Inertial Sensor Based Recognition of 3-D Character Gestures with an Ensemble of Classifiers

Jong K. Oh, Sung-Jung Cho, Won-Chul Bang, Wook Chang,
Eunseok Choi, Jing Yang, Joonkee Cho, Dong Yoon Kim
Samsung Advanced Institute of Technology
PO Box 111, Suwon, 440-600
South Korea
{jong.oh, sung-jung.cho, wc.bang, wook.chang, eunseok.choi, jing.yang, handle.cho, kdy2891}@samsung.com

Abstract

We present a 3-D input medium based on inertial sensors for on-line character recognition and an ensemble classification scheme for the recognition task. The system allows user to write a character in the air as a gesture, with a sensor-embedded device held in hand. The kinds of sensors used are 3-axis accelerometer and 3-axis gyroscope generating acceleration and angular velocity signals respectively. For character recognition, we used the technique of FDA (Fisher Discriminant Analysis). We tried different combinations of sensor signals to test the recognition performance. It is also possible to estimate a 2-D handwriting trajectory from the sensor signals. The best recognition rate of 93.23%, in case we use only raw sensor signals, was attained when all 6 sensor signals were combined. The recognition rate of 92.22% was reached if the estimated trajectory was used as input. Finally we tested the ensemble method and the generalization rate of 95.04% was attained on the ensemble recognizer consisting of 3 FDA recognizers based on acceleration-only, angular-velocity-only and handwriting trajectory respectively.

Keywords: Gesture recognition, on-line character recognition, inertial sensors, accelerometer, gyroscope, trajectory estimation, ensemble fusion method, Fisher discriminant analysis.

1. Introduction

Since the beginning of on-line handwriting recognition field, the category of tablet digitizer devices has been the de facto standard medium of input [8]. A tablet digitizer consists of an electronic pen and a pressure or electrostatic-sensitive surface on which a user forms one's handwriting. Tracking and sampling the movement of the pen-tip, a digitizer is able to register the information about the user's handwriting. The most important is the representation of the handwriting trajectory in the form of a sequence of *x* and *y* coordinate-pairs. Essential among other information include the pen-down and pen-up signals. Pen-down signal is generated when the electronic pen touches the digitizing surface and pen-signal when the user lifts the pen from the surface. The two signals are necessary to define a stroke in on-line handwriting, which is a sequence of points sampled from the pen-down signal to the pen-up signal. With the definition of an on-line stroke, a word is a sequence of strokes and any handwriting is a sequence of words in on-line handwriting.

The analogy between the tablet digitizer and the time-honored pen and paper is a merit of the former because users are familiar with the latter, yet can be indicative of the technology's vulnerability to the similar kind of limitations of pen and paper. Most prominently, the digitizer is tied down on a 2-dimensional writing area of limited size. In general, on-line handwriting needs to be significantly larger than a conventional handwriting on paper for reliable recognition, but the area of a digitizer tablet is not much larger than a sheet of paper in most cases. The size of the digitizer surface needs to get even smaller, because of the mobility requirement, in mobile or ubiquitous computing environments where the on-line handwriting recognition technology may argue for competitiveness over the keyboard that dominates inputting nearly unchallenged on the desktop. Writing a letter or word on the tiny display of a typical PDA, for example, is neither easy nor likable to many users. It would be clearly desirable in such situations if the medium of on-line handwriting input is not limited to the confines of a 2-dimensonal area, and this paper addresses one such possibility using the inertial sensors, namely accelerometer and gyroscope, for transducing the input. That way the user forms one's writing in the air as gesture with the input device embedding the sensors held in hand. The inertial sensors detect the user's hand motion in terms of acceleration and angular velocity signals that can be used directly or be transformed to 2-

Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)
0-7695-2187-8/04 $20.00 © 2004 IEEE

dimensional handwriting trajectory for the gesture recognition. For recognition, various different combinations of sensor and trajectory information have been tested using a recognizer based on FDA (Fisher Discriminant Analysis). Also we have tested an ensemble of FDA recognizers based on different information and have come to verify that ensemble approach improves the accuracy of classification.

