method based on the whale optimization algorithm (WOA) tuning the least-square support vector machine (LSSVM) and trimmed bagging. To be specific, the stimulated annealing algorithm (SA) was hybridized to the WOA to solve the local entrapment problem, and an adaptive trimming strategy is proposed to obtain the optimal trim portion for the trimmed bagging. In addition, inverse quote error (invQE) and cross-validation are employed to estimate the fitness better in training process. The maximum absolute measurement error caused by temperature decreased from 3.34% to $3.9 \times 10^{-3}$% of full scale after being compensated by the proposed method. The analyses of experiments illustrate the ensemble hWOA-LSSVM based on improved trimmed bagging improves the precision and stability of F/T sensors and possesses the strengths of local search ability and better adaptability.

Keywords: temperature compensation; six-axis force/torque sensor; whale optimization algorithm; bagging; least square support vector machine

1. Introduction

Force/torque sensors are widely applied in manipulators to obtain feedback on force and torque, which can help researchers achieve close-loop control. When working in extreme conditions, for instance, in space [1] or in deep-sea [2,3] applications, F/T sensors will suffer severe characteristic drift due to the wide range of ambient temperature. The characteristic drifts will cause bias between measured values and real values, consequently decreasing the precision of measurements. Thus, the effects of temperature should be compensated for to alleviate and even eliminate the characteristic drifts of F/T sensors.

The impacts of temperature on strain gauge sensors are omnifarious. The thermal effects of strain gauges and the different dilatation coefficients between strain gauges and elastic bodies are two major factors which cause characteristic drifts. Hardware and software methods are two ways to compensate for the impacts of temperature. Compensating methods focusing on hardware aim to eliminate the temperature’s impact on the regulating circuit [4–6]. However, compensating on the circuit is usually costly and lacks flexibility. More importantly, it cannot meet the requirement of precision. Thus, hardware compensating methods are used as a kind of auxiliary means in practice [7]. Software compensating methods aim to build inverse models between ambient temperature and thermal output. Sensors compensate for the characteristic drift according to the thermal output obtained by inverse models. Software compensation methods are powerful, flexible,
and easily applied to various sensors. In view of these advantages, scholars proposed lots of software compensating methods, such as least square method (LSM), support vector machine (SVM) [8,9], and all kinds of artificial neutral networks (ANN) [10–12]. LSM is convenient and intuitive, so researchers commonly apply it to solve linear and low rank non-linear fitting problems. However, the compensating performances of LSM are not as good as expected because rank non-linear error is usually involved in characteristic drifts. In practice, LSM is still applied to temperature compensation due to its simplicity when the calculating capacity is limited by the process unit size. The applications of ANN have been more and more extensive in recent years due to their prominent nonlinear fitting ability, and scholars proposed massive compensation methods based on ANN [13–15]. However, compensating methods based on ANN suffer from two issues: low convergence rate and complex network structures. SVM derives from statistical learning theory and structural risk minimization (SRM). Convex optimization algorithms, such as quadratic programs, are adopted to solve SVM. Suykens J.A.K. and Vandewalle J. reformulated standard SVM and proposed the least-square support vector machine (LSSVM) [16]. LSSVM inherits remarkable abilities for solving nonlinear and high rank regression problems from SVM and provides some improvements, including adopting a binary norm in the objective function and converting the optimization problem to solve the linear Karush–Kuhn–Tucker (KKT) condition. Zhang et al. [17] proposed a LSSVM error compensation model to accurately estimate the state of health (SOH) of lithium-ion batteries. Tan et al. [18] predicted the thermal error of machine tool spindles using the segment fusion least-square support vector machine (SF-LSSVM) and improved the prediction precision by up to 51%.

Though LSSVM is powerful in nonlinear fitting, its performance is significantly influenced by the choice of its parameters. Obtaining parameters of LSSVM by blind search is a costly task; therefore, various optimization algorithms are adopted by scholars, such as the Cuckoo search algorithm (CSA), particle swarm optimization (PSO), and the sparrow search algorithm (SSA) [19–21]. The whale optimization algorithm (WOA) is a meta-heuristic optimization algorithm which is inspired by the bubble-net hunting behavior of humpback whales [22]. The WOA obtains competitive performances in global exploration, has high flexibility compared with other optimization algorithms, and owns a simple structure and few parameters, which make it easy to be adopted in many fields. Although the WOA can efficiently locate the latent positions of the solution in the exploration phase, it suffers from a low convergence rate in the exploitation phase and has a tendency for being trapped in local optima [23,24]. Several measures are taken to offset these shortcomings of WOA, and embedding a high-efficiency local algorithm into WOA is gaining widespread traction. Tong [25] designed a hybrid algorithm framework for WOA, and verified the framework by embedding differential evolution (DE) and the backtracking search optimization algorithm (BSA) into WOA. Nadim et al. [26] hybridized WOA and DE to solve multi-objective virtual machine scheduling in cloud computing. Based on the aforesaid idea, this paper enhances the exploitative ability of the WOA by adopting the stimulated annealing algorithm (SA).

Ensemble method is a series of machine learning methods; boosting and bagging are representatives. They follow the "sampling–training–combining" workflow [27]. Ensemble methods are in the spotlight of the machine learning field for their remarkable improvements in accuracy compared to single learners. Bagging adopts bootstrap sampling to obtain training data and combines the output of each learner by the most common strategy, such as voting for classification and averaging for regression [28]. Bagging can reduce the variance of base learners, especially when those base learners are unstable. In view of improving the performance of any classifier, Joossens et al. proposed trimmed bagging, which only averages outputs from the best base learners, rather than all of them, when applying the bagging method. LSSVM is a stable learning method, and the turbulence of training set can have a subtle impact on it. The standard bagging algorithm works not well, and LSSVM acts as the base learner; therefore, trimmed bagging is adopted in the proposed method.
In this study, an ensemble temperature compensation approach is proposed which adopts LSSVM as the base learner and trimmed bagging as the ensemble framework. To achieve the optimal performance of LSSVM, the hybrid whale optimization algorithm (hWOA) was used to configure parameters in LSSVM, which introduced the simulated annealing algorithm (SA) to WOA. In addition, inverse quoted error (inv QE) was taken to weight aggregate outputs of base learners for its extensive usage in practice. Finally, a temperature compensation model was established by ensemble hWOA-LSSVM based on trimmed bagging.

2. Basic Algorithm
2.1. Least-Square Support Vector Machine (LSSVM)

The LSSVM is used vastly in many fields, such as classification, pattern recognition, and regression. The replacement of non-equality constraints by equality constrains in formulation is the main distinction between LSSVM and standard SVM.

