INTELLIGENT SENSOR POSITIONING AND ORIENTATION USING A SGN EMBEDDED FUSION ALGORITHM FOR A MEMS IMU/GPS INTEGRATED SYSTEM

Hsiu-Wen Chang[a], Kuan-Yun Chen[a], Kai-Wei Chiang[a*],Naser El-Sheimy [b]
[a] Department of Geomatics, National Cheng-Kung University, Taiwan
[b] Department of Geomatics Engineering , University of Calgary,Calgary
(Tel:(886)6-237-0876 ext. 857, E-mail: kwchiang@mail.ncku.edu.tw)

Commission VI, WG VI/5

KEY WORDS: GPS/INS, Integration, Mobile Mapping Systems, Constructive Neural Networks, SGN

ABSTRACT:

MMSs have been applied widely for acquiring spatial information in applications such as GIS and 3D city models. Nowadays the most common technologies used for MMS positioning and orientation include using GPS as a major positioning sensor and INS as the major orientation sensor. In the classical approach, the limitation of KF and the price of overall multi-sensor systems have limited the popularization of most land-based MMS applications. Although intelligent sensor positioning and orientation schemes have been proposed consisting of MFNN, one of the most famous ANNs, and KF/RTS, in order to enhance the performance of a low cost MEMS INS/GPS integrated system, the automation of the MFNN applied is not as easy as initially expected. Therefore, this study not only addresses the problems of insufficient automation in the conventional methodology that has been applied in MFNN-KF/RTS algorithms for INS/GPS integrated system proposed in previous studies, but also exploits and analyzes the idea of developing alternative intelligent sensor positioning and orientation schemes that integrate various sensors in a more automatic way. The proposed schemes are implemented using SGN to overcome the limitations of conventional techniques based on the KF/RTS algorithms as well as previously developed MFNN-KF/RTS schemes. The SGN(CCN)also has the advantage of a more flexible topology compared to the MFNN for INS/GPS integration. The results presented in this article illustrate the effectiveness of the proposed schemes over both KF/RTS algorithms as well as the MFNN-KF/RTS schemes.

1. INTRODUCTION

DMMS have been applied widely for acquiring spatial information in the applications such as Geographic Information Systems (GIS) and 3D city model. The basic idea is executed by producing more than one image that includes the same object from different positions, and then the 3D positions of the same object with respect to the camera frame can be measured. Direct geo-referencing is the determination of time-variable position and orientation parameters for a mobile digital imager [El-Sheimy, 1996]. Instead of using ground control points as references for orientating the images in space, the trajectory and attitude of the imager platform could now be determined directly [Park and Gao, 2008]. Caused by the need of faster update rate and the increasing demand, the DMMS has been applied to overcome the prohibitions of conventional survey techniques. This system is less expensive and has higher applicability than the conventional one. In order to attain reasonable accuracy of position and attitude solutions, tactical grade or higher quality IMU along with GPS has been applied as the Position and Orientation System (POS) for current commercial systems. However, the cost of overall system still be maintained at such a high level that limits the popularization, especially the price of the IMU.

The Kalman filter (KF) approach has been widely recognized as the standard optimal estimation tool for current INS/GPS integration schemes. The basic idea of using KF in GPS/INS integration is to fuse those independent and redundant sources of navigation information with a reference navigation solution to obtain an optimal estimate of navigation states such as position, velocity and attitude. However, it has limitations, which have been reported by several researchers [Gelb, 1974; Brown and Hwang, 1992; Vanicek and Omerbasic, 1999]. The major inadequacy related to the utilization of the KF for INS/GPS integration is the necessity to have a predefined accurate stochastic model for each of the sensor errors [Brown and Hwang, 1992]. On the other hand, the smoothing has been applied for the purpose of accurate positioning and orientation determination through post-processing for most of the kinematic positioning applications. In contrast to the KF, the smoothing is implemented after all KF estimates have been solved by the use of past, present and future.