In the rest of the paper, we will first describe the inertial sensors: their characteristics, issues and the trajectory estimation. Next we will present the FDA based recognizer and an ensemble of FDA recognizers trained on different sources of information, followed by the description of the experiments and the performance results.

2. Inertial Sensors, Their Signals and Characteristics

In our lab, we have developed a proprietary test-bed device embedding a suite of inertial sensors, a processor, memory, and an infra-red communication port, in the form factor of a small hand-held wand (Fig. 1). The motivation was to let its user make a gesture in the air with the wand, drawing the shape of a symbol from a pre-defined alphabet. The wand comes with a button and the user presses it to start making a gesture. So the button-press is equivalent to the pen-down signal of a traditional digitizer. With the button kept pressed, the user makes a gesture and upon finishing releases the button, the release having the effect of a pen-up signal. The system activates the wand's sensor chips or IMUs (Inertial Measurement Units) at the button-press and retrieves the measurements at each sampling time.

Fig. 1: A wand-like input device embedding inertial sensors

Upon user's finishing the gesture, the embedded CPU processes the sensor signals collected during the gesture input, performs the recognition, and sends wirelessly (via the infra-red port) the control command corresponding to the recognized symbol to an appliance nearby like a computer, TV, audio-player, room lighting system, etc. So it is a kind of self-contained universal remote controller, and can also serve as a stand-alone 3D pen-like character input system.

The inertial sensors we use come in two categories. One is the accelerometer measuring the translational movement of the hand (assuming the user holds the wand in hand) and the other the gyroscope measuring the rate of angular change of the hand's rotation. For 3-D measurement, each kind of sensor needs 3 components each representing the x, y, and z axes. So we have 6 different sensor signals and at each sampling time a vector of 6 measurement numbers is generated.

One idiosyncrasy concerning the accelerometer is that its measurement always includes the gravity. This leads to the need of compensation measure to take into account the changes in the gravitational components of the acceleration measurements in the axes. Another issue concerning the inertial sensors is the problem of drift. The fundamental cause of the drift is typically rooted at the sensor manufacturing process and there is no known general way of eliminating it altogether. The drift value can be either positive or negative and the magnitude of drift can change gradually over time and also over the temperature change in the sensor itself. The actual sensor output thus includes a certain amount of drift error and without a proper correction the output can be misleading.

With the availability of only inertial sensor information for the purpose of character recognition, we naturally face at least two possibilities. One is using the sensor signals directly, after some normalization. The raw sensor signals may not seem intuitive visually, yet it can be verified by experimentation that they carry enough discriminative information for the character recognition task. A positive aspect of this approach is that using the sensor signals is less vulnerable to the distortions introduced by further applying imperfect processing steps. We will see an example of this shortly when we talk about the 2-D trajectory estimation. The other possibility is deriving the conventional 2-D trajectory corresponding to the gesture made in 3-D space. This process is called trajectory estimation. With the restored trajectory available, a wealth of conventional feature extraction methods of handwriting recognition can be employed. The problem is that the estimation process often inserts various errors during processing so that the output trajectory can be distorted and noisy. Notwithstanding, combining the trajectory information in a fusion with others can increase the performance as we will see later. Now it is clearer to get the rationale of employing an ensemble method for our task. That is, we have a set of different sources of information none of which alone is discriminative or reliable as much as possible, and each of which has some to offer that can make up for what the others lack. So the situation naturally leads to the direction in which an ensemble of different classifiers

based on compensative information, work together to achieve a higher level of performance than what none of the component classifiers can do by itself. We will show later that the synergy from the ensemble approach is indeed productive.

2.1. Trajectory Estimation

In an ideal condition, the theory of INS (Inertial Navigation System) offers the solution to the problem of accurately estimating the 3-D trajectory of a moving object, assuming the availability of information from 3-axis accelerometer and 3-axis gyroscope [4]. For more background, we need to mention the coordinate systems of the body frame (b) and the navigation frame (n) in 3-D space (Fig. 2).

