

Article

Modeling and Compensation of Random Drift of MEMS Gyroscopes Based on Least Squares Support Vector Machine Optimized by Chaotic Particle Swarm Optimization

Haifeng Xing , Bo Hou, Zhihui Lin and Meifeng Guo *

Engineering Research Center for Navigation Technology, Department of Precision Instruments, Tsinghua University, Beijing 100084, China; xhf15@mails.tsinghua.edu.cn (H.X.); houb15@mails.tsinghua.edu.cn (B.H.); linzh15@mails.tsinghua.edu.cn (Z.L.)

* Correspondence: guomf@mail.tsinghua.edu.cn; Tel.: +86-135-2082-6969

Received: 11 September 2017; Accepted: 11 October 2017; Published: 13 October 2017

Abstract: MEMS (Micro Electro Mechanical System) gyroscopes have been widely applied to various fields, but MEMS gyroscope random drift has nonlinear and non-stationary characteristics. It has attracted much attention to model and compensate the random drift because it can improve the precision of inertial devices. This paper has proposed to use wavelet filtering to reduce noise in the original data of MEMS gyroscopes, then reconstruct the random drift data with PSR (phase space reconstruction), and establish the model for the reconstructed data by LSSVM (least squares support vector machine), of which the parameters were optimized using CPSO (chaotic particle swarm optimization). Comparing the effect of modeling the MEMS gyroscope random drift with BP-ANN (back propagation artificial neural network) and the proposed method, the results showed that the latter had a better prediction accuracy. Using the compensation of three groups of MEMS gyroscope random drift data, the standard deviation of three groups of experimental data dropped from $0.00354^\circ/\text{s}$, $0.00412^\circ/\text{s}$, and $0.00328^\circ/\text{s}$ to $0.00065^\circ/\text{s}$, $0.00072^\circ/\text{s}$ and $0.00061^\circ/\text{s}$, respectively, which demonstrated that the proposed method can reduce the influence of MEMS gyroscope random drift and verified the effectiveness of this method for modeling MEMS gyroscope random drift.

Keywords: MEMS gyroscope random drift; phase space reconstruction; back propagation artificial neural network; least squares support vector machine; chaotic particle swarm optimization

1. Introduction

MEMS (Micro Electro Mechanical System) gyroscopes have the advantages of being small in size, lightweight, low cost, vibration resistant, and so on. Therefore, MEMS gyroscopes have been used widely in civil and military fields such as automobiles, UAVs (unmanned aerial vehicles) and weapon guidance systems [1,2]. However, compared to high precision laser gyroscopes or fiber optic gyroscopes, the accuracy of the MEMS gyroscope is low and the random drift has nonlinear and non-stationary characteristics due to the limitation of the current material processing technology [3–5]. These disadvantages make the MEMS gyroscopes still inapplicable in many high-precision fields. Therefore, it is of great significance for modeling to compensate for MEMS gyroscope random drift.

Presently, the common methods for modeling MEMS gyroscope random drift are two classes, one is the statistical modeling method represented by traditional time series analysis, and the other is the intelligence algorithm represented by ANNs (artificial neural networks) for modeling the MEMS gyroscope random drift [6–8]. The time series model of random drift assumes that MEMS gyroscope measurement noise is a linear combination of historical data and historical white noise, and the ARMA

(auto regressive and moving average) model for drift modeling is often used [9]. However, this method assumes the random drift is stationary which will inevitably limit the range of application and the prediction accuracy of the model [3]. Some scholars put forward reducing the adverse effect of the MEMS gyroscope random drift using a wavelet analysis method, since wavelet transform can subdivide the specified signal in the frequency domain and time domain simultaneously to obtain more details. Conversely, Fourier transform can only be used in the frequency domain analysis, so wavelet transform has a better signal processing ability [10,11].

Through the development of intelligent algorithms, there are more ways to model and compensate the random drift of MEMS gyroscope. The ANNs are common intelligent algorithms, and have achieved good results in random drift modeling [12–14]. While ANNs can theoretically approximate any nonlinear function, there may be overfitting problems during training, and the number of nodes in the hidden layer depends on experience and lacks theoretical guidance [15,16]. The LSSVM (least squares support vector machine) proposed by Suyken is one of the most important achievements of statistical learning theory; compared with the standard SVM (support vector machine) algorithm, the LSSVM follows the same principle of SRM (structural risk minimization) and has the advantage of reducing the computational complexity, improving the calculation speed and the anti-interference ability; compared to the ANNs that have attracted much attention in the nonlinear field, this algorithm is not prone to fall into local optimization and has better generalization performance, especially if there are not enough learning samples [17–19]. Some scholars had proposed to use the LSSVM to predict MEMS gyroscope random drift [20,21]. However, too many data were used according to the collection order for prediction, and there was no particularly clear data construction standard and the dimension of the input vector was high, thus leading to computational complexity.

This paper proposes to process the raw data of MEMS gyroscopes with wavelet filtering and PSR (phase space reconstruction), then model the reconstructed data based on LSSVM, and using CPSO (chaotic particle swarm optimization) to optimize the parameters of LSSVM. This algorithm can be abbreviated as CPSO-LSSVM. To the best of the authors' knowledge, it is the first report on the combination of PSR and CPSO-LSSVM for modeling and compensating the random drift of MEMS gyroscope.

The purpose of this paper is to establish a more accurate MEMS gyroscope random drift model based on the CPSO-LSSVM method; to do so, the steps of building the model are described in detail, including dealing with the original data, reconstructing the data and analyzing and modeling the reconstructed data. The comparison with BP-ANN (back propagation artificial neural network) has verified that this algorithm has a better effect.

The structure of this paper is organized as follows: The principle of BP-ANN and the proposed method, that is, CPSO-LSSVM, are clearly described in Section 2; the steps of constructing an MEMS gyro drift model are discussed in Section 3; experiments and results of BP-ANN and CPSO-LSSVM for modeling the gyroscope random drift are illustrated in Section 4. Finally, Section 5 draws the conclusion and ends the paper.

2. The Principles of Algorithms

Artificial neural networks, ANNs, and LSSVMs (least square support vector machines) are two kinds of machine learning algorithms that can achieve nonlinear mapping. Both ANNs and LSSVMs have been applied to analyze data that are not mathematically modeled easily. The principles of two algorithms are described in this section.