Given a training set \( D_t = \{ x_i, y_i \}_{i=1}^{N} \), where \( x_i \in \mathbb{R}^n \) is the input vector and \( y_i \in \mathbb{R} \) is the output vector, the LSSVM can be described as:

\[
f(x) = w^T \varphi(w) + b
\]  

where

\[ w = (w_1; w_2; \cdots; w_d) \]

is a normal vector of the hyperplane,
\[ b \]

is a bias that decides the distance between the hyperplane and origin point,
\[ \varphi( \cdot ) \]

is a mapping function which maps input data into a higher dimension space.

The aim of a regression task is to minimize the difference between predicted output \( f(x_i) \) and real output \( y_i \). Hence, according to the theory of regulation, the optimization object of LSSVM can be written as

\[
\begin{align*}
\min_{w,b,c} & \quad \frac{1}{2} w^T w + \frac{1}{2} \gamma \sum_{i=1}^{N} c_i^2, \\
\text{s.t.} & \quad f(x_i) - y_i - c_i = 0, \quad i = 1, 2, \cdots, N.
\end{align*}
\]  

(2)

where \( \gamma \) is the regulation item which controls the balance between the flexibility and accuracy of LSSVM; \( c_i \) denotes the error between the predicted output and real output.

We can define the Lagrange function as

\[
L(w, b, c; a) = \frac{1}{2} w^T w + \frac{1}{2} \gamma \sum_{i=1}^{N} c_i^2
\]  

\[
- \sum_{i=1}^{N} a_i [f(x_i) - y_i - c_i],
\]

(3)

where \( a_i (i = 1, 2, \cdots, N) \) are Lagrange coefficients and \( L(w, b, c; a) \) should meet the Karush–Kuhn–Tucker (KKT) condition, which is a necessary condition to obtain optimal solution in nonlinear programming [29].

Substitute Equation (1) into Equation (3), and then the KKT condition can be described as follows:

\[
\frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=1}^{N} a_i \varphi(x_i),
\]

\[
\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{N} a_i = 0,
\]

\[
\frac{\partial L}{\partial c_i} = 0 \rightarrow a_i = \gamma e_i,
\]

\[
\frac{\partial L}{\partial a_i} = 0 \rightarrow w^T \varphi(x_i) + b + c_i - y_i = 0.
\]  

(4)
After eliminating $\gamma$ and $w$, Equation (4) can be rewritten immediately as the following set of linear equations:

$$
\begin{bmatrix}
0 & \Theta \\
\Theta & \Omega + \gamma^{-1}I
\end{bmatrix}
\begin{bmatrix}
b \\
a
\end{bmatrix}
=
\begin{bmatrix}
0 \\
Y
\end{bmatrix},
$$

(5)

where

$I$ is the identity matrix,

$$
\Theta = [1, 1, \cdots, 1],
$$

$$
Y = [y_1, y_2, \cdots, y_N].
$$

Considering the complexity of calculating the inner product in high dimensionality, we apply kernel trick here—namely,

$$
\Omega_{ij} = \varphi(x_i)^T \varphi(x_j) = \kappa(x_i, x_j)
$$

(6)

Assuming $A = \Omega + \gamma^{-1}I$, matrix $A$ ought to be symmetric positive-definite according to Mercer’s Theorem. Then, parameter $a$ and $b$ can be solved via Equation (6):

$$
\hat{b} = \Theta^T A^{-1} Y \
\hat{a} = A^{-1} (Y - b \Theta).
$$

(7)

After substituting Equations (4) and (7) into Equation (1), the regression model of LSSVM is obtained:

$$
f(x) = \sum_{i=1}^{N} \hat{a}_i \kappa(x, x_i) + \hat{b}.
$$

(8)

Theoretically, any function which satisfies the Mercer condition can be a kernel function. The linear, polynomial, and radial basis function (RBF) are three common types of kernel function. In this paper, the RBF kernel,

$$
\kappa_{RBF}(x_i, x_j) = \exp\left( - \frac{\|x_i - x_j\|^2}{2\sigma^2} \right),
$$

(9)

is selected, where the only parameter $\sigma^2$ is the width of kernel.

2.2. The Hybrid Whale Optimization Algorithm (hWOA)

2.2.1. Whale Optimization Algorithm

The selection of two parameters $\gamma$ and $\sigma^2$ makes a difference to the performance of LSSVM. Optimization algorithms are practically utilized to search optimal parameters by scholars to avoid trail blindness and obtain better performance of the model [10,21,30].

Meta-heuristic algorithms have been applied for optimization in many fields, such as mathematics, data science, and engineering. The reason why meta-heuristic algorithms become increasingly attractive can be attributed to three characteristics: simplicity, flexibility, and the ability to avoid local optima. The WOA is a swarm-based meta-heuristic optimization algorithm which undoubtedly inherits the advantages of meta-heuristic algorithms. Inspired by the humpback whale, Mirjalili and Lewis proposed the WOA by imitating the hunting pattern called bubble-net preying, which is only practiced by humpback whales. The preying behavior of humpback whales can be categorized into two main processes: searching for prey and the bubble-net attacking method. The two maneuvers just correspond to the exploration phase and exploitation phase, respectively, in meta-heuristic algorithms.

In the first maneuver, humpback whales will randomly select a reference whale and then move away by encircling it. The WOA represents this behavior by the following equations:

$$
\vec{D} = \left\| \vec{C} \cdot \frac{\vec{W}X_t}{\sigma} - \frac{\vec{W}X_t}{\sigma} \right\|,
$$

(10)
\[
\overrightarrow{WX}_{t+1} = \overrightarrow{WX}_t - \overrightarrow{A} \cdot \overrightarrow{D}
\]

(11)

where \(t\) indicates the current iteration, \(\overrightarrow{WX}\) is the position vector, and \(\overrightarrow{WX}_*\) is the optimal position obtained so far. \(\overrightarrow{A}, \overrightarrow{C}\) are both coefficients, and they can be calculated as follows:

\[
\overrightarrow{A} = 2 \overrightarrow{a} \cdot \overrightarrow{r}_1 - \overrightarrow{a}
\]

(12)

\[
\overrightarrow{C} = 2 \cdot \overrightarrow{r}_2
\]

(13)

where \(\overrightarrow{a}\) is linearly decreases along with the iteration between \([2, 0]\), and \(\overrightarrow{r}_1, \overrightarrow{r}_2\) are random vectors in range of \([0, 1]\).

Humpback whales will perform the following maneuver (the bubble-net attacking method) after obtaining the approximate location of prey. In this maneuver, humpback whales will approach toward the prey using two different movement patterns: a shrinking circle and a spiral-shaped path. The model of the former path scheme in the WOA is nearly the same as Equations (10)–(13), except random vector \(\overrightarrow{A}\) in Equation (11) is in the range \([-1, 1]\) to force search agents to move toward the targets. For the latter scheme, the WOA mimics the spiral-shaped path by creating a spiral equation between the target and search agent as follows:

\[
\overrightarrow{WX}_{t+1} = \overrightarrow{D'} \cdot e^b \cdot \cos(2\pi l) + \overrightarrow{WX}_t^*,
\]

(14)

\[
\overrightarrow{D'} = |\overrightarrow{WX}_t^* - \overrightarrow{WX}_t|,
\]

(15)

where \(\overrightarrow{D'}\) indicates the distance between the search agent and the target (the optimal position obtained so far), \(b\) is a random constant for defining the shape of spiral, \(l\) is a random number in \([-1, 1]\).

