

Article

# Performance Analysis of a Deep Simple Recurrent Unit Recurrent Neural Network (SRU-RNN) in MEMS Gyroscope De-Noising

Changhui Jiang <sup>1,2</sup> , Shuai Chen <sup>1,\*</sup>, Yuwei Chen <sup>2</sup>, Yuming Bo <sup>1</sup>, Lin Han <sup>1</sup>, Jun Guo <sup>1</sup>, Ziyi Feng <sup>2</sup> and Hui Zhou <sup>3</sup>

<sup>1</sup> School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China; changhui.jiang1992@gmail.com (C.J.); byuming@mail.njust.edu.cn (Y.B.); soochow\_njust@sina.com (L.H.); guojun1136@163.com (J.G.)

<sup>2</sup> Centre of Excellence in Laser Scanning Research, Finnish Geospatial Research Institute (FGI), Geodeetinrinne 2, FI-02431 Kirkkonummi, Finland; yuwei.chen@nls.fi (Y.C.); ziyi.feng@nls.fi (Z.F.)

<sup>3</sup> Electronic Information School, Wuhan University, 129 Luoyu Road, Wuhan 430079, China; zhouhui@whu.edu.cn

\* Correspondence: c1492@163.com; Tel.: +86-138-139-15826

Received: 27 October 2018; Accepted: 14 December 2018; Published: 17 December 2018



**Abstract:** Microelectromechanical System (MEMS) Inertial Measurement Unit (IMU) is popular in the community for constructing a navigation system, due to its small size and low power consumption. However, limited by the manufacturing technology, MEMS IMU experiences more complicated noises and errors. Thus, noise modeling and suppression is important for improving accuracy of the navigation system based on MEMS IMU. Motivated by this problem, in this paper, a deep learning method was introduced to MEMS gyroscope de-noising. Specifically, a recently popular Recurrent Neural Networks (RNN) variant Simple Recurrent Unit (SRU-RNN) was employed in MEMS gyroscope raw signals de-noising. A MEMS IMU MSI3200 from MT Microsystem Company was employed in the experiments for evaluating the proposed method. Following two problems were furtherly discussed and investigated: (1) the employed SRU with different training data length were compared to explore whether there was trade-off between the training data length and prediction performance; (2) Allan Variance was the most popular MEMS gyroscope analyzing method, and five basic parameters were employed to describe the performance of different grade MEMS gyroscope; among them, quantization noise, angle random walk, and bias instability were the major factors influencing the MEMS gyroscope accuracy, the compensation results of the three parameters for gyroscope were presented and compared. The results supported the following conclusions: (1) considering the computation brought from training dataset, the values of 500, 3000, and 3000 were individually sufficient for the three-axis gyroscopes to obtain a reliable and stable prediction performance; (2) among the parameters, the quantization noise, angle random walk, and bias instability performed 0.6%, 6.8%, and 12.5% improvement for X-axis gyroscope, 60.5%, 17.3%, and 34.1% improvement for Y-axis gyroscope, 11.3%, 22.7%, and 35.7% improvement for Z-axis gyroscope, and the corresponding attitude errors decreased by 19.2%, 82.1%, and 69.4%. The results surely demonstrated the effectiveness of the employed SRU in this application.

**Keywords:** microelectromechanical systems; inertial measurement unit; simple recurrent unit; deep learning

## 1. Introduction

Recently, Positioning, Navigation and Timing (PNT) information is essential for many applications, for example, smart mobile devices. Global Navigation Satellite System (GNSS) is the most widely used PNT provider, since it is easy to access the satellites signals all over the world [1–3]. A cheap receiver in a chip manner is sufficient for many applications, for instance, vehicle, smart phone, or the shared bicycle [1–6]. Narrowly speaking, GNSS refers to the United States of America (USA) Global Positioning System (GPS), European Galileo satellite navigation system, China BeiDou satellite navigation system (constructing), and Russia GLONASS navigation system [4–6], they aim to provide PNT service covering the earth for both civil and military users; broadly speaking, other regional satellite navigation systems are also included, for instance, Japan Quasi-Zenith Satellite System (QZSS) and Indian Regional Navigation Satellite System (IRNSS) [4–6]. With more and more satellites available in orbit broadcasting navigation signals, the user is able to make full use of more satellites signals in navigation solutions determination, which efficiently improves the reliability and continuity of the navigation solutions [4–6]. However, under some signal challenging or degraded environments, for instance, urban canyon and dense forests, the satellites signals are weak or even blocked, which makes it hard to output ideal or precise navigation solutions for the GNSS standalone navigation system [5–8]. Researchers have been devoted to overcome this problem and improve the performance under these signals degrading conditions [7–10]. Basically, there are two popular approaches in the navigation community. The first approach is to develop highly sensitive GNSS receiver for weak signals tracking, specifically high sensitive carrier tracking loop [11]. Abundant results have been published on this topic, including Vector Tracking Loop (VTL), Multiple Vector Tracking Loop (MVTL), and some other advanced carrier tracking loops design [11–18]. However, it is not sufficient for signal outage, no navigation solutions will be output, while GNSS signals are temporarily blocked by the buildings or something else [19,20].

The second approach is to integrate GNSS with other sensors, for instance, the Inertial Navigation System (INS), Visual navigation system, or LiDAR navigation system [21–23]. GNSS/INS integration system is the most popular for providing navigation solutions, including position, velocity, and attitude information. Inertial Navigation System (INS) is a self-contained navigation system which generates continuous navigation solutions by processing the measurements from Inertial Measurement Unit (IMU). IMU usually contains three orthogonal gyroscope and three orthogonal accelerometers, in which the gyroscope measures the angle rate and the accelerometer collects the acceleration motion [24–26]. When compared with GNSS, the navigation solutions updating frequency is higher, while the INS is usually 400 Hz or 200 Hz, and the updating frequency of the GNSS solutions is commonly 1–10 Hz [21–25]. However, the GNSS is capable of providing precise location with well accessible signals, while the errors of the INS diverges over time, due to the various unavoidable noises contained in the raw measurements or signals of the employed gyroscopes and accelerometers. The advantage of the GNSS/INS integration navigation system is that the INS can still provide the navigation solutions, while the GNSS fails to generate navigation solutions. Therefore, it is of great value to improve the INS accuracy during GNSS signal outage.

