Regular paper

Journal of the KIECS. pp. 69-80, vol. 14, no. 1, Feb. 28. 2019, t. 93, pISSN 1975-8170 | eISSN 2288-2189
http://dx.doi.org/10.13067/JKIECS.2019.14.1.69

멀티모달 센서 시스템용 유전자 알고리즘 보정기 및 PnP 플랫폼

이재학[*] · 김병수[*] · 박현문[*] · 김동순[*] · 권진산[**]

Genetic Algorithm Calibration Method and PnP Platform for Multimodal Sensor Systems

Jea Hack Lee[*] · Byung-Soo Kim[*] · Hyun-Moon Park[*] · Dong-Sun Kim[*] · Jin-San Kwon[**]

요 약

본 논문은 PnP(plug and play) 기술을 지원하는 멀티모달 센서 플랫폼을 제안하였다. PnP 기술은 센서 모듈이 연결이 되면 자동으로 인식하여 응용프로그램을 사용하여 손쉬운 센서 제어를 제공한다. 멀티모달 플랫폼을 검증하기 위해, 펌웨어를 사용하여 센서를 실험하였다. 센서 모듈이 연결되면 펌웨어는 센서 모듈을 인지하여 센서 데이터를 읽는다. 따라서, PnP 기술 지원을 통해 소프트웨어 설정 없이 자동으로 센서를 연동할 수 있게 된다. 측정한 센서 데이터는 다양한 왜곡에 의해 오류를 가지고 있다. 따라서, 본 논문은 다항식 계산을 통해 센서의 오류를 보상하고자 한다. 다항식 보상기의 계수를 찾기 위해 유전자 알고리즘 방식을 사용하였다. 실험결과 악조건에서 97%의 오류를 제거하였다. 또한, 제안하는 플랫폼은 다양한 프로토콜의 센서를 지원하기 위해 UART, I2S, I2C, SPI, GPIO를 지원한다.

ABSTRACT

This paper proposes a multimodal sensor platform which supports plug and play (PnP) technology. PnP technology automatically recognizes a connected sensor module and an application program easily controls a sensor. To verify a multimodal platform for PnP technology, we build up a firmware and have the experiment on a sensor system. When a sensor module is connected to the platform, a firmware recognizes the sensor module and reads sensor data. As a result, it provides PnP technology to simply plug sensors without any software configuration. Measured sensor raw data suffer from various distortions such as gain, offset, and non-linearity errors. Therefore, we introduce a polynomial calculation to compensate for sensor distortions. To find the optimal coefficients for sensor calibration, we apply a genetic algorithm which reduces the calibration time. It achieves reasonable performance using only a few data points with reducing 97% error in the worst case. The platform supports various protocols for multimodal sensors, i.e., UART, I2C, I2S, SPI, and GPIO.

키워드

Multimodal sensor, Calibration, Genetic Algorithm, Plug and Play
멀티모달 센서, 보정기, 유전자 알고리즘, 플러그 앤 플레이

* 전자부품연구원 (jhk507@keti.re.kr, bskim4k@keti.re.kr, kimagu@keti.re.kr, dskim@keti.re.kr)
** 교신저자 : 전자부품연구원 SoC 플랫폼연구센터
・접 수 일 : 2018. 11. 22
・수정완료일 : 2019. 01. 03
・게재확정일 : 2019. 02. 15

• Received : Nov. 22, 2018, Revised : Jan. 03, 2018, Accepted : Feb. 15, 2019
• Corresponding Author : Jin-San Kwon
Dept. SoC Plaform Research Center, Korea Electronics Technology Istitute
Email : jinsan.kwon@keti.re.kr

Ⅰ. Introduction

In the development of artificial intelligence (AI), sensor applications are utilized for a wide range of activities including medical diagnosis, electronic trading, robot control, and remote sensing. As the increase in demand and diversity of sensor applications[1-7], the need[8-9] accuracy and reliability of sensors is increasing. However, sensors are not good enough for high reliability application for a variety of reasons. First, sensors subject to heat, cold, shock, and humidity during storage, shipment and/or assembly may show a change in response[10]. In addition, some sensor technologies 'age' and their response will naturally change over time. Moreover, there are many factors such as gain, offset, nonlinearity errors[11], and etc. In order to achieve a high accuracy, a sensor should be calibrated. Researchers have recently conducted a survey into artificial neural networks such as convolutional neural network, spike neural network, and etc. However, achieving accurate data is more important to improve performance for high reliability applications.