Humpback whales will take two path schemes simultaneously when approaching the prey; thus, the WOA assumes the possibility of each scheme happening is 50% during the optimization. The mathematical model of the bubble-net attacking method is as follows:

\[
\begin{align*}
\overrightarrow{WX}_{t+1} &= \begin{cases} 
\overrightarrow{WX}_t^* - \overrightarrow{A} \cdot \overrightarrow{D} & \text{if } p < 0.5 \\
\overrightarrow{D'} \cdot e^b \cdot \cos(2\pi l) + \overrightarrow{WX}_t^* & \text{if } p \geq 0.5
\end{cases}
\end{align*}
\]

(16)

where \(p\) is a random number in \([0, 1]\).

As shown above, the fitness is the only factor by which to judge a position as better or not. Therefore, cost functions are supposed to reflect the differences between output and real values. Generally, \(L_1\) loss (mean absolute error, MAE):

\[
\text{MAE}(f, D) = \frac{1}{m} \sum_{i=1}^{m} |f(x_i) - y_i|
\]

and \(L_2\) loss (mean square error, MSE):

\[
\text{MSE}(f, D) = \frac{1}{m} \sum_{i=1}^{m} (f(x_i) - y_i)^2
\]

are selected to calculate the fitness on regression tasks. Outliers cause greater loss to MSE than MAE; therefore, the model will try to fit outliers to reduce the cost [31]. The F/T sensors focus on static force, and fewer outliers will make a subtle influence on measurements. Thus, MAE is selected as the cost function in hWOA.

Furthermore, the cross-validation strategy is adopted to eliminate accidental error while calculating the fitness of the algorithm. Cross-validation randomly divides data into \(k\) (10 by default) disjoint sets, and then the \(i\)th set is used to estimate the performance of the
model trained on other \( k - 1 \) sets. Finally, the costs of all \( k \) sets are combined by average to obtain the fitness of the model.

2.2.2. The Simulated Annealing Algorithm (SA)

Annealing is a process which toughens steel or glass by gradually heating and cooling. The SA algorithm combines annealing and the metropolis sampling criterion to obtain a strong ability to escape from local optima. The SA tries to make particles travel around all positions in solution space by setting a high initial temperature. Then, the temperature will gradually decrease to reduce the activeness of particles until the global optimal solution is obtained.

The standard SA is performed in following three steps:

Initialization

Set initial values: \( x^* = x_0, T = T_{\text{max}}, \) and \( k \), where \( x^* \) is the position of search agent in the \( i \)th iteration; \( T \) and \( T_{\text{max}} \) indicate the current temperature and initial maximum temperature, respectively; \( k \) is a constant in \([0, 1]\) to control the decreasing speed of temperature.

Annealing

Select a point \( x_i \) in solution space, and then compare its fitness \( f(x_i) \) with \( f(x^*) \), where \( f(\cdot) \) is the cost function. If \( f(x_i) \) is no more than \( f(x^*) \), then \( x^* = x_i \); otherwise, if \( f(x_i) \) is greater than \( f(x^*) \), the SA will accept \( x_i \) to be the current optimal solution with the possibility of \( \exp\left(-\frac{f(x_i) - f(x^*)}{k \cdot T}\right) \).

Termination

The SA turns back to step (2) repeatedly until the maximum iteration or target temperature is satisfied.

2.2.3. Hybridization

The standard WOA takes a greedy strategy in the exploitation phase. If \( \text{fitness}(\overrightarrow{WX}_t) \) is smaller than \( \text{fitness}(\overrightarrow{WX}^*_t) \), \( \overrightarrow{WX}_t \) will become the best position; otherwise, if \( \text{fitness}(\overrightarrow{WX}_t) \) is greater than \( \text{fitness}(\overrightarrow{WX}^*_t) \), \( \overrightarrow{WX}_t \) will just be ignored. The hWOA performs the same as standard WOA in the former condition, but updates the best position according to randomness and the distance between the current position and the best one in the latter condition. To be specific, if \( \text{fitness}(\overrightarrow{WX}_t) \) is smaller than \( \text{fitness}(\overrightarrow{WX}^*_t) \), hWOA will update the best position with \( \overrightarrow{WX}_t \):

\[
\overrightarrow{WX}^*_t = \overrightarrow{WX}_t, \quad \text{fitness}(\overrightarrow{WX}_t) \leq \text{fitness}(\overrightarrow{WX}^*_t). \tag{17}
\]

when \( \text{fitness}(\overrightarrow{WX}_t) \) is greater than \( \text{fitness}(\overrightarrow{WX}^*_t) \), \( \overrightarrow{WX}_t \) could become the best position in the possibility of \( P_{\text{trans}} \), which is the transfer possibility and defined as follows:

\[
P_{\text{trans}} = \exp\left(-\frac{\text{fitness}(\overrightarrow{WX}_t) - \text{fitness}(\overrightarrow{WX}^*_t)}{T}\right) \tag{18}
\]

where \( T \) indicates the system temperature. Equation (18) demonstrates that the transfer possibility \( P_{\text{trans}} \) is inversely proportional to the distance between \( \overrightarrow{WX}_t \) and \( \overrightarrow{WX}^*_t \), and directly proportional to the system temperature \( T \).

In the progress of optimization, the search agents will get closer to the optima with more iterations, and \( P_{\text{trans}} \) will increase as well. The hWOA will set a high initial temperature \( T_0 \) at
the beginning to keep a balance between stability and exploring ability, and then exponentially decrease $T$ along with iterations as follows:

$$T_{t+1} = k \cdot T_t,$$

where $T_t$ is the system temperature of the $i$th iteration; $k \in [0, 1]$ is a coefficient to control the decreasing rate of $T$. The pseudo-code of hWOA is presented in Algorithm 1.