2.1. Back Propagation Artificial Neural Networks

Artificial neural networks (ANNs) simulating biological neural systems have a powerful learning function. The weighted sum of inputs arriving at each neuron generates an output signal by means of an activation function [22–24]. The BP-ANN (back propagation artificial neural network) is one of the most employed ANN methods and has been widely used in many applications. It does not require a

rigorous mathematic model and can obtain calibration parameters for data through a learning step [25]. Good flexibility makes it widely used to express nonlinear relationships in databases [26].

The BP-ANN consists of three types of layers which are the input layer, hidden layer and output layer, and its classical architecture is revealed in Figure 1. The nodes of input layer are used to select independent variables for estimation. The hidden layer is important and has an impact on the learning ability of the BP-ANN. The output layer is used to estimate the result depending on the independent input variables.

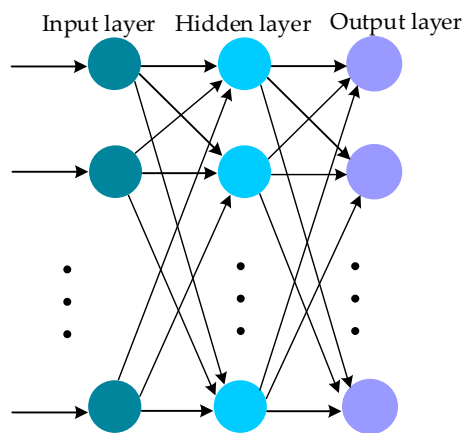


Figure 1. The classical architecture of BP-ANN (back propagation artificial neural network).

The input of neurons in the hidden layer can be described as:

$$v_i = \sum_{j=1}^{n_0} w_{ij}x_j + b_i \quad (1)$$

where v_i is the i th neuron input in the hidden layer, w_{ij} is the connection weight between the i th neuron in the input layer and the j th neuron in the hidden layer, x_j denotes the j th neuron input in the input layer while n_0 is the number of neurons in the input layer, and b_i is the threshold of the i th neuron in the hidden layer.

The sigmoid function presented in (2) is the activation function of the hidden layer:

$$g(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

where x is the independent variable.

The output of the neuron in the hidden layer can be formulated as (3) with the sigmoid function shown in (2):

$$x_i = g(v_i) \quad (3)$$

where x_i is the i th neuron output in the hidden layer, and v_i is the same as for (1).

The activation function in the output layer is the same as that in the hidden layer, and the weight and threshold of the neuron in the output layer are also constantly adjusted to approximate the desired output value.

Figure 2 is the scheme diagram of the BP-ANN illustrating the neural network architecture, where W is the connection weight of each node, b is a threshold, and g and f are the activation function shown in (2).

However, it must be pointed out that ANNs follow the principle of ERM (empirical risk minimization) and might result in over-fit toward the employed dataset [27]. In particular, when the amount of data is not too large, some algorithms must be applied to improve ANNs or other machine learning algorithms should be considered to avoid over-fit.

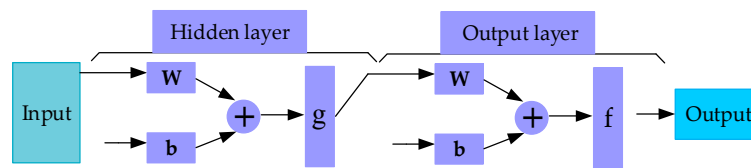


Figure 2. The schematic diagram of the BP-ANN.

2.2. LSSVM Model with CPSO

The LSSVM, one of the new soft computing learning algorithms, was developed by Suyken [17]. The LSSVM is performed by solving a linear system of equations instead of quadratic programming as for the standard SVM case [28]. Since LSSVMs have SRM of the standard SVMs and a faster computation speed, LSSVMs have been successfully applied to pattern recognition, fault diagnosis and function estimation [29–31]. The principle of LSSVMs is shown in Figure 3.

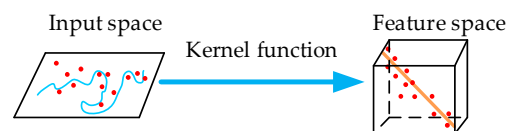


Figure 3. The principle of LSSVMs (least squares support vector machine).

Suppose a training dataset $\{x_i, y_i\}_i^n$, where x_i is the n -dimensional input vector, $x_i \in R_n$, and y_i is the responding desired output, $y_i \in R$. The training data can fit with the function represented as:

$$f(x) = w \cdot \varnothing(x) + b \quad (4)$$

where $\varnothing(x)$ is a nonlinear mapping between input space x and high-dimensional feature space, w denotes a weight vector, and b denotes a constant offset.

The w and b can be estimated through minimization of regularized risk function subjected to the equality constraint shown in (5):

$$\begin{aligned} \text{Minimize } J(w, e) &= \frac{1}{2} \|w\|^2 + \frac{1}{2} \gamma \sum_{i=1}^n (e_i^2) \\ \text{Subject to } y_i [w^T \varnothing(x_i) + b] &= 1 - e_i, \quad i = 1, 2, \dots, n \end{aligned} \quad (5)$$

where e_i is the introduced error variable, and γ is an adjustable parameter.

The corresponding Lagrange for Equation (5) is:

$$L(w, b, e; a) = J(w, e) - \sum_{i=1}^n a_i \{y_i [w^T \varnothing(x_i) + b] - 1 + e_i\} \quad (6)$$

The optimality condition of Equation (6) leads to the following linear system:

$$\begin{bmatrix} 0 & y^T \\ y & \Omega + I/\gamma \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (7)$$

where $\Omega_{ij} = \Phi(x_i, x_j)$ is the kernel function and has optional kernels such as Gaussian kernel, polynomial kernel, B-spline kernel, and so on. A Gaussian kernel is usually chosen due to its simple, efficient and reliable computing power [32]. The Gaussian kernel can be written as follows:

$$\Phi(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right) \quad (8)$$

where σ^2 represents the kernel parameter, and x_i and x_j are two independent vectors in the input space.