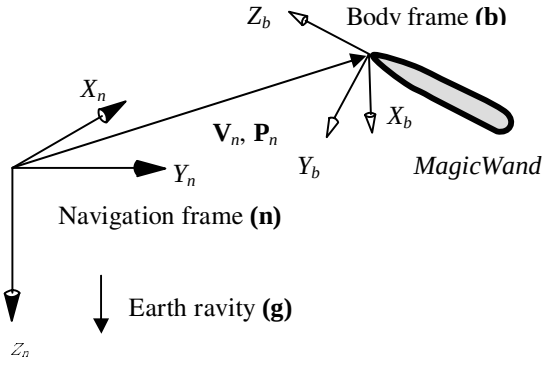

Fig. 2: Coordinate systems of INS: the body coordinates (b) and the navigation coordinates (n)

The origin of the navigation frame is the starting point of the trajectory external to the wand. X_n, Y_n, and Z_n axes are perpendicular to one another, where the direction of Z_n is parallel to the direction of the gravity (g). It does not change even when the wand is in motion. The body frame is fixed on the tip of the wand. It is aligned with the axes of the IMUs. X_b, Y_b, and Z_b axes are perpendicular to each other, where the direction of Z_b axis is aligned with the penetrating axis of the wand's body length. Therefore, it changes according to the motion of the wand.

While the user draws a gesture, the IMUs measure the acceleration $A_b = [A_{bx} \quad A_{by} \quad A_{bz}]^T$ and angular rate $\omega_b = [\omega_{bx} \quad \omega_{by} \quad \omega_{bz}]^T$ of each axis in the body frame (b). Then, using A_b and ω_b the acceleration $A_n = [A_{nx} \quad A_{ny} \quad A_{nz}]^T$ in the navigation frame (n) is calculated by the state-space equation [1, 3] of the wand. By integrating A_n twice, we obtain the handwritten trajectory $P_n = [P_{nx} \quad P_{ny} \quad P_{nz}]^T$ in the navigation coordinate (n). For more details of the INS-based derivation of the 3-D coordinates from the above, see [1, 3].

In reality, however, the situation is less than ideal because the trajectory estimation technique of INS is based on double integration of the sensor outputs and the drift error of the sensors accumulates and magnifies during the integration process. Once 3-D trajectory is attained then it is projected onto an imaginary writing plane that is optimal in the sense of minimum distortion to the original point positions [5]. The projected 2-D trajectory is the output of the estimation procedure. There have been several algorithms of trajectory estimation [1, 3, 5] and most of them are based on what is called motion detection. To deal with the drift, the main motion of the user input needs to be surrounded by a short pause at the front and another at the rear. For reliable estimation of the trajectory, however, the system needs to know the beginning and ending points of the main motion. The process of identifying the main motion in the input is called motion detection. The quality of trajectory estimation can be substantially influenced by the accuracy of motion detection which unfortunately is an imperfect process yet.

Our trajectory estimation algorithm in this paper uses only 2-axis gyroscope signal information and is based on the idea that the movement of the user's hand holding the sensor device is approximately a rotation from the joint of the hand's forearm. With this idea, we calculate the quantity for each gyroscope axis that is the multiplication of the measured angular velocity with a radius that is proportional to the angular velocity. We need to integrate the quantity only once over time to get to the estimation of the trajectory. This technique is simpler than most others yet has the effect of lessening the drift error accumulation over time since it involves just a single integration. Moreover it does not need the motion detection. This point brings more ease and convenience to the user because inserting artificial pauses demands a non-trivial degree of attention to most users. In more detail, let $p(t) = [x(t), y(t)]$ be the x and y coordinates pair to be restored at time t. In theory,

$$p(t) = \int_0^t v(\tau)\, d\tau .$$

Approximately,

$$\begin{aligned} v(t) &= [v_x(t), v_y(t)] \\ &\cong w(t) \cdot r(t). \end{aligned}$$

where $w(t)$ is the angular velocity from the 2-axis gyroscope and $r(t)$ the radius of rotation at time t. So,

$$p(t) \cong \int_0^t w(\tau) \cdot r(\tau)\, d\tau .$$

Ideally, $r(t)$ is a variable, but for our purpose we assume it a constant R, and the formulation becomes

$$\begin{aligned} p(t) &\cong \int_0^t w(\tau) \cdot r(\tau)\, d\tau \\ &\cong \int_0^t w(\tau) \cdot R\, d\tau \\ &= R \cdot \int_0^t w(\tau)\, d\tau . \end{aligned}$$