### Algorithm 1: Hybrid Whale Optimization Algorithm

| Inputs: | Initialize the population of whales $X_i$ ($i = 1, 2, ..., n$), Initialize the parameters $a, p, l, \vec{A}, \vec{C}$ and $X_{curr}$ |
| Outputs: | The optimal position $X^*$ |
| Process: | 1. Calculate fitness of all search agents 2. while ($t < \text{maximum number of iterations}$) || ($\text{stop condition}$): 3. Update $p$ and flag 4. for each search agent: 5. if $p > 0.5$: 6. if $|A| > 1$: 7. Update the position of the current search agent by Equation (11) 8. else if $|A| \leq 1$: 9. Select a random search agent $\vec{X}_{\text{rand}}$ 10. Update the position of the current search agent by Equation (15) 11. end 12. else if $p \leq 0.5$: 13. Update the position of the current search agent by Equation (14) 14. end if 15. if fitness($\vec{X}_{curr}$) < fitness($\vec{X}_{curr}^*$): $\vec{X}_{curr}^* = \vec{X}_{curr}$ 16. else if fitness($\vec{X}_{curr}$) $\geq$ fitness($\vec{X}_{curr}^*$): 17. if flag $> P_{\text{trans}}$: do nothing 18. else if flag $\leq P_{\text{trans}}$: $\vec{X}_{curr}^* = \vec{X}_{curr}$ 19. end if 20. end if 21. end for 22. Check if any search agent travels beyond the search space and amend it 23. $t = t + 1$ 24. $T = k \cdot T$ % Update the system temperature by Equation (18) 25. end while 26. return $\vec{X}^*$

2.3. Improved Trimmed Bagging

Trimmed bagging is a modified version of the bagging algorithm. Since being proposed by Breiman in 1996, the bagging algorithm took a place in ensemble learning field due to its effectiveness and flexibility. The bagging algorithm can outperform a single learner under the limitation that the base learner should be both good on average and unstable to the disturbances of training datasets [32]. Therefore, decision trees, a typically unstable base learner, are proven to be well compatible with the bagging algorithm [33]. On the other hand, if stable algorithms, such as SVM and logistic regression, are selected to be base learners, the performance of bagging could be equivalent or even worse than that of the base learner [34]. To extend the compatibility of bagging to stable algorithms, Joossens and Croux improved the standard bagging algorithm with an intuitive idea: the trimmed bagging only averaging over those “good” base learners instead of all base learners. The complete process of trimmed bagging can be presented as follows:
1. The algorithm begins by applying bootstrap sampling on the given dataset. Assuming 
\( D = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m) \} \) denotes the original training datasets, we can 
obtain the bootstrap distributions \( D_i, i = 1, 2, \ldots, T \), where \( T \) indicates the number of 
base learners.

2. Train all base learners with the corresponding subsets:
\[
h_i = \mathcal{L}(D, D_i), \quad i = 1, 2, \ldots, T
\]
where \( \mathcal{L}(D, D_i) \) stands for training the base learning algorithm \( \mathcal{L} \) with dataset \( D \) under 
distribution \( D_i \), \( h_i \) is the \( i \)th trained learner, and \( T \) is the number of base learners.

3. Calculate the quoted error (QE) of each base learner using OOB sampling:
\[
QE(i) = \text{average} \left\{ f(x) - y \right\}_{(x,y) \in (D, D_oob)}
\]
\[
= \frac{1}{n-k} \sum_{i=1}^{n-k} | f(x_i) - y_i |_{FS}
\]

4. Sort all base learners ascendingly according to their QEs:
\[
QE(\mathcal{L}(D, D_{oob}^1)) \leq QE(\mathcal{L}(D, D_{oob}^2)) \leq \cdots \leq QE(\mathcal{L}(D, D_{oob}^j))
\]
where \( j = 1, 2, \ldots, T \) indicates the quoted error of \( D_{oob}^j \) at the \( j \)th iteration for all 
base learners.

5. Trim off the “worst” learners at the portion of \( \alpha \); then average them over the remainder 
to obtain the ensemble learner:
\[
h_{trim} = \text{average} \mathcal{L}(D, D_b)
\]

As shown above, the performance of the trimmed bagging is directly influenced by 
the trim portion \( \alpha \): too large an \( \alpha \) will cause too few base learners to average over, and the 
bagging method is meaningless. On the other hand, too small an \( \alpha \) will increase the risk of 
involving some “worst” learners. In standard trimmed bagging, the trimmed portion is 
fixed and less than 0.5. Obviously, the portions of the “worst” learners in ensemble models 
are various; a fixed trim portion \( \alpha \) cannot be suitable for all situations. An adaptive strategy 
is designed to choose the trim portion \( \alpha \) to improve the robustness of the model.

The adaptive trimming strategy chooses the portion \( \alpha \) by minimizing the variance of 
QEs of all base learners involved in averaging. To be specific, it calculates the mean and 
variance, respectively, of different trim counts by using the following formulas:

\[
\text{Mean}(k) = \frac{1}{n-k} \sum_{i=1}^{n-k} QE(i),
\]
\[
\text{Var}(k) = \frac{1}{n-k} \sum_{i=1}^{n-k} [QE(i) - \text{Mean}(k)]^2,
\]
where \( k \) indicates the number of trimmed base learners, and \( QE(i) \) stands for the quoted 
error of the \( i \)th base learner ordered by \( QE \). Then, choose the optimal trim count \( k^* \), which 
minimizes the variance:
\[
k^* = \{ k \mid \min \text{Var}(k), \lfloor n/2 \rfloor \leq k \leq n \}.
\]
Correspondingly, the optimal trim portion \( \alpha^* \) can be obtained as follows:
\[
\alpha^* = \frac{k^*}{n}
\]
After trimming off the base learners at portion $α^*$, the standard trimmed bagging will apply arithmetic averaging over the remaining base learners. The arithmetic average distributes each base learner with the same weight, and learners contribute equally to the model. However, base learners perform diversely, and we think the better a base learner performs, the greater its weight should be. Based on this idea, the weights of base learners are equivalent to the inverses of their $QEs$:

$$w_i = \frac{1/QE_i}{\sum_{j=1}^{N-k^*} (1/QE_j)}$$  

(27)

where $w_i$ indicates the weight of the $i$th base learner after trimming.

After obtaining the optimal trim portion $α$ and the weight of each base learner, the improved trimmed bagging assembles the outputs of base learners as follows:

$$h_{trim} = \left\lceil \left(1 - α^* \right) \cdot N \right\rceil \sum_{j=1}^{\left\lceil (1-α^*) \cdot N \right\rceil} w_i \cdot L(D, D_j)$$

$$\sum_{k=1}^{\left\lceil (1-α^*) \cdot N \right\rceil} w_k$$

(28)

The pseudo-code of the improved trimmed bagging algorithm can be presented as Algorithm 2.