The multimodal sensor platform supports plug and play (PnP) technology which automatically recognizes a connected sensor module and an application program easily controls the sensor. The TTAK.KO-60.0290 standard[12] is possible to separate into H/W platform developer and sensor device driver provider, when you develop a sensor node so it makes each part focus on their part and also develop more efficiently and also it can realize low-costs in industry. The multimodal sensor platform supports the PnP standard[12].

In this paper, we develop a platform which compensates multimodal sensors concurrently and supports PnP technology to simply plug sensors without any software configuration. In addition, we support various protocols for multimodal sensors, i.e., UART, I2C, I2S, SPI, and GPIO.

Ⅱ. Genetic Algorithm Calibration

There are a lot of good sensors these days, however, they still suffer from errors. In order to achieve an improved accuracy, a sensor should be calibrated in the system where it will be used. A polynomial calculation compensates the difference between ideal output and measured data of a sensor. To find coefficients for the polynomial, we introduced a generic algorithm.

The goal of genetic algorithms was to solve optimization problems in the way of bio-inspired operators such as mutation, crossover and selection. In a genetic algorithm, the population is evolved toward better solutions. Fig. 1 shows the block diagram of a genetic algorithm processor.

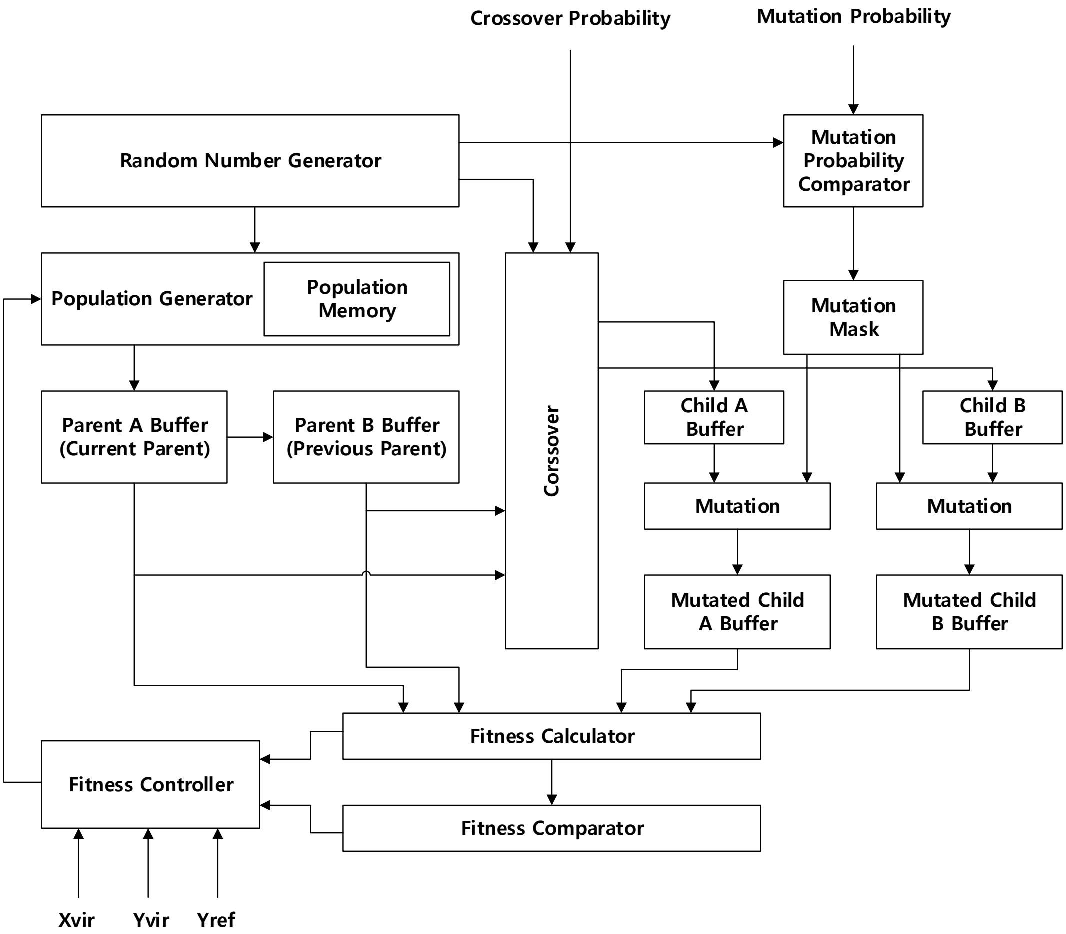


Fig. 1 Block diagram of genetic algorithm processor

2.1 Random Number Generator