3. Character Recognition on Raw Sensor Signals by FDA

FDA is one of the linear projection methods that project the input point (a vector) in the input space to a point in the feature space. One motivation of using a linear method was that the training is easier, faster and requires relatively smaller amount of data for reasonable level of training than the more resource-intensive techniques like neural networks or hidden Markov models. Therefore it expedites, as a fast-running test-bed, one of our purposes, which is to explore the various sensor information combinations and see how the classifier behaves on each combination. One reason for such an exploration was that we wanted to determine the best performing combination of sensors. Another reason was to identify the most economical alternatives (yet performing acceptably) in terms of the number of sensors because the less sensors we use, the cheaper. Another motivation for linear method was to reinforce the overall performance via an ensemble of simple and fast classifiers. Yet another motivation was that the approach has a potential for making a user-tailored adaptation feasible because the training runs fast and demands less on the amount of training data.

The PCA (Principal Component Analysis) is probably one of the most widely known linear projection techniques [10, 9]. The essence of PCA is the construction of the projection matrix that defines the linear mapping having a scattering effect in the feature space and thus facilitating the separation between the classes. One problem with PCA, however, is that it has no provision built into the linear projection that can take the class-specific regularity into account. The projection matrix is constructed with reference to the single global mean and the scattering effect in the projection space is indiscriminate of the classes. More specifically, the projection widens the between-class scatter but also the scatter within a class and this is not desirable for classification purpose. FDA is a technique that addresses the problem by trying to maximize the between-class scatter and minimizing the within-class scatter by reflecting the class-specific distribution structure into the projection. This point was demonstrated by a simple two-class experiment in [2] where PCA partially mixed up the two classes in the projection space while FDA yielded a clean-cut separation. FDA was successfully used in [2] for improving the performance of a face recognition task under extensive variation in lighting conditions and facial expressions and in [7] for continuous on-line handwriting recognition.

4. Ensemble Fusion, Experiments and Results

For the experiments, we used a dataset collected from 16 people (5 females and 11 males). There were 13 character classes: 10 digits and 3 gestures. Each person contributed approximately equal amount of data distributed equally across the classes and the dataset contained a total of 4,945 samples. Each class character had a Graffiti-like uni-stroke shape (Fig. 3).

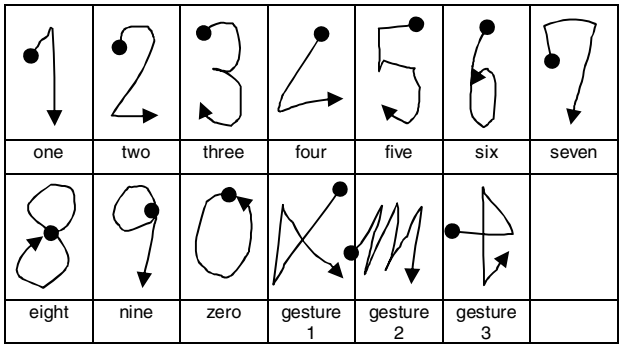

Fig. 3: Character alphabet and corresponding set of shapes

Firstly we have tested various different sensor information configurations with the dataset to see how they behave in terms of the generalization capability. 70% of the data (from each subject) was used for training and 30% for testing. Table 1 shows the result. As expected the full 6-axis configuration (3A-3G) was the top performer. There are, however, other configurations performing near the top. The best 2A-3G and 3A-2G configurations took the second and third places respectively, followed closely by the best 2A-2G configuration (2A (xy)-2G (xy)). In case we are allowed only one kind of sensor information, the table hints that the full gyro-only configuration (i.e. 3G at rank-9) would beat the full accelerometer-only configuration (3A at rank-12). If only one-sensor two-axis configurations are available, the best 2A configuration (i.e. 2A (xy) at rank-10) would outperform the best 2G configuration (2G (xy) at rank-15). It is also notable that the best 2A (at rank-

10) was slightly better than the full 3A (at rank-12) but the full 3G (at rank-9) outperformed the best 2G (at rank-15) significantly. These characteristics would help decide on a sensor configuration choice in case we have practical restrictions on the available sensor information.