However, the LSSVM's parameters are uncertain for different problems and must be carefully determined because this will affect the modeling effect. It was proposed to use CPSO (chaotic particle swarm optimization) to optimize the two parameters of LSSVM. What needs to be noted is that CPSO is an improved algorithm for PSO (particle swarm optimization). Similar to GA (genetic algorithms) and evolutionary programming, PSO is also a heuristic searching method, but it does not contain complex mechanisms such as crossover or mutation [33]. The PSO algorithm is fast and suitable for parallel computing. Since chaotic mapping has deterministic, ergodic and stochastic properties, chaos mapping is introduced into PSO, which is called CPSO [34,35].

During calculation, the particle moving in space is influenced by three factors, including the particle's current velocity $v(t)$, the best point p_{id} where the particle has arrived before and optimum point of the community p_{gd} . Concurrently, the three factors are respectively assigned to a random weight. The velocity and position are updated on the basis of Equations (9) and (10):

$$v_{id}(t+1) = \omega \times v_{id}(t) + c_1 \times r_1 \times (p_{id} - z_{id}) + c_2 \times r_2 \times (p_{gd} - z_{id}) \quad (9)$$

$$z_{id}(t+1) = z_{id}(t) + v_{id}(t+1) \quad (10)$$

$$(i = 1, 2, \dots, n \quad d = 1, 2, \dots, m)$$

where c_1 and c_2 denote the learning factors, usually, $c_1 = c_2 = 2$; ω is the weight; r_1, r_2 are the random numbers within $[0, 1]$; z denotes the current position of the particle; t is the number of iterations; n and m , respectively, represent the number of particles and the dimension of the particle.

The CPSO process can be described as follows:

Step 1. Chaos Initialization: A random vector defined as $z = [z_1, z_2, \dots, z_D]$ is generated, D denotes the dimension of the variable that needs to be optimized. The range of each component of z is $[0, 1]$. Then, M components are obtained according to Equation (11).

$$\begin{aligned} z_{n+1} &= \mu \times z_n \times (1 - z_n) \\ n &= 0, 1, 2, \dots \\ \text{Subject to } &\begin{cases} \mu \in [0, 4] \\ 0 < z_n < 1 \end{cases} \end{aligned} \quad (11)$$

The chaotic interval is mapped to the range of variables based on Equation (12):

$$\begin{aligned} x_{ij} &= a_j + (b_j - a_j) \times z_{ij} \\ (i &= 1, 2, \dots, M; j = 1, 2, \dots, D) \end{aligned} \quad (12)$$

where b_j and a_j are the upper and lower limits of the optimization variables, respectively.

Then, calculating the fitness value of each particle according to the objective function shown in Equation (13):

$$f_{objective} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (13)$$

where n represents the sample size of the training dataset. The output training set is y_i and \hat{y}_i is the corresponding fitting result of y_i .

The N particles with better performance are selected as the initial solution from the initial particle swarm with M particles, and the particle velocity is generated at random.

Defining the current position of the particle is individual best, then calculating the corresponding fitness value and setting the position of the particle whose fitness value is the best is the global best point. The global best corresponds to the minimum fitness value.

Step 2. Velocity and Position Updating: The velocity and position of i th particle in the d th dimension is updated according to Equations (9) and (10), with the global best and individual best. The parameter ω is updated by the Equation (14):

$$\omega(t) = \omega_{max} - t \times \frac{\omega_{max} - \omega_{min}}{t_{max}} \quad (14)$$

where ω_{max} and ω_{min} are the maximum and minimum values of initial weights which can be set as 0.9 and 0.1, respectively, t denotes the current number of iterations, while t_{max} is the maximum number of iterations set.

Step 3. Individual and Global Best Updating: Calculating the fitness value of each particle and evaluating this particle according to its updated position. When the fitness value of this particle is less than that of individual or global best, then individual or global best will be replaced by the position of this particle.

Step 4. Stopping Criteria: If one stopping condition is satisfied then stop; otherwise go to Step 2. Based on descriptions above, the flow chart of CPSO-LSSVM is shown in Figure 4:

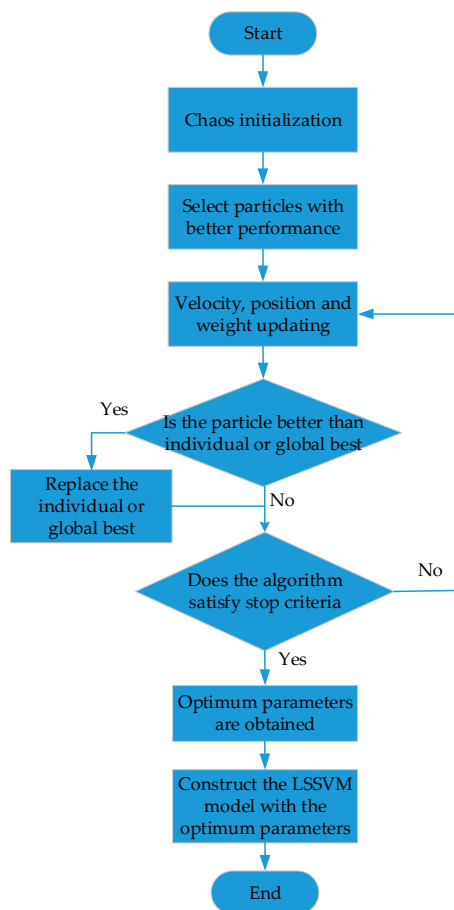


Figure 4. The flow chart of CPSO (chaotic particle swarm optimization)-LSSVM.

3. Construction of MEMS Gyroscope Drift Model

The CPSO-LSSVM method is used to model the MEMS gyroscope random drift, and is compared with BP-ANN, which is widely used in nonlinear fields. The original MEMS gyroscope data unavoidably includes noise information. Therefore, the wavelet filtering method is used to eliminate noise, and the random drift of the MEMS gyroscope is obtained for constructing a model. The random drift is a chaotic time series, and it is reconstructed by PSR since it is an efficient algorithm to

analyze chaotic time series; the dimension of the random time series is improved by embedding the one-dimensional time series into an auxiliary phase space, thus improving the prediction accuracy [3,36]. Then, the reconstructed data is modeled and analyzed. Details of the model are expressed as follows, and the main procedures are illustrated as follows and depicted in Figure 5:

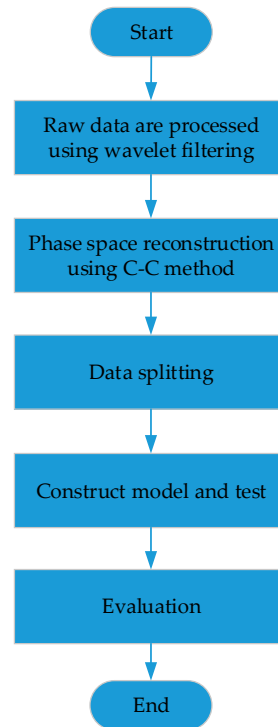


Figure 5. The flow chart of constructing a MEMS (Micro Electro Mechanical System) gyroscope model.