Conventional fiber optic or laser gyroscope is of large size with high cost and it is not proper or suitable for certain applications, especially handheld or portable devices. Microelectromechanical systems (MEMS) IMU is increasingly popular recently, which is manufactured using MEMS technology. MEMS IMU has much smaller size and lower cost, and it has been widely used in mobile or handheld devices, vehicle navigation and weapons guidance [26–35]. However, compared with the highly accurate laser or fiber optic IMU, MEMS IMU usually experiences more complicated noises [26–32]. Scholars are always devoted to model the noises and compensate the divergence errors to improve the accuracy of MEMS IMU based navigation system, which can broaden the MEMS IMU applications furtherly [27–35]. As illustrated in previous papers, the MEMS IMU noise modelling or analysis method can be classified to the statistical method and Artificial Intelligence (AI) method [33–36]. The statistical methods include Auto Regressive Moving Average (ARMA) and Allan Variance (AV),

especially the Allan Method has been widely used to analyze and describe the composition of the gyroscope or the accelerometer noise contained in the output raw signals [33–36]. Generally, the five basic description parameters are termed as: quantization noise, angle random walk, bias instability, rate random walk, and rate ramp [33–36]. The other method is the Artificial Intelligence, which refers to Support Vector Machine (SVM) and Neural Networks (NN). SVM and various Neural Networks are employed in MEMS IMU de-noising and they have been evaluated by many researchers [33–36].

For both statistical or AI methods, gyroscope or accelerometer raw signals are treated as time series, and a model is described or learned to compensate the errors caused by the noises. Performance of the statistical method is limited by the fixed model parameters, and the traditional AI method has restricted learning capacity which is determined by the structure and working principles [26–36]. Recently, Deep Learning (DL) gains a boom in various applications, which has a better learning capacity than conventional SVM or neural networks [37,38]. Deep Recurrent Neural Networks (RNN) is specifically for processing time series data and has been demonstrated effectively in this kind of application [38–40]. In our previous paper, a Long Short Term Memory Unit (LSTM) (a variant of RNN) was evaluated in MEMS gyroscope de-noising, and experiments included a comparison of LSTM-RNN with ARMA, single-layer LSTM, and multi-layer LSTM [38]. Limitations of the proposed algorithms exposed in the paper were listed as following:

- (1) Training data length is fixed and not long enough, it might be meaningful to explore the influence of the training data length on the deep RNN performance; and,
- (2) Only Standard Deviation of the de-noised signals were presented and compared, but no detailed or further analysis of compensation, which could be a support of selecting proper neural networks for each MEMS IMU.

In this paper, an advanced variant of RNN Simple Recurrent Unit (SRU) was investigated in MEMS IMU based navigation system for accuracy improvement. When compared with LSTM, SRU has less simple structure with faster training speed, specifically; the SRU has less parameter that need to be determined during training procedure [39,40]. We think the extensions or contributions of this paper are as follows:

- (1) Influence of training dataset length on the SRU-RNN prediction were investigated and explored, it might be meaningful for reducing computation load;
- (2) Compensation degrees of the major noise parameters describing MEMS IMU performance were presented and compared, which might support the selection of proper or suitable RNN variants for MEMS IMU de-noising;
- (3) SRU was firstly employed in this application; the results could be compared with LSTM presented in our previous paper for selecting proper RNN in MEMS IMU de-noising.

Reminder of this paper is organized as: (1) the second section gives the basic mathematical equations and the information flow of the popular SRU-RNN; (2) in the next section, the experiments results and comparisons are presented to support the conclusions; (3) final sections include the conclusion, discussion, and reference.

## 2. Method

In this section, the basic structure and mathematical equations of the Simple Recurrent Unit Recurrent Neural Networks (SRU-RNN) are introduced and described. This section is divided into two parts: (1) the architecture and mathematical equations are listed in Section 2.1; (2) the implementation and the deep SRU-RNN working flow are given in detail in Section 2.2.

### 2.1. Simple Recurrent Unit

Simple Recurrent Unit (SRU) is the recently proposed variant of RNN, and the SRU has a more concise structure for accelerating the training procedure [39]. When compared with other RNN variants,

for instance, Long Short Term Unit (LSTM) and Grated Recurrent Unit (GRU), the SRU has faster training speed that is brought by its unique structure [39,40]. Figure 1 shows the basic architecture of the employed SRU, and it is constructed based on the “gate” structure, which also is composed of LSTM and GRU. In a single layer SRU illustrated as Figure 1, it is usually has two essential components, which are termed as “light recurrence” and “high network”. The light recurrence component reads the input vector  $x_t$  and calculates the sequence of the state  $c_t$ . This module captures the sequential information [39,40]. The light recurrent procedure can be summarized as the following Equations (1)–(3):

$$\tilde{x}_t = \mathbf{W}x_t \quad (1)$$

$$f_t = \sigma(\mathbf{W}_f x_t + \mathbf{b}_f) \quad (2)$$

$$c_t = f_t \odot c_{t-1} + (1 - f_t) \odot (\mathbf{W}x_t) \quad (3)$$

Where,  $\mathbf{W}_f$ ,  $\mathbf{W}$ , and  $\mathbf{b}_f$  are the parameter matrices, which will be determined through the training process.  $\sigma(\cdot)$  is a sigmoid function and the  $f_t$  is output of this function ranging from 0 to 1.  $\odot$  is the point-wise multiplication operation.

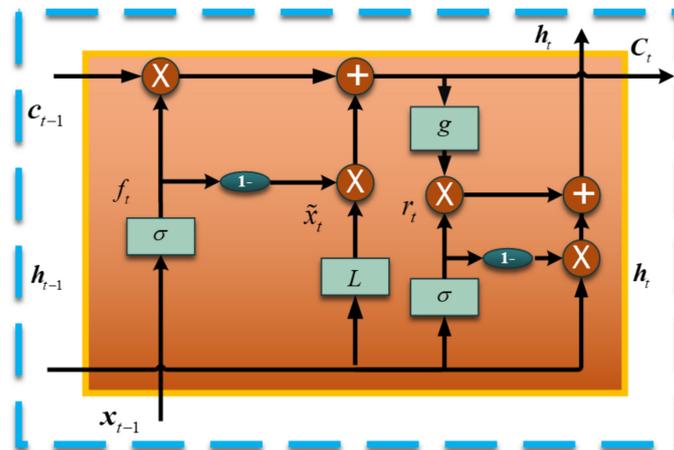


Figure 1. Basic structure of the Simple Recurrent Unit (SRU) Working flow.

As illustrated in Figure 1, the  $f_t$  controls the information flow and the current state  $c_t$  is determined by adaptively averaging the previous state  $c_{t-1}$  and the current input vector according to  $f_t$ . Especially, the SRU is different in using the previous state  $c_{t-1}$ . Traditionally, each dimension of  $c_t$  and  $f_t$  depends on all entries of  $c_{t-1}$ , and the computation has to wait until  $c_{t-1}$  is fully computed. In the SRU, a point-wise multiplication  $\odot$  is employed to hence the parallelization (seen as Equations (1) and (2)).