ANN techniques have been applied to develop alternative INS/GPS integration schemes to overcome the limitations of KF and to improve the positional accuracy of vehicular navigation systems during GPS signal blockages [Chiang, 2004]. Such an integrated approach would have the capability of estimating all navigation states, using the advantages of ANN techniques for practical solutions. The MFNN is the most common use of ANN in the previous studies [Bishop, 1995; Chiang, 2004; Lin, 2008]. However, this approach still has a lot of issues that have not been resolved completely. These include the determination of the number of hidden-layer neurons, convergent time for adjusting weight and the speed of convergence in training. The topology of MFNN such as neurons and layer numbers can be appropriate decided only after numerous trying. Therefore, this thesis aims at using constructive neural network that can grow itself during the learning process. It will effectively reduce the trying process and still maintain the performance generated by MFNN.

2. METHOD

2.1 Problem Statements

In general GPS/INS integration applications, the accuracy of the KF solutions sometime cannot fulfill applications such as a MMS. In detail, an integrated system has to predict state parameters such as position, velocity and attitude using KF when GPS signal blockages exist. In GPS denied environments, the errors of navigation solutions increase rapidly until GPS signal can be recovered to update the measurement. This problem will become more serious when a MEMS (Micro Electro Mechanical Systems) IMU is used. In order to achieve high accuracy requirements for position and attitude determination in DMMS, it is processed in post-mission mode with an optimal smoothing algorithm. Most of the commercial mobile mapping systems use an optimal smoothing algorithm to provide accurate position and orientation for direct geo-referencing [Shin, 2005]. However, INS/GPS integrated POSs use tactical grade IMU or above to provide accurate POS solutions for general MMS applications. Therefore, upgrading the hardware (e.g., IMU) can be considered as an effective solution to improve the accuracy of POS solutions when a low cost MEMS IMU is used. However, such improvement is rather limited as the availability of high grade (navigation) IMUs is regulated by the governmental regulations of certain countries where the IMUs are produced.

Another effective way to improve the accuracy of low cost MEMS INS/GPS integrated POS solutions is through the improvement of POS algorithm. Comparing to the hardware perspective mentioned above, the software perspective can be considered as a cost effective solution to develop a low cost INS/GPS integrated POS for general MMS applications. One of famous algorithm is the combination of ANN and KF or smoothing. The purpose of ANN used in GPS/INS integration is to reduce the drawbacks of KF and reduce remaining errors in KF and smoothing solutions. However, it is difficult to train MFNN well and it is time-consuming for most users learn about how to design the best architecture for MFNN [Alpaydin, 1991]. Compared to fixed topology based neural networks like MFNNs; the ANNs with constructive algorithms are considered computationally economic. Consequently, the proposed scheme is implemented using CCNs to overcome the limitations of the previous one. The two key ideas of CCNs are the cascade architecture and learning algorithm which creates and installs the new hidden unit with maximum correlation.

Therefore, the objectives of this article is to: (1) develop CCN-KF and CCN-RTS smoother schemes for precise position and attitude determination; (2) verify the performance of proposed system using a MEMS IMU/GPS integrated system; (3) compare the performance with the previous developed MFNN-RTS hybrid schemes in terms of complexity of the topology, the learning time and estimated accuracy during GPS signal outages of the proposed schemes and (4) analyze the correlation between several inputs with the specific target of proposed algorithms.

2.2 The Artificial Neural Networks

In this study, the constructive ANN is implemented to learn and compensate for the residual errors of the KF and RTS smoother, respectively, to improve the accuracy of the attitude angles estimated by the KF and RTS smoother, respectively. The proposed scheme is capable of learning how the state vector (i.e., position or attitude errors) behave based on the dynamics of the platform and the error characteristics of the inertial sensors being used. The residual error compensation scheme of the KF involves a series of complicated non-linear function approximations to adapt to the variations of vehicle dynamics or sensor errors [Chiang, 2004]. The self-growing neural network is the obvious choice to learn nonlinear functional relationships, and in particular self-growing neural network is more automatically than fixed neural network such as multilayer feed-forward neural networks (MFNN).