Step 1: Wavelet Filtering. The wavelet analysis overcomes the shortcomings of Fourier analysis in signal processing and can remove the noise while preserving the signal instantaneous dynamic characteristics, so it has a wide range of applications in signal de-noising [37,38]. The original MEMS gyroscope drift contained noise components, and the mother wavelet 'db4' and the soft threshold are adopted for de-noising.

Step 2: Phase Space Reconstruction. Regarding the random time series: $\{x(i)\}$, $i = 1, 2, \dots, N$, the reconstructed sequence is obtained through PSR method as follows:

$$X = \begin{bmatrix} x_1 & x_{1+\tau} & \dots & x_{1+(m-1)\tau} \\ x_2 & x_{2+\tau} & \dots & x_{2+(m-1)\tau} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-(m-1)\tau-1} & x_{n-(m-2)\tau-1} & \dots & x_{n-1} \end{bmatrix} \quad (15)$$

$$Y = \begin{bmatrix} x_{2+(m-1)\tau} & x_{3+(m-1)\tau} & \dots & x_n \end{bmatrix}^T \quad (16)$$

where m represents embedding dimension and τ denotes delay time.

The delay time window, τ_w , shown in Equation (17) is introduced as follows:

$$\tau_w = (m - 1)\tau \quad (17)$$

The C-C method is used to determine optimal parameters τ_w and τ since this method is robust, relatively simple and not computationally demanding [39]. The following three formulas are very important for determining the parameters:

$$\bar{S}(t) = \frac{1}{16} \sum_{m=2}^5 \sum_{j=1}^4 S(m, r_j, t) \quad (18)$$

$$\Delta\bar{S}(t) = \frac{1}{4} \sum_{m=2}^5 \Delta S(m, t) \quad (19)$$

$$S_{cor}(t) = \Delta\bar{S}(t) + |\bar{S}(t)| \quad (20)$$

where $m = 2, 3, 4, 5$, $r_j = \frac{i\sigma}{2}$, $i = 1, 2, 3, 4$ and $S(m, r_j, t)$ and $\Delta S(m, t)$ are two functions whose definitions have been elaborated in the paper [39].

Looking for the first zero point of $\bar{S}(t)$ or the first local minimum of $\Delta\bar{S}(t)$ to find the first locally optimal time for independence of the data gives the delay time τ . Meanwhile, by simply looking for the minimum of $S_{cor}(t)$ and then the delay time window, τ_w can be obtained [37,39].

Step 3: Data Splitting. After the noise removal and data reconstruction, the MEMS gyroscope random drift sequence is split into the training dataset and the testing dataset. To eliminate the impact of amplitude and improve convergence performance, data normalization is needed to be done, based on Equation (21) as follows:

$$x'_k = \frac{(x'_{max} - x'_{min})(x_k - x_{min})}{x_{max} - x_{min}} \quad (21)$$

where x_{max} is the maximum value of the variable and x_{min} represents the minimum value, with x'_{max} set as 1 and x'_{min} is set as -1 . The range of variables is normalized within $[-1, 1]$ after preprocessing.

Step 4: Model Construction and Test. This step requires a training dataset to train the model, and uses a testing dataset to verify the prediction accuracy of the proposed model. The algorithms of different models have been described in detail in Section 2 of this paper.

Step 5: Evaluation. The prediction error indexes, namely MAE (mean absolute error), RMSE (root mean square error), and ARE (average relative error), are applied for evaluating prediction accuracy. The application of these indices is very common, so are only briefly described here:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_{pi} - y_i| \quad (22)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_{pi} - y_i)^2}{n}} \quad (23)$$

$$ARE = \frac{1}{n} \sum_{i=1}^n \frac{|y_{pi} - y_i|}{y_i} \quad (24)$$

where n is the size of prediction data, y_{pi} denotes the predicting value, and y_i represents the actual value.

4. Results and Discussion

4.1. Experiment Setup

The hardware utilized in the trials was an MEMS IMU (inertial measurement unit) designed by this laboratory. The MEMS IMU was placed on the three-axis turntable. Preheating the MEMS IMU lasted for 20 min, then 1500 s of data were collected, and the sampling rate was 10 Hz. The output data of the X-axis gyroscope were analyzed and modeled in this paper. The experiment environment and MEMS IMU are shown in Figure 6.

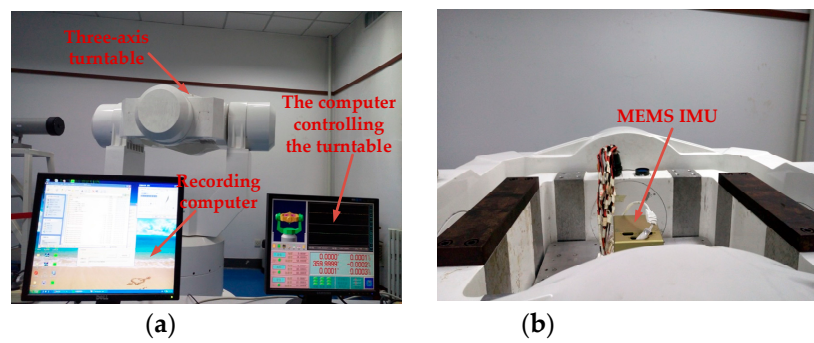


Figure 6. (a) Experimental environment; (b) the MEMS IMU (inertial measurement unit) used in the experiment.

4.2. Data Preprocessing

Section 3 notes the task of data preprocessing was to perform wavelet filtering and phase space reconstruction for the collected data of MEMS gyroscope. Following deterministic error compensation, the time series of random drift was obtained. First wavelet filtering with mother wavelet 'db4' and the soft threshold was used to remove most of the noise of original data under three scales. Figure 7 shows the de-noised data contained the essential features of random drift.