A genetic algorithm employs a random number generator to operate crossover, mutation, and generation of populations. In order to implement hardware architecture, we employ a linear-feedback shift register (LFSR) which produces sequences that depend on the number of states, the feedback register connections, and initial conditions[13]. For a well-chosen feedback function, an LFSR can produce a sequence of bits that appears random and has a very long cycle. Two polynomials of the LFSRs are as follows

$$P_{16}(x) = x^{13} + x^4 + x^2 + x + 1,$$

$$P_{12}(x) = x^9 + x^7 + x + 1.$$

Fig. 2 and Fig. 3 are the logical circuits for LFSR-12 and LFSR-16. The LFSR-12 is made up of a 12-stage register for storage and shifting, XOR gates, and a feedback path from the last XOR to the input of the register.

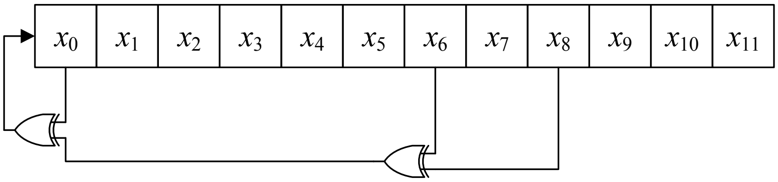


Fig. 2 LFSR-12

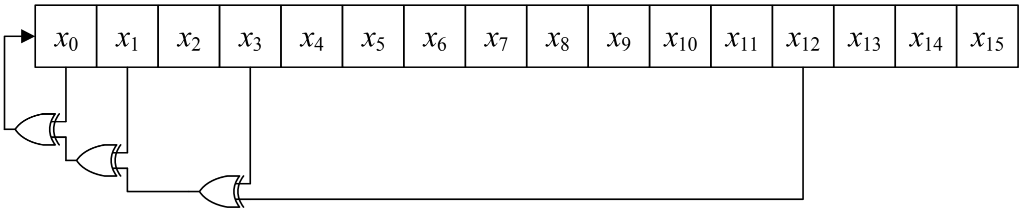


Fig. 3 LFSR-16

2.2 Population Generator

The algorithm begins by a creating initial population. There are two methods to initialize population. A random initialization generates an initial population with completely random solutions. Another method is a heuristic initialization that populates an initial population using a known heuristic for the problem. In this paper, a genetic algorithm adopts a random initialization using a LFSR-16 and stores in a population memory. The number of populations is variable and one chromosome is 32-bit fixed-point number. The arithmetic operations in the genetic algorithm are fixed-point calculations.

2.3 Crossover

A crossover operation combines the genetic information of two parents to generate new children. The two parents are randomly selected by involving a random number generator. We adapt three types of crossover (uniform type, 1-point type, and 2-point type).