ANNs have been motivated right from its inception by the recognition that the human brain functions in an entirely different way from the conventional digital computer. Therefore, the simplest form of ANN can be depicted like human nervous system. The receptors are used to convert input signals into appropriate vector that could be processed by central network. And the effectors are used to transfer the output vector into readable response. In general, the basic model of the neuron contains three major components: (a) weight links $\left\langle w_{i,j}, W_{j,k} \right\rangle$; (b) an adder for summing the input signals ϕ_i that are weighted by respective synapses of the neuron and external bias (b_k); and (c) an activation function $\varphi(\bullet)$ for limiting the amplitude of the neuron output and the final output y_k .

To determine the weight values one must have a set of examples of how the outputs, $\hat{y}_i$, should relate to the input, ϕ_l , the process of obtaining the weights from these examples is called supervised learning and it is basically a conventional estimation process. That is, the weights are estimated from existing examples in such a way that the network, according to some metric, models the true relationship as accurate as possible. This supervised learning process can be implemented through the use of backpropagation learning algorithm.

There are several constructive models. The overall reviews of current constructive algorithms can be found in [Alpaydin, 1991]. In reference, the CCN is the most famous one because of its ability to speed up the training process and design topology automatically. CCN was developed in 1990 by Scott E. Fahlman and Christian Lebiere [Fahlman and Lebiere, 1990]. The two key ideas of this implementation are: (1) a cascade architecture and (2) a unique learning algorithm for training and installing new hidden neuron. CCN begins with a minimal network that only consists of input layer and output layer, as shown in Figure (6). Then automatically trains and adds new layer with hidden neuron one by one. The optimal values of input-output synaptic weights are computed during the training process. Any conventional training algorithm for single layer network can be applied. According to [Fahlman and Lebiere, 1990], the better choice of training algorithm is a second-order method, based loosely on Newton's method, Quickprop.

CCN consists of three parts: (a) starts from the simplest topology and pass the input vector to generate corresponding output vector then adjust output side weights using Quickprop algorithm. (b) When the goal performance can't be achieved, pools of candidate neurons that have different set of random initial weights are applied to execute the second step while the output side weights are frozen. All the candidate neurons receive the input signals from the input layer and from all preexisting hidden layer. Also the same residual error for each training pattern feedback from the output neurons will be received by all candidate neurons. Then the weights between candidate layer, input layer, and preexisting hidden layer are adjusted to maximize the correlation (C) between the output of each candidate neurons (V) and the residual error (E) at the output neuron.;

$$C = \sum_{o} \left| \sum_{p} (V_p - \bar{V})(E_{p,o} - \bar{E}_o) \right| \qquad (1)$$

where o is the network output at which the error $E_{p,o}$ is measured and p is the training pattern. The $\bar{V}$ and $\bar{E}_o$ are the mean values of V and E_o. The Quickprop algorithm is applied to adjust the incoming weights for each candidate neurons to maximize its own correlation(C). The derivative of correlation is computed by:

$$\delta_p = \sum_{o} \sigma_o (E_{p,o} - \bar{E}_o) \varphi'_p \qquad (2)$$

$$\frac{\partial C}{\partial w_i} = \sum_{p} \delta_p I_{i,p} \qquad (3)$$

where σ_o is the sign of the correlation between the candidate's value and output o, φ'_p is the derivative for pattern p of the candidate unit's activation function with respect to the sum of its inputs, and $I_{i,p}$ is the input that candidate unit receives from unit i for pattern p. Equations (1) are used to adjust incoming weights until no more improvement in each candidate neuron's correlation.

The neuron with highest correlation will be inserted into network as a new hidden layer shown in Figure (1); (c) frozen the input side weights and retraining all weights connect to the output layers. It is worth to mention that the hidden layer are all connect to output layer like new input neuron. If the output performance still cannot meet the requirement, it goes back to (b) and grows another new hidden neuron. On the other hand, the network will stop automatically if the goal performance is achieved.

2.3 System Architecture

In conventional algorithm, the KF and RTS smoother are used to provide optimal navigation solutions (position, velocity, and attitude). The EKF applied in this study has 21 states: $[\delta p_{1\times3} \quad \delta v_{1\times3} \quad \delta A_{1\times3} \quad b_{a,1\times3} \quad b_{g,1\times3} \quad s_{a,1\times3} \quad s_{g,1\times3}]^T$.