The second component of SRU is the highway network, which is employed to facilitate gradient-based training of deep networks [39,40]. A reset ‘gate’  $r_t$  is used to combine the input vector  $x_t$  and the current state  $c_t$  from the light recurrence [39,40]. The highway network equations are as following:

$$r_t = \sigma(\mathbf{W}_r x_t + \mathbf{b}_r) \quad (4)$$

$$h_t = r_t \odot g(c_t) + (1 - r_t) \odot x_t \quad (5)$$

Where,  $\mathbf{W}_r$  and  $\mathbf{b}_r$  are also the parameters learned by the training procedure. Especially,  $(1 - r_t) \odot x_t$  is a skip connection allowing the gradient to directly propagate to the previous layer, which has shown to improve the scalability [39,40].

## 2.2. Deep SRU-RNN Implementation

Figure 1 is just a single-layer SRU unit and Figure 2 shows the information flow of the two SRU units. The cell state is conveyed to the next SRU unit and the next SRU combines it with the input vector to decide the outputs. Basically, a sequence SRU decides the output together, and the structure is shown in Figure 3. A large dataset is necessary for training the deep SRU-RNN, and the parameters in each SRU (shown in Equations (1)–(5)) are determined or learned during the training procedure.

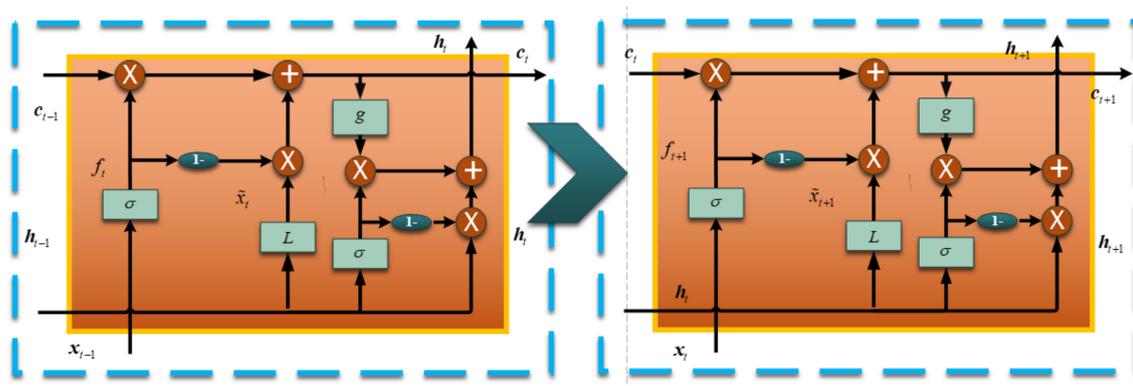


Figure 2. Working flow of two Simple Recurrent Unit–Recurrent Neural Networks.

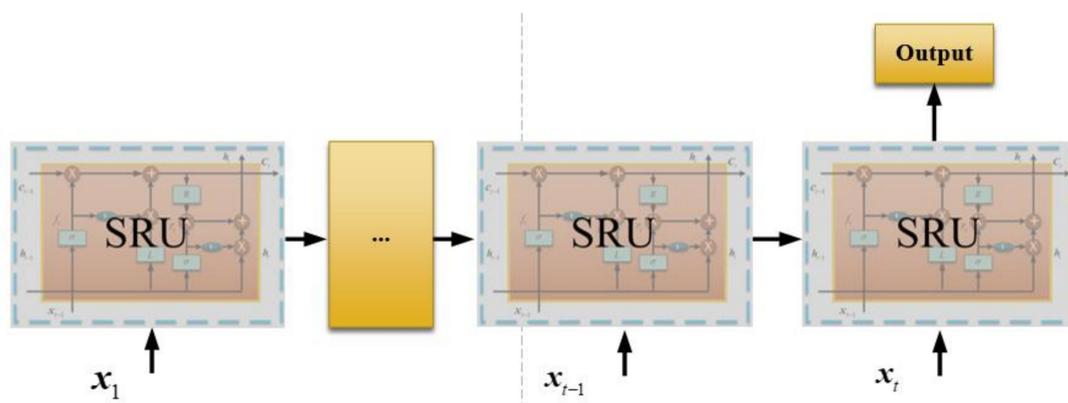


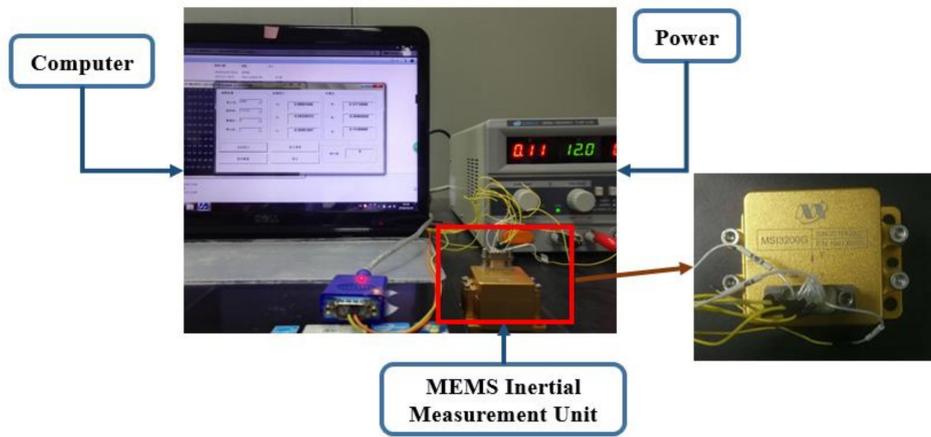
Figure 3. Basic structure of Deep Simple Recurrent Unit–Recurrent Neural Networks Training.

## 3. Experiments

With the aim of evaluating and verifying the proposed method, a MEMS IMU MSI3200 manufactured by MT Microsystems Company (Hebei, China) is employed in the following experiments [41]. Since attitude errors play an important role in position accuracy, only gyroscope de-noising results are presented and analyzed in this paper. A dataset with approximately 10 min time length is collected and the data collecting set up is shown in Figure 4. Several devices are employed in the operation including the MEMS IMU, a laptop, power supply and some cables. The MEMS IMU works at 12 volts and a laptop is employed to store the data. The IMU is composed of three-orthogonal gyroscopes and three-orthogonal accelerometers, and Table 1 lists the parameters of the employed MEMS IMU in detail. During the raw signal collecting, the MEMS IMU is placed on the table statically and the sampling frequency is set to 400 Hz here. The gyroscope output unit is degree/s and the accelerometer output unit is g ( $1\text{ g} = 9.8\text{ m/s}^2$ ). Furtherly, according to the IMU setting on table, specifically, the X-axis gyroscope measures the pitch angle, the Y-axis measures the roll angle, and the Z-axis measures the yaw angle.