A 1-point crossover picks a point randomly which designated a ‘crossover point’. The tails of two parents are swapped to get new children. For example, there are two parents (P_0, P_1) with 10 genes shown in Fig. 4. $p_{n,m}$ is the mth gene of the nth parent. Assume that a crossover point is randomly selected after the 5th gene. This results in two children (C_0, C_1), each carrying some genetic information from both parents as shown in Fig. 5.

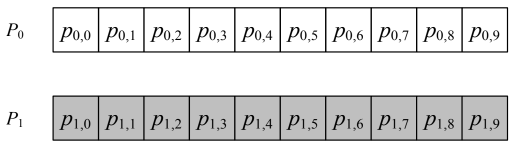


Fig. 4 Two parents (P_0, P_1)

C_0	$p_{0,0}$	$p_{0,1}$	$p_{0,2}$	$p_{0,3}$	$p_{0,4}$	$p_{1,5}$	$p_{1,6}$	$p_{1,7}$	$p_{1,8}$	$p_{1,9}$
C_1	$p_{1,0}$	$p_{1,1}$	$p_{1,2}$	$p_{1,3}$	$p_{1,4}$	$p_{0,5}$	$p_{0,6}$	$p_{0,7}$	$p_{0,8}$	$p_{0,9}$

Fig. 5 Two children (C_0,C_1) using 1-point crossover with the 5th point

A 2-point crossover randomly picks two points from the parent chromosomes. The genes in between the points are swapped between the parents[14]. Let us illustrate a 2-point crossover operation on two parents (P_0, P_1). Assume that the two points of the 2-point crossover are selected as 3rd and 6th points. The two resulting children are as shown in Fig. 6.

C_0	$p_{0,0}$	$p_{0,1}$	$p_{0,2}$	$p_{1,3}$	$p_{1,4}$	$p_{1,5}$	$p_{0,6}$	$p_{0,7}$	$p_{0,8}$	$p_{0,9}$
C_1	$p_{1,0}$	$p_{1,1}$	$p_{1,2}$	$p_{0,3}$	$p_{0,4}$	$p_{0,5}$	$p_{1,6}$	$p_{1,7}$	$p_{1,8}$	$p_{1,9}$

Fig. 6 Two children (C_0,C_1) using 2-point crossover with the 3rd and 6th points

In a uniform crossover, a genetic algorithm does not divide the chromosome into segments, rather treat each gene separately. Fig. 7 represents an example of a uniform crossover. The 2nd, 4th, 5th, and 8th genes are randomly selected and swapped. We essentially flip a coin for each chromosome to decide whether or not it'll be included in the children. We can also bias the coin to one parent, to have more genetic material in the child from that parent.

C_0	$p_{0,0}$	$p_{0,1}$	$p_{1,2}$	$p_{0,3}$	$p_{1,4}$	$p_{1,5}$	$p_{0,6}$	$p_{0,7}$	$p_{1,8}$	$p_{0,9}$
C_1	$p_{1,0}$	$p_{1,1}$	$p_{0,2}$	$p_{1,3}$	$p_{0,4}$	$p_{0,5}$	$p_{1,6}$	$p_{1,7}$	$p_{0,8}$	$p_{1,9}$

Fig. 7 Two children (C_0,C_1) using uniform crossover

2.4 Mutation

Mutation alters on or more genes (positions in a chromosome) with a probability equal to the mutation probability[14]. Assume that the nth gene from a chromosome (v) was selected for a mutation.

$$v=\{v_{N-1},v_{N-1},\cdots,v_n,v_{n-1},\cdots,v_1,v_0\}$$

If the nth gene in this chromosome is 0 (or 1), it would be flipped into 1 (or 0). So the chromosome v after this mutation would be

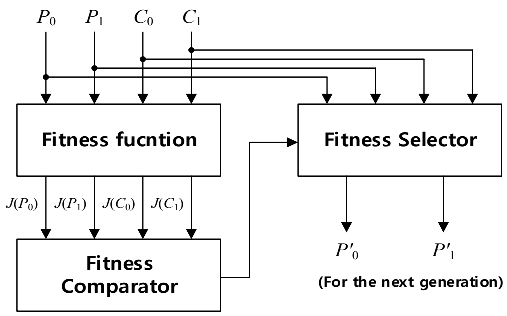


Fig. 8 Block diagram of fitness and selection

$$v'=\{v_{N-1},v_{N-1},\cdots,\bar{v}_n,v_{n-1},\cdots,v_1,v_0\}$$

, where $\bar{x}$ is not x.