As shown in Figure (3), KF and RTS smoother are utilized to optimally estimate those 21 states and to compensate for their effect in real-time and post-mission modes, respectively. In fact, either approach can provide optimally estimated navigation parameters. In addition, sensor biases ($b_{a,1\times3}$ and $b_{g,1\times3}$) and scale factors ($S_{a,1\times3}$ and $S_{g,1\times3}$) can be estimated and feedback to the INS mechanization to correct the raw measurements provided by and IMU. However, since the scope of the study is limited to POS parameters, including positions and attitude angles. That means the sensors errors are not included in the input signal to ANN.

The errors of POS parameters estimated by KF and RTS smoother are used as the desired output or target values during the learning process of the proposed ANN architectures that both MFNN and CCN all are applied. The POS parameters estimated by KF and RTS smoother along with the time information in each scenario are used as the inputs of the proposed architectures. The goal of proposed schemes is to compensate for the errors of the POS states estimated by KF and RTS smoother during GPS outages. A superior IMU is applied as the reference system to generate the reference solutions computed by the post-mission process (e.g. RTS smoother) with the full availability of GPS, respectively. Then the target values are the errors of the KF and RTS smoother with intentionally added GPS outages with respect to reference solutions.

An ANN with an optimal topology is expected to provide the best approximation accuracy to the unknown model using the most appropriate number of hidden neurons and hidden layers. The CCN has flexible topology as mentioned before that there is no need to design these two parameters. But in MFNN, there are many ways to decide on the most appropriate number of hidden neurons; see [Haykin, 1991] for details. The common principle indicates that the most appropriate number of hidden neurons is application dependent and can only be decided empirically during the early stages of the topology design. It is very common in the design phase of neural networks to train many different candidate networks that have different numbers of hidden neurons and then to select the best, in terms of its performance based on an independent validation set [Bishop, 1995].

The MFNN used in this study uses the topology proposed by Lin [2008]. The way Lin [Lin, 2008] used to decide the optimal number of hidden neurons required for the proposed scheme is the empirical approach.

After being well trained, the proposed ANN compensation scheme was added to a loosely coupled INS/GPS integration architecture (closed loop) as shown in Figure (1). The intelligent architectures first receives raw data from an IMU and then use the INS mechanization along 21 states of KF and RTS smoother to estimate POS parameters, respectively. Meanwhile, the estimated POS parameters are sent to the proposed ANN architecture along with time information to generate predicted errors to compensate for the estimated POS parameters provided by KF and RTS smoother simultaneously. Errors of POS parameters are predicted with the proposed ANN scheme. The correction can be completed after the predicted errors have been removed from the outputs of KF and RTS smoother, respectively. It is worth mentioning that if the ANN has been well trained, there is almost no need to wait for the output from neural network. Therefore, the proposed ANN-KF hybrid scheme has the ability to be used in real-time.

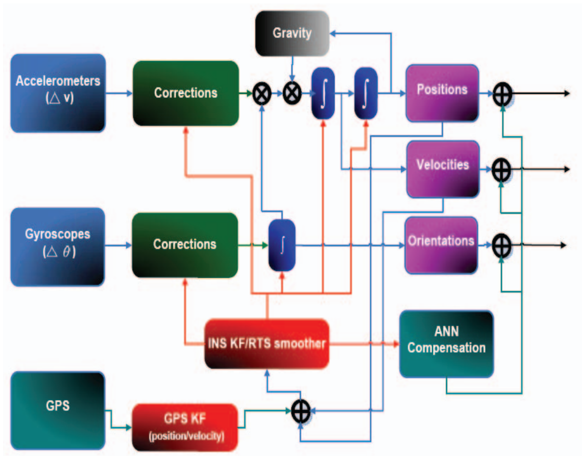

Figure 1: The implementation of ANN embedded KF and RTS smoother.