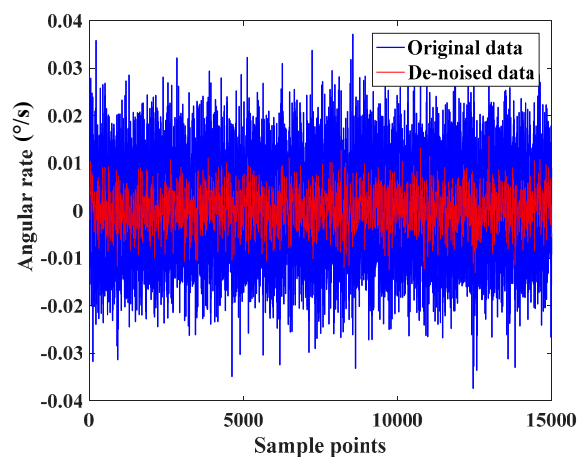


Figure 7. Original data and de-noised data of the MEMS gyroscope.

The next step was to reconstruct the filtered series. The C-C method was used to determine the delay time τ and the time window τ_w . The related equations are presented as Equations (18)–(20).

Figure 8 shows the time delay τ corresponding to the first local minimum of $\Delta\bar{S}(t)$ was 10. The time window τ_w corresponding to the minimum value of $S_{cor}(t)$ was 20. According to the Equation (17), $m = 3$ was easily obtained. Then, the reconstructed mathematical matrixes were obtained as follows:

$$X = \begin{bmatrix} x_1 & x_{11} & x_{21} \\ x_2 & x_{12} & x_{22} \\ \vdots & \vdots & \vdots \\ x_{n-21} & x_{n-11} & x_{n-1} \end{bmatrix} \quad (25)$$

$$Y = \begin{bmatrix} x_{22} & x_{23} & \dots & x_n \end{bmatrix}^T \quad (26)$$

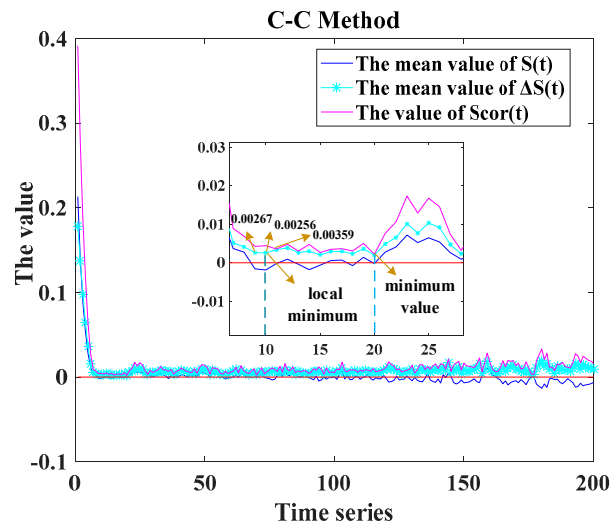


Figure 8. Determining the delay time and the time window by using the C-C method.

4.3. Comparing the Effects of Modeling

To verify the applicability of the proposed model, the reconstructed data (25) and (26) were normalized by using Equation (21) at first, then the data were divided into three groups. Each group then was divided into two data sets: the training dataset (80%), and the testing dataset (20%). Furthermore, the training set was further divided into an input-training set and an output-training set. Detailed descriptions were as follows.

The first group containing 5000 sets of data was used for establishment and validation of BP-ANN and CPSO-LSSVM models at first. The training dataset contained 4000 sample points, and the other 1000 sample points were for testing. The BP-ANN method, with 10 neurons in hidden layer, was used for the training dataset. The parameters of LSSVM, which had been acquired on the basis of CPSO, were $\gamma = 29.358$, $\sigma^2 = 83.162$. The fitting effect of the training dataset was shown in Figure 9.

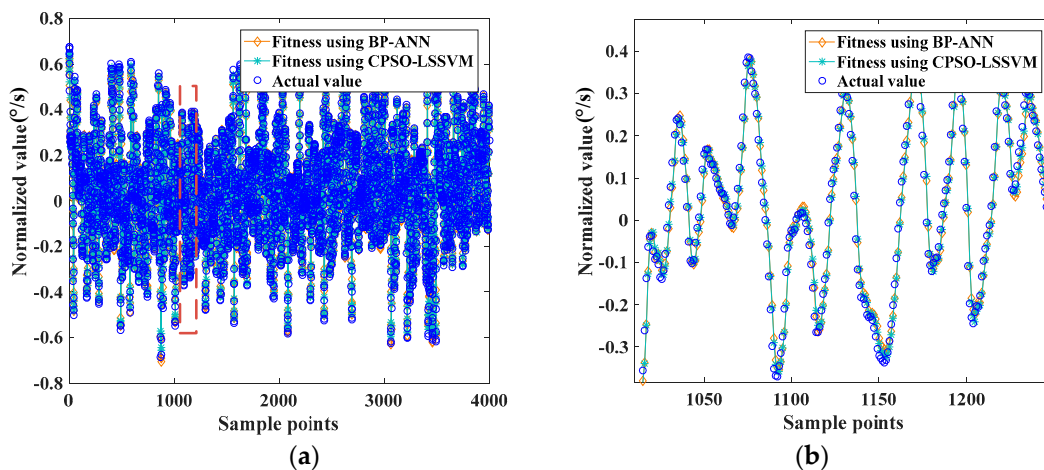


Figure 9. (a) The fitting effect of the training dataset using BP-ANN and CPSO-LSSVM; (b) partially enlarged detail of (a).