2.5. Fitness and Selection

A fitness function is a computation that evaluates the quality of the chromosome as a solution to a particular problem[15]. In this paper, a chromosome represents coefficients of the polynomial for a calibration. The fitness function computes a difference between the output of the polynomial using a chromosome and its golden reference result. The generic algorithm processor computes the fitness results using two parents and two children. The fitness comparator selects two chromosomes having lower output of the fitness function than the others, and these chromosomes are new parents for the next generation. Fig. 8 shows an example of a fitness and a selection where $J(v)$ is the fitness function of chromosome v and P' is a parent for the next generation.

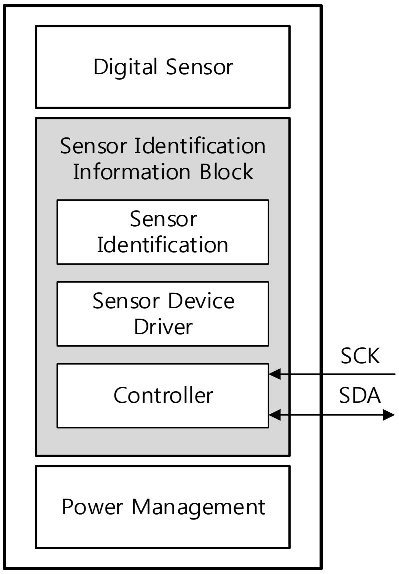


Fig. 9 Sensor module for PnP platform

III. Plug and Play Technology

The multimodal sensor platform supports PnP technology[12] which automatically recognizes a connected sensor module and an application program easily controls a sensor. A sensor module contains a digital sensor, a sensor identification, a sensor device driver, a PnP controller, and a power management shown in Fig. 9.

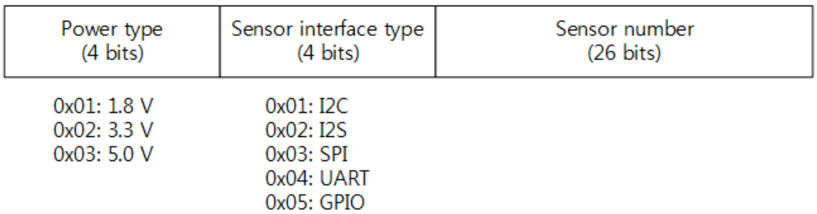


Fig. 10 Sensor identification

Fig. 10 shows a sensor identification (32 bits) which consists of a power type (4 bits), an interface type (4 bits), and a sensor number (26 bits).

To recognize a PnP module, a multimodal sensor platform requires a sensor identification from the module. Fig. 11 illustrates the recognition process. When a sensor module is connected to a multimodal sensor platform, the platform receives an interrupt signal. The platform reads a sensor information using a I2C interface shown in Fig. 12, and initializes a related bus interface.