3. RESULTS AND DISCUSSIONS

To evaluate the performance of the proposed schemes, three field tests are used. The field tests are used to verify the performance of the proposed schemes. Those tests were conducted in land vehicle environments using different integrated systems consisting of one tactical grade IMU, Litton LN200 (1 deg/hr), a low cost MEMS IMU, BEI MotionPak II and two NovATel OEM-4 receivers. In this study, those IMUs were applied to collect inertial measurements in the field and then those measurements along with carrier phase DGPS solutions were fed into software that has inertial navigation algorithm and EKF to estimate inertial states optimally. The integrated system with LN200 IMU was used as the reference system. The measurements and navigation solutions provided by the integrated system with MotionPak II were used to verify the performance of proposed schemes.

The GPS measurements were processed using GrafNav™ software (Waypoint Consulting Inc.) in carrier phase DGPS to achieve ten centimeter level accuracy. The reference trajectories were generated by the integrated system with LN 200 IMU. They were determined using 21 states EKF and RTS backward smoothing. The parameters of EKF and the smoother applied in this article were well tuned so that they can represent the best achievable navigation accuracy for tactical grade IMUs.

The outputs of KF and RTS smoother provided by those systems were applied as the inputs for the proposed architectures. Several inputs dimension are considered by choosing some of the outputs from KF and RTS smoother. In addition, the outputs of KF and RTS smoother with simulated GPS outages were then compared with the reference trajectory. The errors, which can be interpreted as the error behavior of KF and RTS smoother, were then applied as the desired output for training. The dynamic variations experienced by the vehicle during the simulated outages include straight line segments, sharp turns, accelerations and decelerations. It is worth mentioning that five simulated outages, marked by triangles, were used as the independent dataset for cross validation during training process to ensure generalization capability as well as to avoid possible over-training problems.

On the other hand, sixteen GPS outages in total, each of them has 30 seconds in length, were simulated using the measurements collected in the first and second field test using the INS/GPS integrated with the MotionPak II (MEMS), respectively.

3.1 The Training of Proposed Schemes

To show the meaning of the significant improvements, the proposed scheme's ability to catch the error behavior, including the impacts of dynamic variations and INS sensor errors of KF and RTS smoother, during training should be confirmed. The performance of proposed schemes still needs to be verified using other independent data sets, which will be presented in the next section. As indicated in Table 1, the proposed schemes both learn the error behavior at the similar level in position and attitude.
As shown in Table 1, the columns labeled "original" represent the "raw" attitude errors of the KF and RTS smoother comparing to the reference solutions, respectively. Similarly, the columns labeled "compensated" represent the "corrected" POS parameters of the KF and RTS smoother after applying proposed ANN-KF and ANN-RTS smoother schemes comparing to the reference solutions, respectively. As indicated in Table 1, the proposed ANN-KF and ANN-RTS smoother schemes learn the error behaviors of the KF and RTS smoother well during simulated GPS outages, especially in the heading angles and height.

Table 1: Training results summary

Method 6I Train	POS	RMS value			Improvement. (%)	
		original	residual CCN	residual MFNN	Against KF(RTS) CCN	Against KF(RTS) MFNN
	North(m)	1.0806	0.3188	0.2733	70	75
Tj-3	East(m)	1.7810	0.3421	0.2149	81	88
(KF + CCNs)	Height(m)	0.3651	0.0773	0.0678	79	81
(KF+MFNNs)	Roll(deg)	0.9193	0.0488	0.0375	95	96
	Pitch(deg)	0.5733	0.0399	0.0343	93	94
	Heading(deg)	6.4335	0.9213	0.4959	86	92
	North(m)	0.2331	0.0420	0.0395	82(96)	83(96)
Tj-3	East(m)	0.2165	0.0387	0.0470	82(98)	78(97)
(RTS+CCNs)	Height(m)	0.1032	0.0360	0.0335	65(90)	68(91)
(RTS+MFNNs)	Roll(deg)	0.9289	0.0314	0.0230	97(97)	98(97)
	Pitch(deg)	0.5862	0.0199	0.0190	97(97)	97(97)
	Heading(deg)	2.7762	0.5667	0.4838	80(91)	83(92)

3.2 Performance Verification of Proposed Schemes

The networks trained form trajectory three can be used to predict error compensation in other trajectory [Chang and Li, 2008]. The reason for using networks generated by trajectory three to test other trajectories is the dynamic variations experienced by the vehicle during the simulated outages include straight line segment, sharp turn, accelerations and decelerations. In Figures 2, the attitude test results in trajectories one is successful in roll and pitch but fails in heading.