The two well-trained models were used to predict the output of the testing dataset, and the predicted values were compared with the actual values, which are shown in Figure 10:

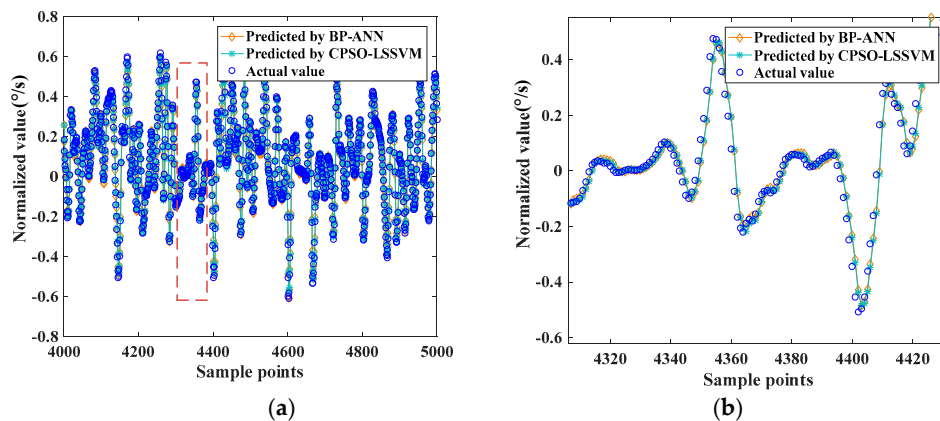


Figure 10. (a) The effect of prediction of the testing dataset using BP-ANN and CPSO-LSSVM; (b) partially enlarged detail of (a).

Statistical parameters, such as MAE, RMSE and ARE, depicted in Equations (22)–(24), were used to evaluate the prediction accuracy. Table 1 depicts the statistical analysis of the comparison results of the two models.

Table 1. The statistical analysis of BP-ANN and CPSO-LSSVM. MAE (mean absolute error); RMSE (root mean square error); ARE (average relative error).

Model	MAE (°/s)	RMSE (°/s)	ARE
BP-ANN	0.0421	0.0554	11.10%
CPSO-LSSVM	0.0099	0.0263	8.86%

Figures 9 and 10 show that BP-ANN and CPSO-LSSVM both had a good effect on fitting the random drift, but the CPSO-LSSVM was better according to Table 1. The training time of BP-ANN was about 22.301 s, while the CPSO-LSSVM was about 14.556 s in the first group, so the latter had a better computing speed. It further demonstrated the applicability of the proposed method.

Furthermore, the reproducibility of both methods was evaluated by building models for the other two groups. The results of the three groups were vividly presented in Figure 11. Comparing the prediction results of the other two groups reaches conclusions similar to that which the first group obtained.

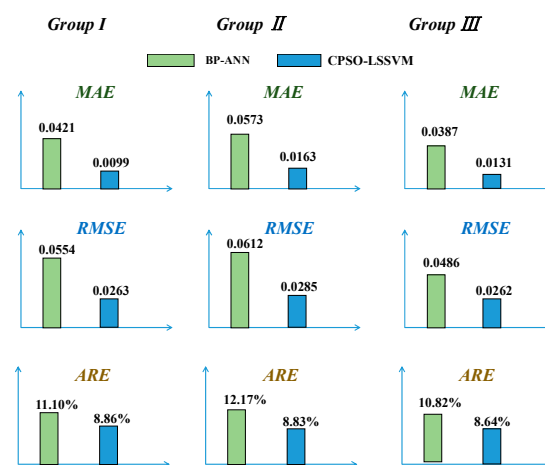


Figure 11. The statistical results of the three groups.

Based on the results above, the proposed CPSO-LSSVM method outperforms the BP-ANN. The better prediction performance may be due to the tendency for the CPSO to enjoy certainty, ergodicity and stochastic property, and to have a good global search capability; LSSVM has the characteristics of SRM and anti-interference ability; CPSO-LSSVM combines both the advantages of CPSO and LSSVM.

The compensation of MEMS gyroscope drift then was done based on the prediction values of CPSO-LSSVM method. The anti-normalization of prediction data were used to compensate the random drift of MEMS gyroscope. Comparing the testing data in the first group before and after the compensation, the results showed that the compensative effect was remarkably effective. The standard deviation of the random drift after compensation was $0.00065^\circ/s$, while it was $0.00354^\circ/s$ without compensation. Figure 12 shows the random drift of the MEMS gyroscope before and after the compensation.

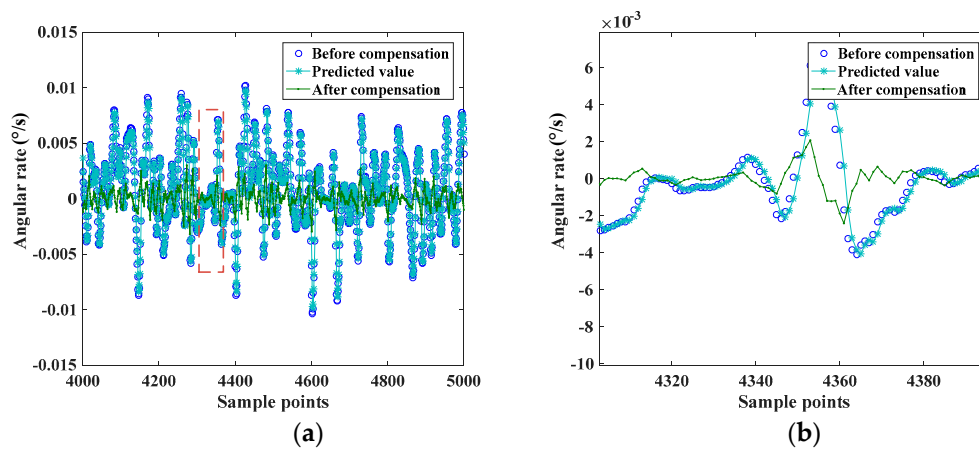


Figure 12. (a) The change of random drift before and after compensation; (b) partially enlarged detail of (a).

The effect of compensating the random drift using CPSO-LSSVM method for the three groups was shown in Table 2, where the evaluation index was the standard deviation of the random drift.

Table 2. The standard deviation of the random drift before and after compensation.

Group	I	II	III
Before compensation ($^\circ/s$)	0.00354	0.00412	0.00328
After compensation ($^\circ/s$)	0.00065	0.00072	0.00053

Observing Table 2, it was clear that the standard deviation of the random drift had been decreased significantly after compensation. The results further demonstrated the applicability and validity of the CPSO-LSSVM method. This method was a feasible and satisfactory way to establish the model for MEMS gyroscope random drift.