**Table 1.** Specifications of MSI3200 IMU.

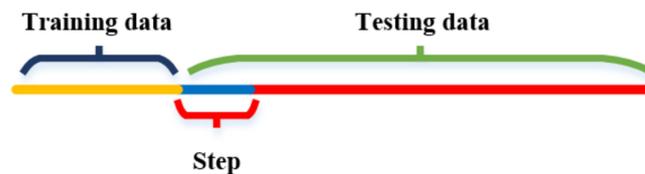
		Range	$\pm 300^\circ/s$
MEMS IMU	Gyroscope	Bias instability ( $1\sigma$ )	$\leq 10^\circ/h$
		Bias instability (Allan)	$\leq 2^\circ/h$
		Angle random walk	$\leq 0.15^\circ/\sqrt{h}$
	Accelerometer	range	$\pm 15\text{ g}$
		bias instability ( $1\sigma$ )	0.5 mg
		bias repeatability (Allan)	0.5 mg
	Power consumption	1.5 W	
	Weight	250 g	
	Size	70 mm $\times$ 54 mm $\times$ 39 mm	
	Sampling rate	400 Hz	



**Figure 4.** Gyroscope and Accelerometer raw signals collecting.

*3.1. Training Data Length Analysis*

In this sub-section, the influence of the training length on the SRU-RNN is discussed. Parameters of the employed SRU-RNN are listed in Table 2. Specifically, the training epoch is fixed as 100, while the length of the input data varies. The structure of the training data, input data length termed as step and testing data are explained in Figure 5. The yellow part of the line represents the training data, the middle blue part is the “step”, and the testing data includes the blue and red parts of the line. Since longer training data means more computation, and thus there might be a trade-off between training data length and prediction accuracy. Table 2 also lists the other specifications of the SRU-RNN employed. The learning rate is set to 0.01, and the hidden unit amount is 1. Moreover, the input data size or step is set 10.



**Figure 5.** Data structure.

**Table 2.** Specifications of Simple Recurrent Unit Recurrent Neural Networks (SRU-RNN).

Batch size	128
Training epoch	100
Learning rate	0.01
Hidden unit amount	1

Table 3 lists the prediction results of different training data length for the three-axis gyroscope, while the input data size and testing data size are fixed. The training data size varies from 200 to 10,000. For the X-axis gyroscope, the standard deviation (STD) values varying from 0.054 to 0.062 with the software running time varies from 57.8 s to 77.97 s. Figure 6 shows the SRU-RNN training loss comparison between data length of 200 and 500. The blue line represents the SRU training loss, with 200 training data length. The red line represents the SRU-RNN training loss with 500 training data length. From the Figure 6, it can be seen that SRU with a training data length of 200 does not converge within the set epoch of 100. However, with the 500 input data length for training, the SRU-RNN is able to converge within 100 training epoch.

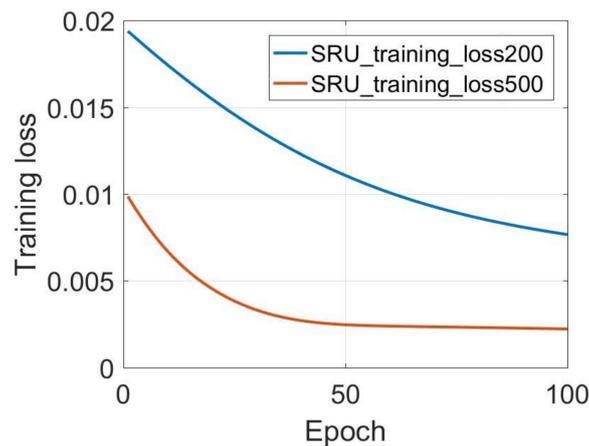


Figure 6. X-axis gyroscope training loss comparison.

In addition, the Y-axis and Z-axis gyroscopes results are also listed in Table 3. Similarly, the SRU-RNN is unable to converge when the data length is not sufficient with the 100 training epoch. Figures 7 and 8 show the training loss comparisons. It can be seen that SRU-RNN with 3000 training data length converge, while the SRU-RNN with 1000 training data length is unable to converge within the set training epoch values. In theory, more training data will lead to slower convergence speed. However, from the results, the SRU-RNN with more training data converges faster. SRU-RNN is a deep recurrent neural network depending on the memory. Under this condition, while the training data is of small length, it is not sufficient for this SRU-RNN training and learning of the model. Thus, the SRU-RNN is able to converge with sufficient training data length. In aspects of the standard deviation (STD) of the de-noised testing data, the STD values keeps almost the same, which might demonstrate that the SRU-RNN has been well trained. Therefore, for the X-axis gyroscope (Figure 6), 500 is sufficient, however, for Y-axis and Z-axis gyroscopes (Figures 7 and 8), 3000 is sufficient. The difference between the three gyroscopes might be caused by the MEMS manufacturing technology, which leads to the difference in the three-axis gyroscopes signals characteristics.

Table 3. Training results of the three-axis gyroscope.

Training Data Length	X		Y		Z	
	STD (Degree/s)	Time (Second)	STD (Degree/s)	Time (Second)	STD (Degree/s)	Time (Second)
10,000	0.062	77.0	0.057	121.5	0.025	86.9
3000	0.054	61.3	0.054	99.0	0.023	82.3
1000	0.055	57.8	/	/	/	/
500	0.055	60.7	/	/	/	/
200	/	57.7	/	/	/	/
/	0.073	/	0.082	/	0.045	/

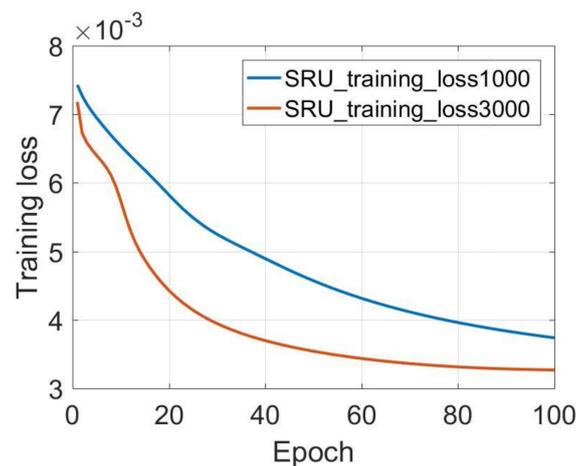


Figure 7. Y-axis gyroscope training loss comparison.