Usually, the heading state of a vehicle is more complex than roll and pitch states. The failure could be caused by the variation of heading, the heading information in trajectory one and two are simpler than the heading in trajectory three. The above results are using four input vectors (time, roll, pitch, and heading) because the velocities in three directions could not effectively reduce the output error. The heading error in those trajectories is too different; the ANN could not effectively reduce it. However, it seems that CCN has a higher stability in making the networks' output smoother and the predict solutions under reasonable range. In the experiments, the different training epoch of MFNN causes different results in other samples. Although adding training epoch can make the training output approximate the target clearly, the prediction of other sample may be even worse than the seldom one. This characteristic makes the MFNN time consuming in tuning the most appropriate training results. However, both of them eliminate system bias in roll and pitch error state. This is caused by the different location between reference system and test system.

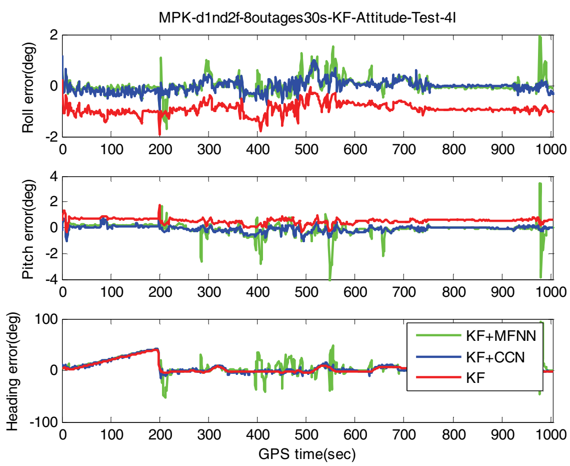

Figure 2: ANN-KF attitude test results (Tj-1).

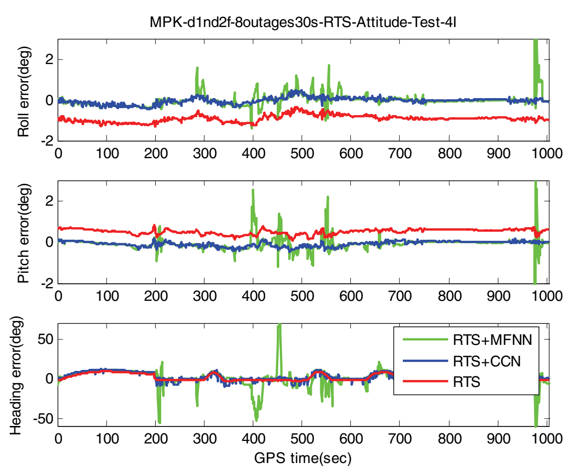

Figure 3: ANN-RTS attitude test results (Tj-1)

The networks trained from trajectory three are used to predict positional error in other trajectory. However, it may easily fail because numbers of outages are not enough. Here the training samples in trajectory three were added to twenty in order to increase the successful opportunity. The training improvements are about 85%, 90% and 93% in average in CCN-KF and 90%, 91% and 90% in average in MFNN-KF. They all learned the 20 GPS blockages well in training process. In Figures (2) to (3), each line is composed of eight line segments. Each line segment (about 29 points) represents one GPS outages. When there is no GPS blockages, the position error are close to zero due to GPS provide excellent position solutions. The method that cuts off GPS outages information to training sample is only being used to predict error in other trajectory. This way can make sure that the networks output will not affect the solutions under no GPS blockages.