Gyro / Acc	none	xyz	xy	yz	xz
none	n/a	91.15%	88.93%	64.65%	62.24%
xyz	90.61%	93.23%	92.69%	90.68%	92.02%
xy	90.81%	92.76%	92.56%	92.56%	91.82%
yz	70.29%	85.18%	87.32%	84.98%	69.68%
xz	83.97%	89.74%	89.67%	85.04%	91.21%

Table 1: Generalization rates of sensor configurations

Next we demonstrate one of our earlier point that FDA-based recognizer takes smaller amount of training data than most other techniques. In the above setup we used 70% of the data for training and the rest for testing. In the new setup we varied the percentage of the data for training from 10% to 90 % in steps of 10% increment. For compiling the table, 3A-3G configuration was used for data (Table 2). One point to note about our dataset is that it may not be completely homogeneous. The data were accumulated into the dataset in chronological order. The order of the data in the dataset, however, may reflect something more than just temporal order of placement. For example, each subject contributed about 310 samples in one session using a novel input device in a not-so-familiar manner. So the subject might get used to using the device better than it was near the start of the session, and he/she might be more influenced by a fatigue towards the end. There is also the possibility of the sensors' internal temperature rising across several collection sessions and affecting the level of drift mentioned earlier. Therefore we used both non-permuted and randomly permuted dataset for the tests and the result indeed indicates the existence of such effects reflected in the data. The table shows that the FDA's generalization rate is robust even with a small training set if its representativeness is sustained.

Training vs testing	10% vs. 90%	20% vs. 80%	30% vs. 70%	40% vs. 60%	50% vs. 50%	60% vs. 40%	70% vs. 30%	80% vs. 20%	90% vs. 10%
No permutation	83.3%	80.6%	85.8%	86.1%	87.7%	93.3%	92.2%	93.1%	92.2%
Random permutation	88.2%	90.9%	92.2%	93.1%	92.9%	92.8%	93.1%	93.0%	90.6%

Table 2: Generalization rate of FDA with different training set sizes

Data normalization is also important in using inertial sensor signals for character recognition since the raw signals can be very noisy. We applied two kinds of normalizations: Gaussian smoothing and translation correction. To see the effect of the normalization we use, we did the 70%-for-training generalization test with the 3A-3G data (Table 3).

with both	without smoothing	without translation normalization	without both
93.23%	85.24%	92.22%	84.10%

Table 3: Effects of data normalization on generalization rate

Lastly we talk about the ensemble fusion method and the improvements thereof. We computed a set of 4 different Fisher projection kernels based on the 3-axis accelerometer-only dataset (3A), the 3-axis gyroscope-only dataset (3G), the full suite of sensor signals dataset (3A-3G) and the estimated trajectory dataset (TE) computed from the 2-axis gyroscope based algorithm described earlier. All of them were constructed with 70% of non-permuted data. Their individual generalization rates on the remaining 30% of the non-permuted data are in Table 4.

3A	3G	3A-3G	TE
90.61%	91.15%	93.23%	92.22%

Table 4: Generalization performances of single-information-source recognizers

About the fusion rule, let V_i be a vector of class scores (of class-1, class-2, etc. to the last) returned by the i-th FDA recognizer. Assume that we have k such FDA recognizers and their corresponding output vectors V_1 ,V_2 , … , V_k. Then the fusion rule we used is

$$F(\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_k) = \mathrm{CompMult}(\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_k) + \mathrm{Mean}(\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_k)$$

where CompMult(~) is the component-wise multiplication of the input vectors and the Mean(~) takes the mean of the vectors. Table-5 lists the generalization rates of 5 different ensemble recognizers each consisting of the fusion rule and a subset of the 4 kernels mentioned above. As the table shows, all ensemble fusion