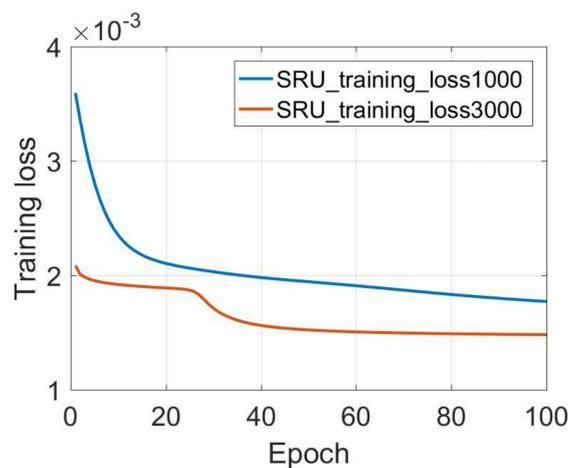


Figure 8. Z-axis gyroscope training loss comparison.

### 3.2. Different Parameters Compensation Analysis

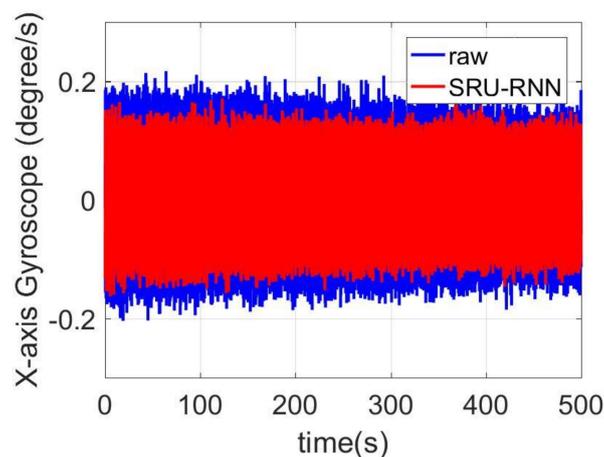
In Section 3.1, the influence of the training data length on the training or prediction results were analyzed. As aforementioned in the introduction section, quantization noise, angle random walk, and bias instability are the major index for describing the MEMS IMU performance. Thus, Figures 9–11 shows the comparison of the de-noised and raw signals, Figures 12–14 presented the Allan Variance comparison results, and Tables 4–6 show the comparison of the parameters between raw signals and de-noised signals. For the X-gyroscope, the quantization noise has a minor improvement, while the angle random walk and bias instability have a 6.8% and 12.5% improvement, respectively. The Y-axis gyroscope has a 60.5% improvement in quantization parameters and 17.3% and 34.1% in angle random work and bias instability, respectively. The rest Z-axis gyroscope has an improvement of 11.3%, 22.7%, and 35.7% in parameters of quantization noise, angle random walk, and bias instability individually. The attitude errors are listed in Table 7, the roll, pitch, and yaw angles errors decreased by 19.2%, 82.1%, and 69.4% individually. In this experiment, the X-axis gyroscope measured the pitch angle, the Y-axis gyroscope measures the roll angle, and the Z-axis gyroscope measures the yaw angles. These attitude angles were calculated based on the quaternion algorithms, and more detailed could be found in the reference [42]. Since the IMU was placed statically in a table, the initial values were all set to zero.

As aforementioned in Section 3, the X-axis gyroscope measures the pitch angle, the Y-axis measures the roll angle, and the Z-axis measures the yaw angle. Figures 15–17 shows the attitude errors. The roll angle performs the least decrease, which is reflected from the corresponding X-axis gyroscope analysis. For the raw and pitch angles, they have an improvement of 82.1% and 69.4% with the 500 s time length

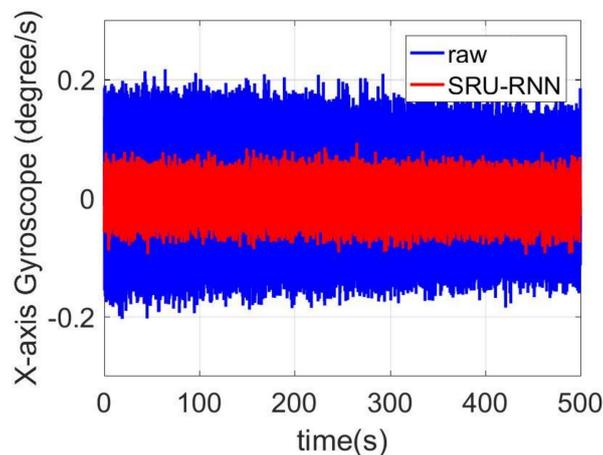
data. The difference between raw angle and pitch angle is caused by that the Y-axis gyroscope has smaller quantization noise. The Y-axis and Z-axis gyroscope have similar angle random walk and bias instability (listed in Tables 5 and 6). However, the Y-axis has a 60.5% decrease in quantization noise, while the Z-axis gyroscope just has 11.3% improvement.

**Table 4.** Parameters of X-axis gyroscope results.

	X		
	Raw	SRU-RNN	Percentage
quantization noise ( $\text{deg}/\sqrt{h}$ )	0.15	0.15	0.6%
Angle random walk ( $\text{deg}/\sqrt{h}$ )	0.44	0.41	6.8%
Bias instability ( $\text{deg}/h$ )	2.48	2.17	12.5%



**Figure 9.** De-noised and raw signals comparison for X-axis gyroscope.



**Figure 10.** De-noised and raw signals comparison for Y-axis gyroscope.

**Table 5.** Parameters of Y-axis gyroscope results.

	Y		
	Raw	SRU-RNN	Percentage
quantization noise ( $\text{deg}/\sqrt{h}$ )	1.0	0.40	60.5%
Angle random walk ( $\text{deg}/\sqrt{h}$ )	0.23	0.19	17.3%
Bias instability ( $\text{deg}/h$ )	1.29	0.85	34.1%

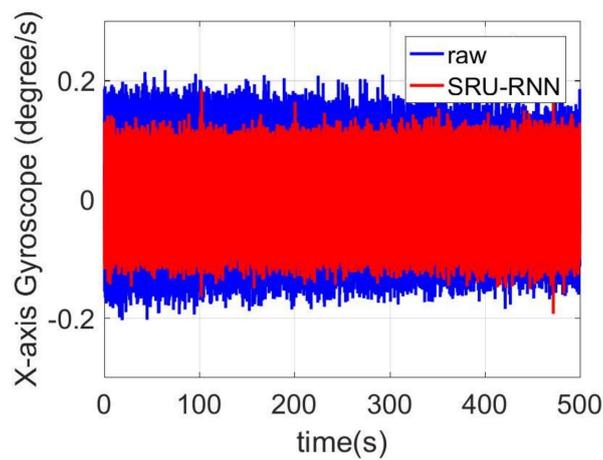


Figure 11. De-noised and raw signals comparison for Z-axis gyroscope.