---

**Algorithm 2: Improved Trimmed Bagging Algorithm**

**Inputs:**
- Dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$
- Base learning algorithm $L$
- The trimming portion $α$
- Number of base learners $T$

**Outputs:**
The ensemble learner $h_{trim}$

**Process:**
1. Initial the distributions $D_i$ $(i = 1, 2, \ldots, T)$ by applying bootstrap sampling over dataset $D$
2. Initial out-of-bag samples $D_{i^{\text{ob}}}$ $(i = 1, 2, \ldots, T)$ correspond to distribution $D_i$
3. for $i = 1, 2, \ldots, T$
4. $h_i = L(D, D_i)$ %Train the base learning algorithm by distribution $D_i$
5. $qe_i = QE(L(D, D_{i^{\text{ob}}}))$ %Calculate quoted error of current learner
6. end
7. Sort all learners ascendingly according to quoted error.
8. Find the optimal trim portion $α$ by Equation (23)–(26).
9. Calculate the weight $w_i$ of each base learner by Equation (27).
10. $h_{trim} = \left[\sum_{i=1}^{\left\lceil (1-α^*) \cdot N \right\rceil} w_i \cdot L(D, D_i)\right] / \left[\sum_{i=1}^{\left\lceil (1-α^*) \cdot N \right\rceil} w_i\right]$ %Average on the remaining base learners
11. return $h_{trim}$

---

3. Hybrid WOA-LSSVM Ensembled by Improved Trimmed Bagging

This section introduces how the presented algorithms interconnect and how to constitute the proposed temperature compensation model.

1. We prepared the datasets with a high–low temperature experiment. The input vector was formed by a series of ambient temperatures ($T$), and the voltage signal ($V$) of the strain gage sensor constituted the output vector. Then, 80% of origin data were randomly separated for training and the remaining 20% were prepared for testing. More details about the datasets are shown in the High–Low Temperature Experiment section.
2. We divided the training set into $N$ distributions by bootstrap sampling and obtained base learners by training LSSVM on each distribution. Two parameters of LSSVM,
the regulation item ($\gamma$) and the width of RBF kernel ($\sigma^2$), were tuned by hWOA in this process.

3. We calculated quoted error ($QE$) of each base learner according Equation (21) by out-of-bag (OOB) sampling; then, we determined the optimal trim portion ($\alpha^*$) and weight ($\omega_i$) by the adaptive trim strategy.

4. We assembled the final model by trimming off and averaging over the outputs of base learners according to $\alpha^*$ and $\omega_i$.

The flowchart of the hybrid whale optimization algorithm optimized least-squares support vector machine (hWOA-LSSVM) ensembled by improved trimmed bagging is demonstrated in Figure 1.

![Flowchart of ensemble hWOA-LSSVM based on improved trimmed bagging.](image)

**Figure 1.** Flowchart of ensemble hWOA-LSSVM based on improved trimmed bagging.

4. **Experiment**

All experiments were conducted on a six-axis force/torque sensor designed by the Institute of Intelligent Machines (IIM), Chinese Academy of Sciences (CAS). The F/T sensor consists of strain gauges and a novel double E-shape elastic body. The rated ranges of the F/T sensor are $F_x = F_y = \pm 600$ N, $F_z = \pm 1000$ N, and $M_x = M_y = M_z = \pm 30$ N·m.

4.1. **Calibration and Decoupling**

A calibration experiment was conducted to obtain the transfer expression from outer stimuli to voltage signal responded by the sensor. Assuming $U = [u_1, u_2, \ldots, u_6]^T$ indicates the output vector, which consists of the voltage of each dimension, and $L = [F_x, F_y, F_z, M_x, M_y, M_z]^T$
indicates the measured load vector, the transfer expression can be presented by the following equation:

\[ L = W \cdot U + B. \]  
\(29\)

where \( W \) is the weight matrix, which is also called the calibration matrix, and \( B \) is the bias vector.

The basic procedures for multi-axis F/T sensor calibration apply a series of known loads which increase from minimum to maximum rated range with a certain step. The voltage responded by the sensor is recorded at each sample point. The above procedures were repeated for three times in this calibration experiment, and all record data were used for calibration and decoupling. Configurations of calibration experiment are shown in Table 1, and the environment temperature and humidity are 25 °C and 60%, respectively.

Table 1. Calibration experiment configuration.

| Dimensions | Load Points | Units |
|------------|-------------|-------|
| Fx         | -600, -400, -200, 0, 200, 400, 600 | N     |
| Fy         | -600, -400, -200, 0, 200, 400, 600 | N     |
| Fz         | 0, 200, 600, 800, 1000           | N     |
| Mx         | -30, -20, -10, 0, 10, 20, 30     | N·m   |
| My         | -30, -20, -10, 0, 10, 20, 30     | N·m   |
| Mz         | -30, -20, -10, 0, 10, 20, 30     | N·m   |

After all calibration procedures completed, the least-squares (LS) algorithm was adopted to calculate the calibration matrix \( W \) and bias vector \( B \), which are as follows:

\[
W = \begin{bmatrix}
643.017 & -7.270 & 148.091 & -0.068 & -430.623 & 6.890 \\
544.952 & 46.697 & -170.297 & 28.248 & -349.738 & -452.637 \\
-11.149 & 10.105 & 266.068 & 2.265 & 11.459 & -5.324 \\
6.419 & -6.059 & -2.374 & 0.359 & -4.202 & -5.568 \\
-1.132 & 0.109 & -2.443 & 0.010 & 5.178 & 0.060 \\
0.374 & -0.023 & -4.341 & -0.017 & -0.111 & 6.426
\end{bmatrix}, \quad (30)
\]

\[
B = \begin{bmatrix}
-258.561, -202.996, -559.61, -2.9339, 0.3304, -2.7932
\end{bmatrix}^T, \quad (31)
\]

Finally, the transfer expression could be obtained by substituting \( W \) and \( B \) into Equation (29).

4.2. High–Low Temperature Experiment

A high–low temperature experiment was conducted to analyze the measurement error of sensors caused by temperature drift and obtain the data for the training model. Wang shows that exerting loads on F/T sensors makes no difference to the temperature drift phenomenon [35]; therefore, the sensors were not loaded with any force/toque during the temperature experiment.

The six-axis F/T sensor was placed in a high–low temperature chamber and run at 5V DC voltage in the experiment. The temperature in the chamber was varied from -30 to 70 °C, and kept for 2 h at thirteen temperature sampling points (marked as \( T_s \)): -30, -20, -10, 0, 10, 25, 30, 35, 40, 50, 60, and 70 °C. The gathering module sampled about 450 outputs (marked as \( U_o \)) of the F/T sensor during each \( T_s \) and transmitted them to the PC for processing and storage. The temperature experiment configuration is demonstrated in Figure 2.
The sensor measurement error $E_m$ is convenient for comparing, which can be defined as follows:

$$E_m = \frac{\theta_T - \theta_{\text{ref}}}{\theta_{FS}} \times 100\%$$ (32)

where

- $\theta_T$ is the measured value of the F/T sensor under $T^\circ\text{C}$;
- $\theta_{\text{ref}}$ denotes the measured value under the temperature of calibration, which means 25 $^\circ\text{C}$ here;
- $\theta_{FS}$ denotes the full scale of the corresponding dimension.

Measurement errors caused by temperature in all six dimensions before compensation are illustrated in Figure 3.