<3A, 3G>	<3A, 3G, TE>	<3A-3G, TE>	<3A, TE>	<3G, TE>
93.43%	95.04%	93.36%	94.37%	94.16%

Table 5: Generalization performances of ensemble recognizers

approaches outperformed the best recognizer based on single source of information (i.e. 3A-3G configuration at 93.23%). Adding the trajectory information (TE)to 3A-3G configuration, however, did not boost the performance significantly over the 3A-3G-only information. It is worth noting that an ensemble fusion of recognizers each based on different source of information performed better than a single recognizer whose input was the merger of the same sources. For example the ensemble recognizer of 3A and 3G got higher generalization rate than the one based on 3A-3G. The best performance of 95.04% rate came when we set up each recognizer on 3A, 3G and TE respectively and merge their outputs in the ensemble fusion. It is beyond the scope of this paper to analyze quantitatively and generalize the observation but the intuition is that specializing each component recognizer for a single kind of information and then integrating their behaviors synergistically leads to better performance than a best single recognizer handling all information simultaneously.

5. Conclusion

In this paper we introduced a 3-D input device for on-line character recognition using inertial sensors and presented the viability of freeing the character input from the confines of limited 2-D surface. Further developments in the direction would lead to a new dimension of usability of handwritten input especially in mobile or ubiquitous computing environments. In trajectory estimation we presented a method that does not depend on motion detection, eliminating the need of artificial pauses therefore enhancing the user convenience. We also tested various different combinations of sensor information with the FDA technique to see how they affect the recognition performance. We conclude that as much diverse sources of information as possible lead to the best overall performance but also identified the viable alternatives in case of facing practical limitations on the available sensors. With a set of different sources of information none of which alone is discriminative as much as possible, we applied the ensemble fusion to the problem with the simple and fast classification of FDA. Our FDA-based fusion method provided an experimental evidence that an ensemble of specialized recognizers working together outperforms the best recognizer working alone on all-in-one information.

References

[1] W.-C. Bang, W. Chang, K.-H. Kang, E.-S. Choi, A. Potanin, and D.-Y. Kim, "Self-contained Spatial Input Device for Wearable Computers," in Proc., 7th IEEE Int. Symp. on Wearable Computers, pp. 26- 34, 2003.

[2] Peter N. Bellhumeur, Joao P. Hespanha, David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," IEEE Trans. PAMI, vol. 19, no. 7, July 1997.

[3] W. Chang, J. Yang, E.-S. Choi, W.-C. Bang, K.-H. Kang, S.-J. Cho, and D.-Y. Kim "A Miniaturized Attitude Estimation System for a Gesture-based Input Device with Fuzzy Logic Approach," in Proc, 4th Int. Symp. on Advanced Intelligent Systems, pp. 616-619, September 2003.

[4] D. B. Cox, "Integration of GPS with Inertial Navigation Systems," Navigation, Vol. 25, pp. 236-245, 1978.

[5] G. Dissanayake, S. Sukkarieh, and H. Durrant-Whyte, "The Aiding of a Low-cost Strapdown Inertial Measurement Unit Using Vehicle Model Constraints for Land Vehicle Applications," IEEE Transactions on Robotics and Automation, vol. 17, pp. 731–747, October 2001.

[6] R. Duda, P. Hart, Pattern Classification and Scene Analysis, New York: Wiley, 1973.

[7] Jong K. Oh, Davi Geiger, "On-line Handwriting Recognition with Fisher Discrimination and Hypotheses Propagation Network," in Proc. Int'l Conf. Computer Vision and Pattern Recognition, May 2000.

[8] R. Plamondon and S.N. Srihari, "On-line and Off-line Handwriting Recognition: A Comprehensive Survey," IEEE Trans. PAMI, Vol. 22 No. 1, pp. 63 –84, 2000.

[9] M. Turk, A. Pentland, "Face Recognition Using Eigenfaces," in Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp.586-591, 1991.

[10] S. Watanabe, "Karhunen-Loeve expansion and factor analysis," Theoretical Remarks and Applications, in Proc. 4th Prague Conf. On Information Theory, 1965.