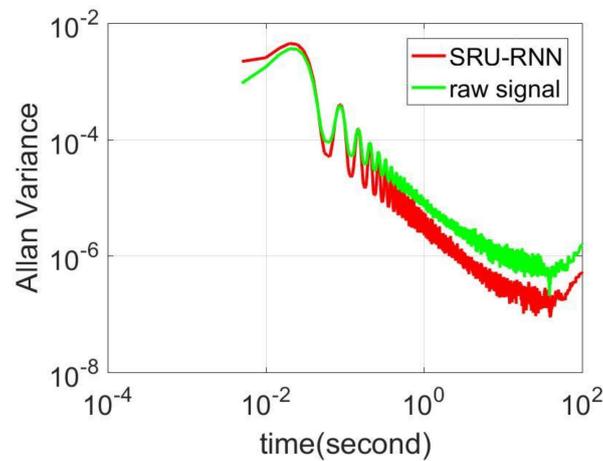


Figure 12. Allan variance comparison between de-noised and raw signals for X-axis gyroscope.

Table 6. Parameters of Z-axis gyroscope results.

	Z		
	Raw	SRU-RNN	Percentage
quantization noise (deg/ $\sqrt{h}$ )	0.62	0.55	11.3%
Angle random walk (deg/ $\sqrt{h}$ )	0.22	0.17	22.7%
Bias instability (deg/h)	1.12	0.72	35.7%

Table 7. Attitude results comparison.

	Attitude		
	Raw	SRU	Percentage
Pitch/(degree)	-1.04	-0.84	19.2%
Roll/(degree)	7.69	1.38	82.1%
Yaw/(degree)	-5.18	-1.58	69.4%

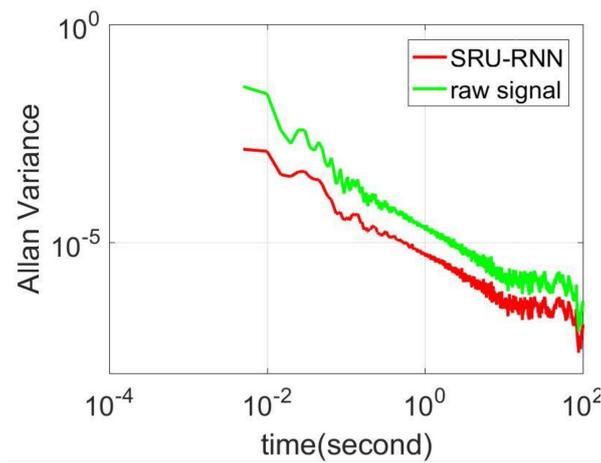


Figure 13. Allan variance comparison between de-noised and raw signals for Y-axis gyroscope.

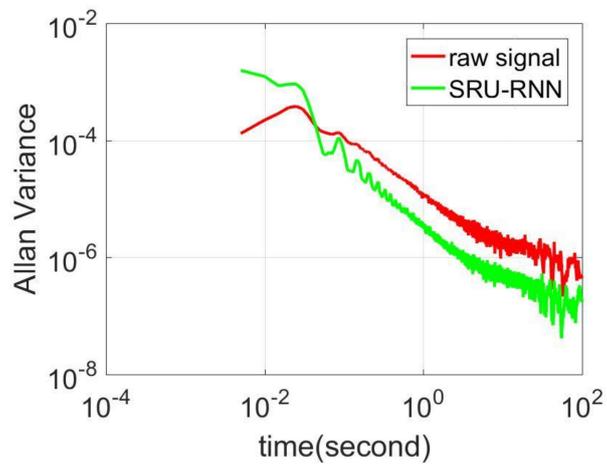


Figure 14. Allan variance comparison between de-noised and raw signals for Z-axis gyroscope.

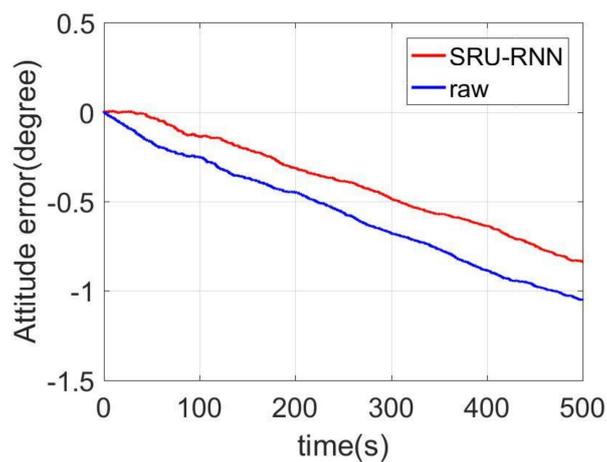


Figure 15. Pitch angles.

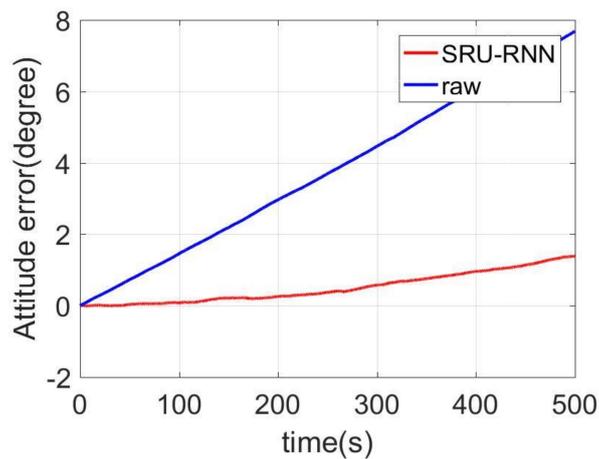


Figure 16. Roll angle errors comparison.

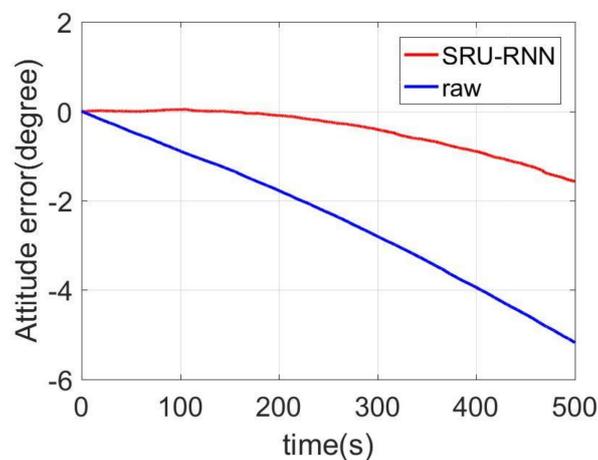


Figure 17. Yaw angle errors comparison.