Table 2: Testing results summary

Method 4I predict	POS	RMS value			Improvement (%)	
		original	residual CCN	residual MFNN	Against KF(RTS) CCN	Against KF(RTS) MFNN
	North(m)	4.4081	3.4422	4.5573	22	-3
Tj-1	East(m)	10.5150	9.9911	7.7041	5	27
(KF + CCNs)	Height(m)	2.5148	1.9186	2.6232	24	-4
(KF+MFNNs)	Roll(deg)	0.9432	0.2388	0.3205	75	66
	Pitch(deg)	0.5614	0.2385	0.5094	58	9
	Heading(deg)	10.9320	11.5690	14.8860	-6	-36
	North(m)	9.0268	8.6431	5.7135	4	37
Tj-2	East(m)	9.9548	7.8694	11.7520	21	-18
(KF+CCNs)	Height(m)	3.3027	3.1350	3.2281	5	2
(KF+MFNNs)	Roll(deg)	0.9436	0.1646	0.3329	83	65
	Pitch(deg)	0.5174	0.1543	0.4019	70	22
	Heading(deg)	4.1803	5.1983	4.3370	-24	-4
	North(m)	0.3235	0.2820	0.3048	80(13)	79(6)
Tj-1	East(m)	0.2311	0.2226	0.2551	86(4)	84(-10)
(RTS+CCNs)	Height(m)	0.4149	0.4077	0.4855	70(2)	65(-17)
(RTS+MFNNs)	Roll(deg)	0.8692	0.2311	0.3502	75(73)	63(62)
	Pitch(deg)	0.6675	0.1875	0.5415	67(72)	4(40)
	Heading(deg)	3.5811	3.7488	7.2621	66(-5)	34(-205)
	North(m)	0.2719	0.2670	0.2569	89(2)	89(6)
Tj-2	East(m)	0.3080	0.3027	0.2720	67(2)	71(12)
(RTS +CCNs)	Height(m)	0.2686	0.2514	0.2837	66(6)	61(-6)
(RTS+MFNNs)	Roll(deg)	0.8698	0.1360	0.3738	86(84)	57(57)
	Pitch(deg)	0.6631	0.1183	0.4157	77(82)	38(37)
	Heading(deg)	2.7187	4.9899	9.4589	-19(-84)	-164(-248)

Table 2 illustrates the improvements produced by the proposed ANN-RTS smoother scheme. The proposed ANN-RTS smoother scheme improve all the errors of roll angles, pitch angles and heading angles estimated by the KF by 80%, 75% ,and 14% in average, respectively. In addition, all of the improvements in positional POS parameters reach 76% in average comparing to the KF. On the other hand, the proposed ANN-RTS smoother scheme improve all the errors of roll angles and pitch angles estimated by the RTS smoother by 79% and 77% in average, respectively. In addition, all of the improvements in positional POS parameters reach 5% in average comparing to the RTS smoother.

The proposed ANN-RTS smoother scheme improves all the errors of POS parameters estimated by the KF and RTS smoother significantly for the MEMS systems. Among those POS parameters compensated by proposed ANN-RTS smoother scheme, the improvement for the orientation parameters is more significant than positional parameters. Consequently, for the low cost MEMS system with proposed ANN-RTS smoother compensation, the POS parameters estimated by RTS smoother can be improved to the level of using a medium tactical grade system.

4. CONCLUSIONS

This study developed an ANN embedded POS algorithm to reach higher estimation accuracy of POS parameters using a novel procedure that combines a SGN architecture and RTS smoother for post-mission processing. The ANN architectures were first trained to learn the error behavior of the KF and RTS smoother using one of the field data sets collected with a tactical grade INS/GPS integrated system. Then, the well-trained to schemes were verified using the rest of the test data sets. The preliminarily results that indicate the proposed ANN-KF compensation scheme is able to improve the accuracies of positional components as well as orientation components. In addition, using SGN has the advantage of higher stability than using MFNN. MFNN usually generate large undesirable output because of different level from other data sets. Although the improvements in heading errors are not all positive, the SGN has less wrongful prediction than MFNN.

In this study, the SGN performances reach the same goal of applying MFNN in compensating POS parameters. It starts from minimum topology and learning knowledge in the new neurons one by one. It has the advantage of less try and error, stability output, higher nonlinear characteristic and quicker learning process. The variation in input vectors can make MFNN generated different performance. In preliminary experiments, MFNN have worse performance when the input vectors are complex (more than four dimensions). But in CCN, more input vectors can be applied to teach the SGN to be smarter and make the right prediction about errors in position. It also learns quicker than MFNN-RTS and required less pre-required knowledge in training process. The growing process of learning new knowledge is also carry out in this study. The preliminary results verity the SGN has less moving target problems than MFNN.