As is shown in Figure 3, all dimensions of the F/T sensor suffer characteristic drift, which consists of both linear and nonlinear components. The relation between $E_m$ and $T_s$ presents more nonlinear features when ambient temperature is higher than 50 $^\circ\text{C}$. In addition, $F_y$, $F_z$, $M_y$, and $M_z$ have positive correlations with $E_m$ and $T_s$; and $F_x$ and $M_x$ have negative correlations with them.

Overall, $F_z$, $M_x$, and $M_z$ have less $E_m$ and lower linearity than $F_x$, $F_y$, and $M_y$. $M_x$ had the lowest $E_m$, which is no more than $0.6\%_{FS}$, and $F_x$ had the largest $E_m$, which reached $3.3\%_{FS}$ at $-30^\circ\text{C}$. The measurement error of the F/T sensor caused by temperature variation was manifest; therefore, temperature compensation is vital for six-axis F/T sensors to meet the requirements of space manipulator control.
4.3. F/T Sensor Temperature Compensation

4.3.1. Model Training

The original dataset gathered in the temperature experiment was divided into 80% and 20% for training and testing, respectively. To evaluate the proposed model better, several algorithms, including standard support vector regression (Std-SVR), LSSVM optimized by particle swarm optimization (PSO-LSSVM), LSSVM optimized by the standard whale optimization algorithm (WOA-LSSVM), and the RBF neural network optimized by the standard whale optimization algorithm (WOA-RBFNN) were compared.

In the RBFNN, the number of neurons in the hidden layer was 13, and the centers of RBF function were set to $-30, -20, -10, 0, 10, 20, 25, 30, 35, 40, 50, 60,$ and $70$, which correspond to the 13 sampling points in the aforesaid high–low temperature experiment.

The parameters for all algorithms were as identical as possible; all the parameters are demonstrated in Table 2. Additionally, the mean square error (MSE) was used as the fitness function to evaluate the performances of algorithms.

Table 2. Parameters of all models.

| Parameters                      | Std-SVR | EhW-LSSVM | PSO-LSSVM | WOA-LSSVM | WOA-RBFNN |
|---------------------------------|---------|-----------|-----------|-----------|-----------|
| $\gamma$                        | -       | [0.01, 300]| [0.01, 300]| -         | [0.01, 300]|
| $\sigma^2$                      | 50      | [1, 1000] | [1, 1000] | [1, 1000] | [1, 1000] |
| Maximum iteration               | -       | 30        | 30        | 30        | 30        |
| Count of search agents          | -       | 20        | 20        | 20        | 20        |
| Count of base learners          | -       | 10        | -         | -         | -         |

4.3.2. Compensation

We compensated for the characteristic drift by connecting the trained temperature compensation model to outputs of the F/T sensor. To be specific, as is shown in Figure 4, the multi-axis F/T sensor changed force and torque into voltage outputs, and then the compensation model calculated thermal outputs (which could be either positive or negative) according to the voltage outputs and current temperature. Compensation outputs can be obtained by subtracting thermal outputs from raw outputs of the sensor. Finally, the measured values were obtained by substituting the compensation outputs into the transfer Expression (29).
WOA-RBFNN deteriorated greatly in the testing set, which indicates that the RBFNN was the worst performer in the testing set and training set. Second, using the bagging ensemble method, the ensembled model did not show worse performance in training than the single learner, but obtained better predicting ability.

Table 3. Compensation results on the training set by different methods.

| Parameters | Std-SVR | EhW-LSSVM | PSO-LSSVM | WOA-LSSVM | WOA-RBFNN |
|------------|---------|-----------|-----------|-----------|-----------|
| Fx         | $2.6340 \times 10^{-5}$ | $1.4106 \times 10^{-5}$ | $1.0572 \times 10^{-5}$ | $1.4136 \times 10^{-5}$ | $1.2504 \times 10^{-5}$ |
| Fy         | $9.7145 \times 10^{-6}$ | $9.0614 \times 10^{-6}$ | $6.1726 \times 10^{-6}$ | $9.0729 \times 10^{-6}$ | $6.8415 \times 10^{-6}$ |
| Fz         | $4.4808 \times 10^{-5}$ | $3.3610 \times 10^{-5}$ | $2.7122 \times 10^{-5}$ | $3.3695 \times 10^{-5}$ | $2.9582 \times 10^{-5}$ |
| Mx         | $4.6609 \times 10^{-5}$ | $4.5047 \times 10^{-5}$ | $3.2643 \times 10^{-5}$ | $4.5115 \times 10^{-5}$ | $3.5834 \times 10^{-5}$ |
| My         | $3.2747 \times 10^{-5}$ | $1.0729 \times 10^{-5}$ | $1.6889 \times 10^{-5}$ | $1.0775 \times 10^{-5}$ | $1.8464 \times 10^{-5}$ |
| Mz         | $6.4732 \times 10^{-5}$ | $2.9451 \times 10^{-5}$ | $3.1305 \times 10^{-5}$ | $2.9531 \times 10^{-5}$ | $3.5159 \times 10^{-5}$ |

Table 4. Compensation results on the testing set by different methods.

| Parameters | Std-SVR | EhW-LSSVM | PSO-LSSVM | WOA-LSSVM | WOA-RBFNN |
|------------|---------|-----------|-----------|-----------|-----------|
| Fx         | $3.6100 \times 10^{-5}$ | $2.9417 \times 10^{-5}$ | $2.9311 \times 10^{-5}$ | $3.3722 \times 10^{-5}$ | $4.2461 \times 10^{-2}$ |
| Fy         | $1.8291 \times 10^{-5}$ | $1.0058 \times 10^{-5}$ | $1.0456 \times 10^{-5}$ | $1.0266 \times 10^{-5}$ | $1.5098 \times 10^{-3}$ |
| Fz         | $5.8494 \times 10^{-5}$ | $4.0726 \times 10^{-5}$ | $4.2809 \times 10^{-5}$ | $4.2876 \times 10^{-5}$ | $9.6410 \times 10^{-2}$ |
| Mx         | $7.2215 \times 10^{-5}$ | $4.5470 \times 10^{-5}$ | $4.6506 \times 10^{-5}$ | $4.5767 \times 10^{-5}$ | $8.1106 \times 10^{-2}$ |
| My         | $4.3461 \times 10^{-5}$ | $3.0991 \times 10^{-5}$ | $2.9197 \times 10^{-5}$ | $3.2692 \times 10^{-5}$ | $4.0547 \times 10^{-2}$ |
| Mz         | $8.7759 \times 10^{-5}$ | $1.1178 \times 10^{-5}$ | $1.3900 \times 10^{-5}$ | $6.8330 \times 10^{-5}$ | $7.8865 \times 10^{-5}$ |

In the testing set, as Table 4 shows, the EhW-LSSVM gained the lowest fitness in four dimensions (Fy, Fz, Mx, Mz), and the PSO-LSSVM performed better in Fx and My. The performance of the EhW-LSSVM was better than that of the single learner WOA-LSSVM in most dimensions. In addition, the fitness of all dimensions except Mz obtained by the WOA-RBFNN deteriorated greatly in the testing set, which indicates that the RBFNN suffers severe over-fitting under the same parameter configuration as the other algorithms.