#### 4. Conclusions

This paper investigated a deep Simple Recurrent Unit Recurrent Neural Networks (SRU-RNN) based MEMS gyroscope de-noising method, from the experimental results, these conclusions were obtained:

- (1) There was a trade-off between the training data length and the de-noising performance, for the employed Inertial Measurement Unit, 500, 3000, and 3000 was sufficient for learning the model with set 100 training epoch;
- (2) Among the major three Inertial Measurement Unit errors describing parameters, there was no regular pattern for the compensation degree of the parameters;
- (3) The three-axis attitude had an improvement of 19.2%, 82.1%, and 69.4%, and which is consistent with the analysis from the three gyroscope signals. The results demonstrated the effectiveness of the proposed SRU-RNN method.

However, there were following limitations of this paper:

- (1) In the experiments, the SRU-RNN was trained with fixed parameters including the learning rate and batch size. Parameters optimization might improve the performance of the SRU-RNN in this application. Some optimization methods are available in the AI community;
- (2) As aforementioned, the SRU-RNN is single layer, and actually a multi-layer SRU-RNN might improve the performance of the SRU-RNN;
- (3) In the experiments, limited by the lab equipment, only the static dataset was collected and employed.

Future work will include: (1) further analysis in signals characteristics is necessary for exploring what causes the difference in training data length; (2) comparing SRU-RNN, LSTM-RNN, and some other variants of RNN using same MEMS IMU dataset, finding suitable RNN for MEMS gyroscope and accelerometer raw signals de-noising; (3) dynamic or field testing dataset will be employed for furtherly investigated the deep learning method in application of MEMS IMU noises modeling; and, (4) it will be great significance of embedding a deep learning module in MEMS IMU, a feasibility study will be conducted in a GNSS/MEMS IMU integrated navigation system for evaluating the performance in GPS signal outage of 60 s.

**Author Contributions:** C.J. proposed the idea, designed the experiments and written the first version of the paper. S.C. discussed the idea, guided the paper writing and provided the funding. Y.C. revised the paper and guided the paper writing. L.H. and J.G. collected and processed the MEMS IMU raw data. H.Z. and Y.B. reviewed the paper before submission, and provided valuable revision for this paper.

**Funding:** This research was funded by the Fundamental Research Funds for the Central Universities (Grant No. 30917011105); the National Defense Basic Scientific Research program of China (Grant No. JCKY2016606B004); Jiangsu Planned Projects for Postdoctoral Research Funds, grant number (Grant No. 1501050B); the China Postdoctoral Science Foundation, grant number (Grant No. 2015M580434) and the special grade of the financial support from the China Postdoctoral Science Foundation with grant number (Grant No. 2016T90461).

**Acknowledgments:** The author gratefully acknowledges the financial support from China Scholarship Council (CSC, Grant No. 201706840087) and excellent doctor training fund of Nanjing University of Science and Technology (NJUST). This paper was supported by the Fundamental Research Funds for the Central Universities (Grant No. 30917011105), the National Defense Basic Scientific Research program of China (Grant No. JCKY2016606B004); Jiangsu Planned Projects for Postdoctoral Research Funds (Grant No. 1501050B), the China Postdoctoral Science Foundation (Grant No. 2015M580434) and the special grade of the financial support from the China Postdoctoral Science Foundation (Grant No. 2016T90461). Source code and dataset are available, do not hesitate to contact me, and I will share the all the materials.

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

## References

1. Lutwak, R. Micro-technology for positioning, navigation, and timing towards PNT everywhere and always. In Proceedings of the IEEE International Symposium on Inertial Sensors and Systems (ISISS), Laguna Beach, CA, USA, 25–26 February 2014.
2. Shkel, A. The chip-scale combinatorial atomic navigator. *GPS World* **2013**, *24*, 8–10.
3. Jiang, C.H.; Chen, S.; Chen, Y.W.; Bo, Y.M. Research on Chip Scale Atomic Clock Driven GNSS/SINS Deeply Coupled Navigation System for Augmented Performance. *IET Radar Sonar Navig.* **2018**. [[CrossRef](#)]
4. Li, X.X.; Ge, M.R.; Dai, X.L.; Ren, X.D.; Mathias, F.; Jens, W.; Harald, S. Accuracy and reliability of multi-GNSS real-time precise positioning: GPS, GLONASS, BeiDou, and Galileo. *J. Geod.* **2015**, *89*, 607–635. [[CrossRef](#)]
5. Lou, Y.D.; Zheng, F.; Gu, S.F.; Wang, C.; Guo, H.L.; Feng, Y.M. Multi-GNSS precise point positioning with raw single-frequency and dual-frequency measurement models. *GPS Solut.* **2016**, *20*, 849–862. [[CrossRef](#)]
6. He, H.B.; Li, J.L.; Yang, Y.X.; Xu, J.Y.; Guo, H.L.; Wang, A.B. Performance assessment of single-and dual-frequency BeiDou/GPS single-epoch kinematic positioning. *GPS Solut.* **2014**, *18*, 393–403. [[CrossRef](#)]
7. Groves, P.D. Shadow matching: A new GNSS positioning technique for urban canyons. *J. Navig.* **2011**, *64*, 417–430. [[CrossRef](#)]
8. Chen, X.Y.; Shen, C.; Zhang, W.B.; Tomizuka, M.; Xu, Y.; Chiu, K. Novel hybrid of strong tracking Kalman filter and wavelet neural network for GPS/INS during GPS outages. *Measurement* **2013**, *46*, 3847–3854. [[CrossRef](#)]
9. Qin, H.L.; Li, C.; Sun, X.L. Accuracy improvement of GPS/MEMS-INS integrated navigation system during GPS signal outage for land vehicle navigation. *J. Syst. Eng. Electron.* **2012**, *23*, 256–264. [[CrossRef](#)]
10. Chiang, K.W.; Huang, Y.W. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. *Appl. Soft Comput.* **2008**, *8*, 722–733. [[CrossRef](#)]
11. Lashley, M.; David, M.; John, Y.H. Performance analysis of vector tracking algorithms for weak GPS signals in high dynamics. *IEEE J. Sel. Top. Signal Process.* **2009**, *3*, 661–673. [[CrossRef](#)]
12. Zhao, S.H.; Lu, M.Q.; Feng, Z.M. Implementation and performance assessment of a vector tracking method based on a software GPS receiver. *J. Navig.* **2001**, *64*, S151–S161. [[CrossRef](#)]