5. Conclusions

The modeling of MEMS gyroscope random drift is a hot research topic since an accurate model is beneficial to improve the accuracy of MEMS gyroscopes. The key contribution of this paper is to reconstruct the random drift data of the MEMS gyroscope with PSR using the C-C method, and then to analyze the reconstructed data by the BP-ANN and CPSO-LSSVM methods. Based on the results and analyses, the following conclusions can be drawn: (a) Through a comparison of the results from the above two analysis methods, the statistical indicators, including MAE, RMSE, and ARE, for the

testing dataset indicate that the proposed method has a better prediction precision; (b) the availability and effectiveness of CPSO-LSSVM method are demonstrated by an obvious decrease in the standard deviation of the random drift after compensation, which is shown in Table 2; (c) PSR plays an important role in the modeling of MEMS gyroscope random drift since it can reduce the dimension of input vector during modeling, thus reducing computation cost and complexity; (d) the CPSO can perform more powerful searching capability for parameters to construct the proposed model, and it can also be used for function optimization, data mining, and so on; (e) the combination and improvement of intelligent algorithms can lead to a better algorithm, and the CPSO-LSSVM method presented in this paper validates this point; (f) whether the modeling accuracy and compensation effectiveness of the method are affected by the different sensing mechanisms should be further studied. Considering using an algorithm to compensate for random drift of MEMS gyroscope is not enough, as well as from the structure and design standpoint to improve the accuracy of the MEMS gyroscope.

To summarize, the results show the better prediction capacity of the proposed model. It is believed that this algorithm can be regarded as a reliable method for modeling and compensating the MEMS gyroscope random drift. This method is expected to be applied in the field of north-seeking and short-time navigation based on MEMS gyroscopes. Additionally, the method proposed in this paper also implies that this method can be a new way for pedestrian step estimation, pattern recognition and many other fields. Better algorithms and more methods need to be studied further, and more work needs to be done in the future.

Author Contributions: Haifeng Xing wrote the paper and analyzed the data; Bo Hou designed the experiments; Zhihui Lin helped to process the data; Meifeng Guo gave some valuable suggestions and supervised the work.

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

References

1. Wang, X.Y.; Meng, X.Y. Research on Time-series Modeling and Filtering Methods for MEMS Gyroscope Random Drift Error. In Proceedings of the Materials Science and Engineering Conference Series, Harbin, China, 5 September 2017; p. 012005.
2. Li, B.-W.; Yao, D.-Y. Low-cost MEMS IMU navigation positioning method for land vehicle. *J. Chin. Inert. Technol.* **2014**, *22*, 719–723.
3. Wang, S.; Deng, Z.; Yin, G. An Accurate GPS-IMU/DR Data Fusion Method for Driverless Car Based on a Set of Predictive Models and Grid Constraints. *Sensors* **2016**, *16*, 280. [[CrossRef](#)] [[PubMed](#)]
4. Chia, J.; Low, K.; Goh, S.; Xing, Y. A low complexity Kalman filter for improving MEMS based gyroscope performance. In Proceedings of the Aerospace Conference, Big Sky, MT, USA, 5–12 March 2016; pp. 1–7.
5. Cao, H.; Li, H.; Kou, Z.; Shi, Y.; Tang, J.; Ma, Z.; Shen, C.; Liu, J. Optimization and experimentation of dual-mass MEMS gyroscope quadrature error correction methods. *Sensors* **2016**, *16*, 71. [[CrossRef](#)] [[PubMed](#)]
6. Hsu, Y.L.; Chou, P.H.; Kuo, Y.C. Drift modeling and compensation for MEMS-based gyroscope using a Wiener-type recurrent neural network. In Proceedings of the IEEE International Symposium on Inertial Sensors and Systems, Kauai, HI, USA, 28–30 March 2017; pp. 39–42.
7. Liu, Y.; Niu, R.; Wang, C.; Wang, L. Random Error Model and Compensation of MEMS Gyroscope Based on BP Neural Network. *Int. J. Digit. Content Technol. Its Appl.* **2013**, *7*, 284–290.
8. Ansari, A.; Bakar, A.A. A Comparative Study of Three Artificial Intelligence Techniques: Genetic Algorithm, Neural Network, and Fuzzy Logic, on Scheduling Problem. In Proceedings of the Artificial Intelligence with Applications in Engineering and Technology (ICAJET), Kota Kinabalu, Malaysia, 3–5 December 2014; pp. 31–36.
9. Chen, M.M.; Gao, G.W. Research on MEMS gyroscope random error compensation algorithm based on ARMA model. *Appl. Mech. Mater.* **2014**, *602–605*, 891–894. [[CrossRef](#)]
10. Yuan, J.; Yuan, Y.; Liu, F.; Pang, Y.; Lin, J. An improved noise reduction algorithm based on wavelet transformation for MEMS gyroscope. *Front. Optoelectron.* **2015**, *8*, 413–418. [[CrossRef](#)]
11. Shi, Y.S.; Gao, Z.F. Study on MEMS gyro signal de-noising based on improved wavelet threshold method. *Appl. Mech. Mater.* **2013**, *433–435*, 1558–1562.