We can draw the following inferences by analyzing the temperature compensation results: first, the EhW-LSSVM shows competitive predictive ability because of its reliable performance on the testing set and training set. Second, using the bagging ensemble method, the ensembled model did not show worse performance in training than the single learner, but obtained better predicting ability.

Figure 4. Compensating procedures of ensemble hWOA-LSSVM based on improved trimmed bagging.

4.4. Compensation Results and Analysis

The compensation results of various dimensions on the training set and testing set are shown in Tables 3 and 4, respectively. As demonstrated in Table 3, in the training set, PSO-LSSVM performed best in four dimensions (Fx, Fy, Fz, Mx), and EaW-LSSVM had the worst fitness in the other two dimensions (My, Mz). The ensemble model EhW-LSSVM performed slightly better than the single model WOA-LSSVM in the training set. The performance of WOA-RBFNN was average overall, and Std-SVM was inferior to other algorithms.
The best fitness of $F_z$ in each iteration obtained on the training set was taken as an example to evaluate the search convergence characteristics. Additionally, the convergence curves of EhW-LSSVM (average on all base learners), PSO-LSSVM, and WOA-LSSVM are demonstrated in Figure 5. As depicted in Figure 5a,c, the hybrid whale optimization algorithm jumps out of local optima but drops in fitness sometimes. It is inferred that, benefiting from the hybridization of the SA, the hWOA has a remarkable ability to jump out of local optima and still retains good global searching ability. In addition, the convergence characteristic of PSO in Figure 5b shows that the PSO converges too early and has a poor ability to get out of local optima.

![Figure 5](image)

Figure 5. Best fitness obtained on the training set by each algorithm. (a) Best fitness obtained throughout iterations by improved WOA. (b) Best fitness obtained throughout iterations by PSO. (c) Best fitness obtained throughout iterations by standard WOA.

After compensating with the EhW-LSSVM, the measurement errors caused by temperature variation are demonstrated in Figure 6. Overall, the measurement errors of all dimensions decreased dramatically after compensation by the EhW-LSSVM. $F_z$ and $M_y$ had the best compensating performances, whose maximum measurement errors were less than $2.2 \times 10^{-5}\%_{FS}$ and $1.7 \times 10^{-4}\%_{FS}$, respectively. The performances of $F_x$ and $F_y$ were somewhat inferior to those of other dimensions, and the maximum absolute measurement error of $F_x$ was still less than $3.9 \times 10^{-3}\%_{FS}$. In addition, the measurement errors of all dimensions showed no noticeable changes with temperature variation. Above all, the
six-axis F/T sensor suffered from the temperature drift negligibly and met the request of cosmic operation after compensation by the EhW-LSSVM.

Figure 6. Measurement errors after compensating by EhW-LSSVM.

5. Conclusions

Our novel temperature compensating method consisting of LSSVM optimized by hybrid WOA and improved trimmed bagging was presented in this work for eliminating the characteristic drift of six-axis force/torque sensors in cosmic space. In addition, simulated annealing (SA) is applied to WOA to cover the shortage in exploiting. Furthermore, the optimal trim portion of trimmed bagging is determined by an adaptive trimming strategy, which automatically adjusts the trim portion according to the performances of base learners. Cross-validation and inverse quoted error are utilized to evaluate the model more accurately.

A high–low temperature experiment was conducted to investigate the impacts of temperature variation on six-axis F/T sensors and provide data for model training. The compensating results indicate that EhW-LSSVM possesses excellent predicting ability and dramatically decreased the measurement errors of six-axis F/T sensors to a level of $10^{-3}\%_{\text{FS}}$. The hybrid WOA showed better ability than standard WOA during the process with search optimal parameters. In addition, the adaptive trimmed bagging lifted the effect of a single model in the testing set while losing no accuracy in the training set. According to temperature compensating results and comparisons with other algorithms, the EhW-LSSVM algorithm is a feasible and competitive temperature compensating method for six-axis F/T sensors.

The compensating performance of the EhW-LSSVM is satisfactory, but the complexity of its structure is also high. In future research, we will aim to reduce the model’s complexity and try to integrate the presented EhW-LSSVM into compact six-axis F/T sensors.

Author Contributions: X.L. and H.C. conceived this research and designed the proposed model; Y.S. and M.J. devised experiments; X.L. programmed the algorithms; Y.Z. conducted experiments; X.L. prepared this manuscript; L.G. revised the language expression and technical coherence. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Key research projects supported by the National Natural Science Foundation of China, grant number 92067205; in part by the Major science and technology project of Anhui Province, grant number 202103a05020022; in part by Key Research and Development Project of Anhui Province, grant number 2022a05020035; in part by the Strategic Priority Research Program of the Chinese Academy of Sciences, grant number XDA22040303; and in part by the HFIPS Director’s Fund, grant number YZJJ2021QN25.


**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Yuxiang, S.; Yuyun, Y.; Huibin, C.; Shuang, F.; Lifu, G.; Yunjian, G. Research on Joint Torque Sensor for Space Manipulator Based on Redundant Measurement. *Chin. J. Sens. Actuators* **2018**, *31*, 1621–1627.

2. Chen, Y.; Zhang, Q.; Tian, Q.; Huo, L.; Feng, X. Fuzzy Adaptive Impedance Control for Deep-Sea Hydraulic Manipulator Grasping Under Uncertainties. In *Proceedings of the Global Oceans: Singapore—U.S. Gulf Coast*, Biloxi, MS, USA, 5–30 October 2020; IEEE: Biloxi, MS, USA, 2020; pp. 1–6. [CrossRef]

3. Bai, Y.; Zhang, Q.; Tian, Q.; Yan, S.; Tang, Y.; Zhang, A. Performance and experiment of deep-sea master-slave servo electric manipulator. In *Proceedings of the OCEANS MTS/IEEE Seattle*, OCEANS, Seattle, WA, USA, 27–31 October 2019; IEEE: Seattle, WA, USA, 2019; pp. 1–5. [CrossRef]

4. Luo, H.; Cabot, J.; Duan, M.; Lee, Y.K. An Integrated Temperature Compensation Method for Thermal Expansion-based Angular Motion Sensors. In *Proceedings of the IEEE Sensors*, Sydney, Australia, 31 October–3 November 2021; IEEE: Sydney, Australia, 2021; pp. 1–4. [CrossRef]

5. Pereira, R.d.S.; Cima, C.A. Thermal Compensation Method for Piezoresistive Pressure Transducer. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 9510807. [CrossRef]