13. Paul, A.; Barbeau, S.J. Positional accuracy of assisted GPS data from high-sensitivity GPS-enabled mobile phones. *J. Navig.* **2011**, *64*, 381–399.
14. Wang, J.H.; Gao, Y. High-sensitivity GPS data classification based on signal degradation conditions. *IEEE Trans. Veh. Technol.* **2007**, *56*, 566–574. [[CrossRef](#)]
15. Lachapelle, G. Pedestrian navigation with high sensitivity GPS receivers and MEMS. *Pers. Ubiquitous Comput.* **2007**, *11*, 481–488. [[CrossRef](#)]
16. Yuting, N.; Gao, X.X. GNSS Multi-receiver Vector Tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 2583–2593.
17. Jiang, C.H.; Chen, S.; Chen, Y.W.; Bo, Y.M.; Wang, C.C.; Tao, W. Performance Analysis of GNSS Vector Tracking Loop Based GNSS/CSAC Integrated Navigation System. *J. Aeronaut. Astronaut. Aviat.* **2017**, *49*, 289–297.
18. Jiang, C.H.; Chen, S.; Bo, Y.M.; Lu, Q.W.; Sun, Z.H. An Adaptive Tuning Method of GNSS Carrier Tracking Loop for High Dynamic Application. *J. Aeronaut. Astronaut. Aviat.* **2016**, *48*, 271–277.
19. Sameh, N.; Niu, X.J.; El-Sheimy, N. Land-vehicle INS/GPS accurate positioning during GPS signal blockage periods. *J. Surv. Eng.* **2007**, *133*, 134–143.
20. Cui, Y.J.; Ge, S.Z. Autonomous vehicle positioning with GPS in urban canyon environments. *IEEE Trans. Robot. Autom.* **2003**, *19*, 15–25.
21. Bhatt, D.; Aggarwal, P.; Devabhaktuni, V.; Bhattacharya, P. A novel hybrid fusion algorithm to bridge the period of GPS outages using low-cost INS. *Expert Syst. Appl.* **2014**, *41*, 2166–2173. [[CrossRef](#)]
22. Yuting, N.; Gao, X.X. Joint GPS and vision direct position estimation. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS), Savannah, GA, USA, 11–16 April 2016.
23. Brown, A.K. GPS/INS uses low-cost MEMS IMU. *IEEE Aerosp. Electron. Syst. Mag.* **2005**, *20*, 3–10. [[CrossRef](#)]
24. Lee, J.G.; Chan, G.P.; Heung, W.P. Multi-position alignment of strap-down inertial navigation system. *IEEE Trans. Aerosp. Electron. Syst.* **1993**, *29*, 1323–1328. [[CrossRef](#)]
25. Rhee, I.; Mamoun, F.A.; Jason, L.S. Observability of an integrated GPS/INS during maneuvers. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *40*, 526–535. [[CrossRef](#)]
26. Jiang, C.H.; Chen, S.; Liu, Y.L.; Han, X. Implementation and performance evaluation of a distributed GNSS/SINS ultra-tightly integrated navigation system. In Proceedings of the IEEE International Instrumentation and Measurement Technology Conference Proceedings (I2MTC), Taipei, Taiwan, 23–26 May 2016.
27. Naser, E.S.; Hou, H.Y.; Niu, X.J. Analysis and modeling of inertial sensors using Allan variance. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 140–149.
28. Quinchia, A.G.; Falco, G.; Falletti, E.; Dosis, F.; Ferrer, C. A comparison between different error modeling of MEMS applied to GPS/INS integrated systems. *Sensors* **2013**, *13*, 9549–9588. [[CrossRef](#)] [[PubMed](#)]
29. Syed, Z.F.; Aggarwal, P.; Goodall, C.; Niu, X.J.; El-Sheimy, N. A new multi-position calibration method for MEMS inertial navigation systems. *Meas. Sci. Technol.* **2007**, *18*, 1897. [[CrossRef](#)]
30. Vaccaro, R.J.; Ahmed, S.Z. Statistical modeling of rate gyros. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 673. [[CrossRef](#)]
31. Aggarwal, P.; Syed, Z.; Niu, X.J.; El-Sheimy, N. A standard testing and calibration procedure for low cost MEMS inertial sensors and units. *J. Navig.* **2008**, *61*, 323–336. [[CrossRef](#)]
32. Bekkeng, J.K. Calibration of a novel MEMS inertial reference unit. *IEEE Trans. Instrum. Meas.* **2009**, *58*, 1967–1974. [[CrossRef](#)]
33. Ning, Y.P.; Wang, J.; Han, H.Z.; Tan, X.L.; Liu, T.J. An Optimal Radial Basis Function Neural Network Enhanced Adaptive Robust Kalman Filter for GNSS/INS Integrated Systems in Complex Urban Areas. *Sensors* **2018**, *18*, 3091. [[CrossRef](#)]
34. Bhatt, D.; Aggarwal, P.; Bhattacharya, P.; Devabhaktuni, V. An enhanced mems error modeling approach based on nu-support vector regression. *Sensors* **2012**, *12*, 9448–9466. [[CrossRef](#)] [[PubMed](#)]
35. Chu, Y.D.; Fei, J.T. Adaptive global sliding mode control for MEMS gyroscope using RBF neural network. *Math. Probl. Eng.* **2015**, *2015*, 403180. [[CrossRef](#)]
36. El-Rabbany, A.; El-Diasty, M. An efficient neural network model for de-noising of MEMS-based inertial data. *J. Navig.* **2004**, *57*, 407–415. [[CrossRef](#)]
37. Hosseinyalamdary, S. Deep Kalman Filter: Simultaneous Multi-Sensor Integration and Modelling; A GNSS/IMU Case Study. *Sensors* **2018**, *18*, 1316. [[CrossRef](#)] [[PubMed](#)]

38. Jiang, C.H.; Chen, S.; Chen, Y.W.; Zhang, B.Y.; Feng, Z.Y.; Zhou, H.; Bo, Y.M. A MEMS IMU De-Noising Method Using Long Short Term Memory Recurrent Neural Networks (LSTM-RNN). *Sensors* **2018**, *18*, 3470. [[CrossRef](#)]
39. Lei, T.; Zhang, Y.; Wang, S.I.; Dai, W.; Artzi, Y. Simple recurrent units for highly parallelizable recurrence. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018.
40. SRU. Available online: <https://www.zhihu.com/question/65244705> (accessed on 1 September 2017).
41. MSI3200. Available online: [http://www.mtmems.com/product\\_view.asp?id=28](http://www.mtmems.com/product_view.asp?id=28) (accessed on 1 December 2017).
42. Großekathöfer, K.; Yoon, Z. Introduction into Quaternions for Spacecraft Attitude Representation. TU Berlin, Berlin, Germany, 2012.



© 2018 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/>).