12. Zha, F.; Xu, J.; Li, J.; He, H. IUKF neural network modeling for FOG temperature drift. *J. Syst. Eng. Electron.* **2013**, *24*, 838–844. [[CrossRef](#)]
13. Chong, S.; Rui, S.; Jie, L.; Xiaoming, Z.; Jun, T.; Yunbo, S.; Jun, L.; Huiliang, C. Temperature drift modeling of MEMS gyroscope based on genetic-Elman neural network. *Mech. Syst. Signal Process.* **2016**, *72*, 897–905. [[CrossRef](#)]
14. Xu, D.; Yang, Z.; Zhao, H.; Zhou, X. A temperature compensation method for MEMS accelerometer based on LM_BP neural network. In Proceedings of the 2016 IEEE Sensors, Orlando, FL, USA, 30 October–3 November 2016; pp. 1–3.
15. Fei, J.; Wu, D. Adaptive control of MEMS gyroscope using fully tuned RBF neural network. *Neural Comput. Appl.* **2017**, *28*, 695–702. [[CrossRef](#)]
16. Deng, F.; Guo, S.; Zhou, R.; Chen, J. Sensor multifault diagnosis with improved support vector machines. *Trans. Autom. Sci. Eng.* **2017**, *14*, 1053–1063. [[CrossRef](#)]
17. Suykens, J.A.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [[CrossRef](#)]
18. You, H.; Ma, Z.; Tang, Y.; Wang, Y.; Yan, J.; Ni, M.; Cen, K.; Huang, Q. Comparison of ANN (MLP), ANFIS, SVM, and RF models for the online classification of heating value of burning municipal solid waste in circulating fluidized bed incinerators. *Waste Manag.* **2017**, *68*, 186–197. [[CrossRef](#)] [[PubMed](#)]
19. Sun, T.; Liu, J. Predicting MEMS gyroscope's random drifts using LSSVM optimized by modified PSO. In Proceedings of the Guidance, Navigation and Control Conference (CGNCC), Nanjing, China, 12–14 August 2016; pp. 146–149.
20. Bo, R.; Huan, L. On forecast modeling of MEMS gyroscope random drift error. In Proceedings of the 2015 34th Chinese Control Conference (CCC), Hangzhou, China, 28–30 July 2015; pp. 4563–4567.
21. Qin, W.-W.; Zheng, Z.-Q.; Liu, G.; Wang, L.-X. Modeling method of gyroscope's random drift based on wavelet analysis and LSSVM. *J. Chin. Inertial Technol.* **2008**, *16*, 721–724.
22. Torrecilla, J.S.; Deetlefs, M.; Seddon, K.R.; Rodríguez, F. Estimation of ternary liquid–liquid equilibria for arene/alkane/ionic liquid mixtures using neural networks. *Phys. Chem. Chem. Phys.* **2008**, *10*, 5114–5120. [[CrossRef](#)] [[PubMed](#)]
23. Bhatti, M.S.; Kapoor, D.; Kalia, R.K.; Reddy, A.S.; Thukral, A.K. RSM and ANN modeling for electrocoagulation of copper from simulated wastewater: Multi objective optimization using genetic algorithm approach. *Desalination* **2011**, *274*, 74–80. [[CrossRef](#)]
24. Zhao, G.; Wang, H.; Liu, G.; Wang, Z. Optimization of Stripping Voltammetric Sensor by a Back Propagation Artificial Neural Network for the Accurate Determination of Pb(II) in the Presence of Cd(II). *Sensors* **2016**, *16*, 1540. [[CrossRef](#)] [[PubMed](#)]
25. Suah, F.B.M.; Ahmad, M.; Taib, M.N. Optimisation of the range of an optical fibre pH sensor using feed-forward artificial neural network. *Sens. Actuators B Chem.* **2003**, *90*, 175–181. [[CrossRef](#)]
26. Bade, R.; Bijlsma, L.; Miller, T.H.; Barron, L.P.; Sancho, J.V.; Hernández, F. Suspect screening of large numbers of emerging contaminants in environmental waters using artificial neural networks for chromatographic retention time prediction and high resolution mass spectrometry data analysis. *Sci. Total Environ.* **2015**, *538*, 934–941. [[CrossRef](#)] [[PubMed](#)]
27. Torrecilla, J.S.; García, J.; Rojo, E.; Rodríguez, F. Estimation of toxicity of ionic liquids in Leukemia Rat Cell Line and Acetylcholinesterase enzyme by principal component analysis, neural networks and multiple lineal regressions. *J. Hazard. Mater.* **2009**, *164*, 182–194. [[CrossRef](#)] [[PubMed](#)]
28. Alharbi, N.; Hassani, H. A new approach for selecting the number of the eigenvalues in singular spectrum analysis. *J. Frankl. Inst.* **2016**, *353*, 1–16. [[CrossRef](#)]
29. Suykens, J.; Lukas, L.; Van Dooren, P.; De Moor, B.; Vandewalle, J. Least squares support vector machine classifiers: A large scale algorithm. In Proceedings of the European Conference on Circuit Theory and Design, ECCTD, Stresa, Italy, 4–6 September 1999; pp. 839–842.
30. Long, B.; Huang, J.; Tian, S. Least squares support vector machine based analog-circuit fault diagnosis using wavelet transform as preprocessor. In Proceedings of the International Conference on Communications, Circuits and Systems, Fujian, China, 25–27 May 2008; pp. 1026–1029.
31. Lin, W.-M.; Tu, C.-S.; Yang, R.-F.; Tsai, M.-T. Particle Swarm Optimisation Aided Least-Square Support Vector Machine for Load Forecast with Spikes. *IET Gener. Transm. Distrib.* **2016**, *10*, 1145–1153. [[CrossRef](#)]

32. Zhang, X.; Wang, J.; Zhang, K. Short-term electric load forecasting based on singular spectrum analysis and support vector machine optimized by Cuckoo search algorithm. *Electr. Power Syst. Res.* **2017**, *146*, 270–285. [[CrossRef](#)]
33. Pan, A.; Zhou, J.; Zhang, P.; Lin, S.; Tang, J. Predicting of Power Quality Steady State Index Based on Chaotic Theory Using Least Squares Support Vector Machine. *Power* **2017**, *3*, 5. [[CrossRef](#)]
34. Bian, Q.; Wang, X.; Xie, R.; Li, T.; Ma, T. Small scale helicopter system identification based on Modified Particle Swarm Optimization, Guidance. In Proceedings of the 2016 IEEE Chinese Navigation and Control Conference (CGNCC), Nanjing, China, 10 March 2016; pp. 182–186.
35. Lagos-Eulogio, P.; Seck-Tuoh-Mora, J.C.; Hernandez-Romero, N.; Medina-Marin, J. A new design method for adaptive IIR system identification using hybrid CPSO and DE. *Nonlinear Dyn.* **2017**, *88*, 2371–2389. [[CrossRef](#)]
36. Chuanwen, J.; Bompard, E. A hybrid method of chaotic particle swarm optimization and linear interior for reactive power optimisation. *Math. Comput. Simul.* **2005**, *68*, 57–65. [[CrossRef](#)]
37. Tao, H.Y.C. Phase-space reconstruction technology of chaotic attractor based on CC method. *J. Electron. Meas. Instrum.* **2012**, *5*, 010.
38. Liu, F.; Liu, F.; Wang, W.; Xu, B. MEMS gyro's output signal de-noising based on wavelet analysis. In Proceedings of the International Conference on Mechatronics and Automation, Harbin, China, 5–9 August 2007; pp. 1288–1293.
39. Kim, H.S.; Eykholt, R.; Salas, J. Nonlinear dynamics, delay times, and embedding windows. *Phys. D Nonlinear Phenom.* **1999**, *127*, 48–60. [[CrossRef](#)]



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).