6. Hewes, A.; Medvescek, J.I.; Mydlarski, L.; Baliga, B.R. Drift compensation in thermal anemometry. *Meas. Sci. Technol.* **2020**, *31*, 045302. [CrossRef]

7. Wang, L.; Zhu, R.; Li, G. Temperature and Strain Compensation for Flexible Sensors Based on Thermosensation. *ACS Appl. Mater. Interfaces* **2020**, *12*, 1953–1961. [CrossRef]

8. Yang, Y.; Liu, Y.; Liu, Y.; Zhao, X. Temperature Compensation of MEMS Gyroscope Based on Support Vector Machine Optimized by GA. In *Proceedings of the 2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, Xiamen, China, 6–9 December 2019; IEEE: Xiamen, China, 2019; pp. 2989–2994. [CrossRef]

9. Deng, J.; Chen, W.L.; Liang, C.; Wang, W.F.; Xiao, Y.; Wang, C.P.; Shu, C.M. Correction model for CO detection in the coal combustion loss process in mines based on GWO-SVM. *J. Loss Prev. Process Ind.* **2021**, *71*, 104439. [CrossRef]

10. Ma, H.G.; Zeng, G.H.; Huang, B. Research on Temperature Compensation of Pressure Transmitter Based on WOA-BP. *Instrum. Technol. Sens.* **2020**, *6*, 33–36.

11. Wang, Y.; Xiao, S.; Tao, J. Temperature Compensation for MEMS Mass Flow Sensors Based on Back Propagation Neural Network. In *Proceedings of the Annual International Conference on Nano/Micro Engineered and Molecular Systems (NEMS)*, NEMS, Xiamen, China, 25–29 April 2021; IEEE: Xiamen, China, 2021; pp. 1601–1604. [CrossRef]

12. Chung, V.P.J.; Lin, Y.C.; Li, X.; Guney, M.G.; Paramesh, J.; Mukherjee, T.; Fedder, G.K. Stress-and-Temperature-Induced Drift Compensation on a High Dynamic Range Accelerometer Array Using Deep Neural Networks. In *Proceedings of the IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*, Gainesville, FL, USA, 25–29 January 2021; IEEE: Gainesville, FL, USA, 2021; pp. 188–191. [CrossRef]

13. Yanmei, S.; Shudong, L.; Fengjuan, M.; Bairui, T. Temperature Compensation Model Based on the Wavelet Neural Network with Genetic Algorithm. *Chin. J. Sens. Actuators* **2012**, *25*, 77–81.

14. Seo, Y.B.; Yu, H.; Yu, M.J.; Lee, S.J.; IEEE. Compensation Method of Gyroscope Bias Hysteresis Error with Temperature and Rate of Temperature using Deep Neural Networks. In *Proceedings of the International Conference on Control, Automation and Systems (ICCAS)*, PyeongChang, Korea, 17–20 October 2018; number WOS:000457612300164; pp. 1072–1076.

15. Shi, S.; Wang, Z.; Guo, J.; Huang, Y. Temperature Compensation Technology of Speckle Structured Light Based on BP Neural Network. In *Proceedings of the Sixth Symposium on Novel Photoelectronic Detection Technology and Application*, Beijing, China, 3–5 December 2019; Volume 11455. [CrossRef]

16. Suykens, J.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [CrossRef]

17. Zhang, J.A.; Wang, P.; Gong, Q.R.; Cheng, Z. SOH estimation of lithium-ion batteries based on least squares support vector machine error compensation model. *J. Power Electron.* **2021**, *21*, 1712–1723. [CrossRef]

18. Tan, F.; Yin, G.; Zheng, K.; Wang, X. Thermal error prediction of machine tool spindle using segment fusion LSSVM. *Int. J. Adv. Manuf. Technol.* **2021**, *116*, 99–114. [CrossRef]

19. Mohammed, A.J.; Ghathwan, K.I.; Yusof, Y. A hybrid least squares support vector machine with bat and cuckoo search algorithms for time series forecasting. *J. Inf. Commun. Technol.-Malays.* **2020**, *19*, 351–379. [CrossRef]

20. Song, Y.; Niu, W.; Wang, Y.; Xie, X.; Yang, S.; IEEE. A Novel Method for Energy Consumption Prediction of Underwater Gliders Using Optimal LSSVM with PSO Algorithm. In *Global Oceans 2020: Singapore—US Gulf Coast*; IEEE: Piscataway, NJ, USA, 2020. [CrossRef]

21. Wu, H.; Wang, J. A Method for Prediction of Waterlogging Economic Losses in a Subway Station Project. *Mathematics* **2021**, *9*, 1421. [CrossRef]
22. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* 2016, 95, 51–67. [CrossRef]
23. Abd Elaziz, M.; Mirjalili, S. A hyper-heuristic for improving the initial population of whale optimization algorithm. *Knowl. Based Syst.* 2019, 172, 42–63. [CrossRef]
24. Feng, W. Convergence Analysis of Whale Optimization Algorithm. *J. Phys. Conf. Ser.* 2021, 1757, 012008. [CrossRef]
25. Tong, W. A Hybrid Algorithm Framework with Learning and Complementary Fusion Features for Whale Optimization Algorithm. *Sci. Program.* 2020, 2020, 5684939. [CrossRef]
26. Rana, N.; Abd Latiff, M.S.; Abdulhamid, S.M.; Misra, S. A hybrid whale optimization algorithm with differential evolution optimization for multi-objective virtual machine scheduling in cloud computing. *Eng. Optimiz.* 2021. [CrossRef]
27. Zhou, Z.H. *Ensemble Methods: Foundations and Algorithms*; CRC Press: Boca Raton, FL, USA, 2012. [CrossRef]
28. Breiman, L. Bagging predictors. *Mach. Learn.* 1996, 24, 123–140. [CrossRef]
29. Breiman, L. Arcing classifiers. *Ann. Stat.* 1998, 26, 801–824.
30. Kouziokas, G.N. SVM kernel based on particle swarm optimized vector and Bayesian optimized SVM in atmospheric particulate matter forecasting. *Appl. Soft. Comput.* 2020, 93, 106410. [CrossRef]
31. ELSEBACH, R. Evaluation of forecasts in ar models with outliers. *OR Spektrum* 1994, 16, 41–45. [CrossRef]
32. Breiman, L. Arcing classifiers. *Ann. Stat.* 1998, 26, 801–824.
33. Breiman, L. Random forests. *Mach. Learn.* 2001, 45, 5–32. [CrossRef]
34. Dietterich, T. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Mach. Learn.* 2000, 40, 139–157. [CrossRef]
35. Wang, Z. Research on Static and Dynamic Characteristics and Temperature Compensation of Six-Axis Force Sensor. Ph.D. Thesis, Hefei University of Technology, Hefei, China, 2021.