This study improves the accuracy of POS parameters through evolving the POS algorithms instead of taking the direct route by using a tactical grade IMU or higher. Of course the replacement of a low cost MENS IMU with a tactical grade IMU or higher can enhance the performance of POS directly, however, the availability of tactical grade IMUs or higher is limited in terms of cost or government regulation. For low cost MEMS based integrated systems with the proposed CCN-RTS smoother scheme, the accuracies of the POS parameters estimated can be improved to the level of medium tactical grade system. Therefore, future inclination of low cost MENS based integrated systems for land based MMS applications can be anticipated with sufficient accuracies of POS parameters required for direct geo-referencing procedure and with reduced costs for the hardware used.

ACKNOWLEDGMENTS

The authors would acknowledge Dr Naser El-Sheimy and Dr. Xiaoji Niu from the MMSS group at the Department of Geomatics engineering, the University of Calgary are acknowledged by providing the field test data sets applied in this research. Dr. Eun-Hwan Shin is acknowledged for providing the INS mechanization and INS/GPS extended Kalman filter used in this article.

REFERENCES

1. Alpaydin, E. (1991): GAL, Neural Networks that grow when theylearn and shrink when they forget, International Computer Science Institute, TR 91-032

2. Bar-Itzhack(1988), I.Y. and Berman, N.: Control Theoretic Approach to Inertial Navigation System, AIAA Journal of Guidance, Control & Dynamics, Vol. 11, 237-24

3. Bishop, C.M(1995).: Neural Networks for Pattern Recognition, Oxford University Press

4. Brown R.G(1992). and Hwang P.Y.C.: Introduction to random signals, John Wiley and Sons, New York, 1992

5. Chang, H.W. and Li, C.Y. (2008): An ANN Embedded POS Algorithm for a low cost MEMS INS/GPS Integrated System, GNSS 2008, ION, Savannah, GA, USA

6. Chiang, K. W. Noureldin, A. and El-Sheimy, N. (2003): A New Weight Updating Method for INS/GPS Integration Architectures Based on Neural Networks, Measurement Science and Technology Vol.15**,** No.10, 2053-2061

7. Chiang, K.W. (2004): INS/GPS Integration Using Neural Networks for Land Vehicular Navigation Applications, Department of Geomatics Engineering, the University of Calgary, Calgary, Canada, UCGE Report 20209

8. El-Sheimy, N. (1996): The Development of VISAT - A Mobile Survey System for GIS Applications, Ph.D. thesis, UCGE Report No. 20101, Department of Geomatics Engineering, the University of Calgary

9. El-Sheimy, N. (2002): Introduction to Inertial Navigation, ENGO 699.71 lecture notes, Geomatics department, University of Calgary

10. Gelb, A. (1974): Applied Optimal Estimation, The Analytic Science Corporation

11. Shin, E.H. and El-Sheimy, N. (2005): Navigation Filter Design for Pipeline Pigging, Journal of Navigation (2005), 58: 283-295 Cambridge University Press

12. Fahlman, S.E. and Lebiere, C. (1990): The cascade learning Architecture, In D.S. Touretzky (Ed.), Advances in Neural Information Processing System 2, Denver, CO, Morgan Kaufmann Puvlishers, 198-209

13. Haykin, S. (1999): Neural Networks: A Comprehensive Foundation, 2nd Edition, Prentice-Hall, New Jersey

14. Shin, E.H. (2005): Estimation Techniques for Low-Cost Inertial Navigation, Ph.D. thesis, UCGE Report 20219, Department of Geomatics Engineering, The University of Calgary

15. Vanicek, P. and Omerbasic, M. (1999): Does a navigation algorithm have to use Kalman filter?,Canadian Aeronautics and Space Journal, Vol. 45, No. 3

16. Lin, C.Y. (2008): The Development of Hybrid INS/GPS integration Schemes Using Artificial Neural Network, Master’s thesis, Department of Geomatics, National Cheng Kung University

17. Park, M. and Gao, Y. (2008): Error and Performance Analysis of MEMS-based Inertial Sensors with a Low-cost GPS Receiver, Sensors, 8, 2240-2261.