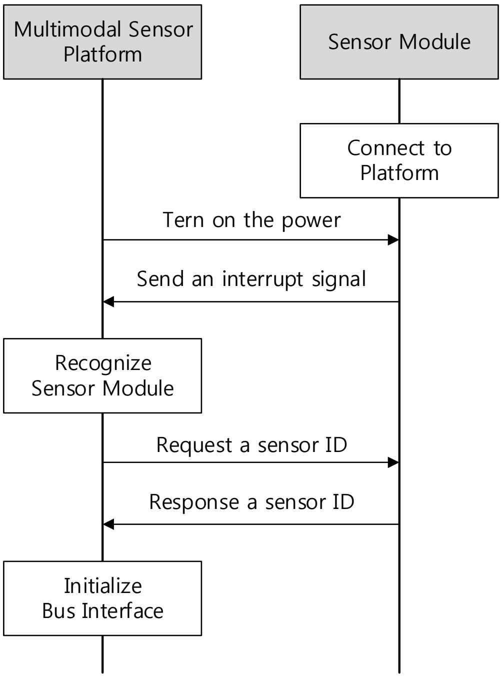


Fig. 11 Recognition process

Ⅳ. Multimodal Sensor Platform

We develop a multimodal sensor platform using a genetic algorithm and PnP technology. Fig. 13 shows a block diagram for the multimodal sensor platform. The sensor subsystem supports various interfaces, i.e., UART, I2C, U2S, SPI, and GPIO. A measured raw data arrives in the genetic algorithm processor and it is compensated to the accurate data. This platform supports a PnP system which automatically tells system software (such as drivers) where to find attached sensors, and how to operate with the sensors[10].

The interface controller sets for the bus interface and the genetic algorithm processor. To reduce processor load, we use a controller for each interface. The controller initializes a connected sensor and reads data without a processor. The sensor data is calibrated by the genetic algorithm processor. For PnP technology, each sensor module contains a sensor information block which involves its sensor identification, power type, and interface type.

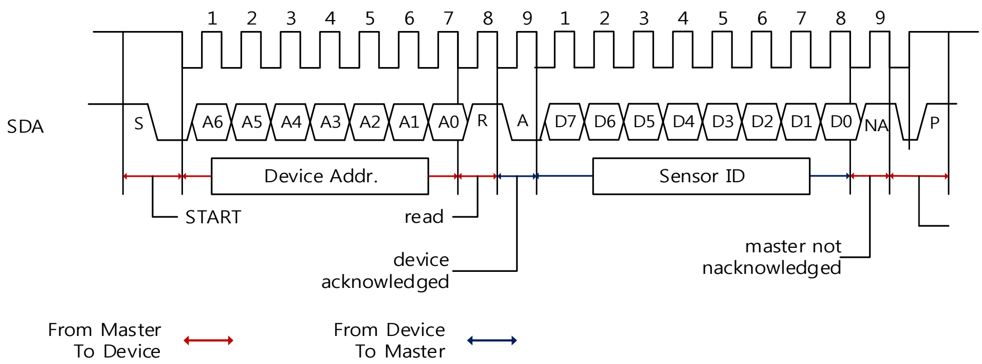


Fig. 12 Timing diagram for I2C interface

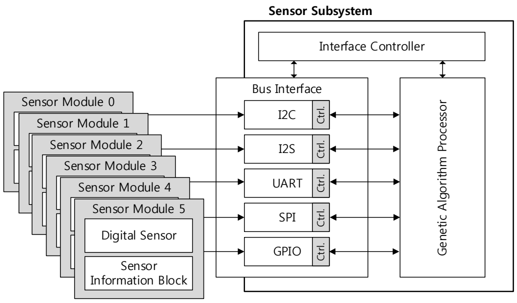


Fig. 13 Block diagram for multimodal sensor platform

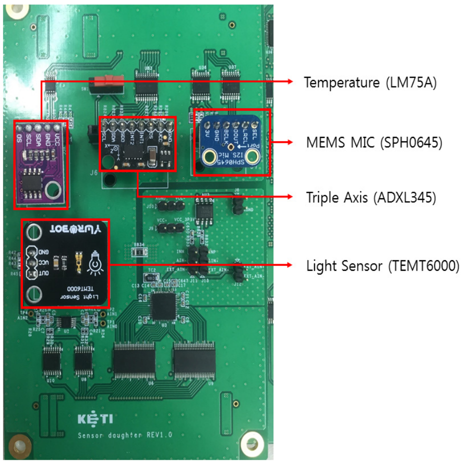


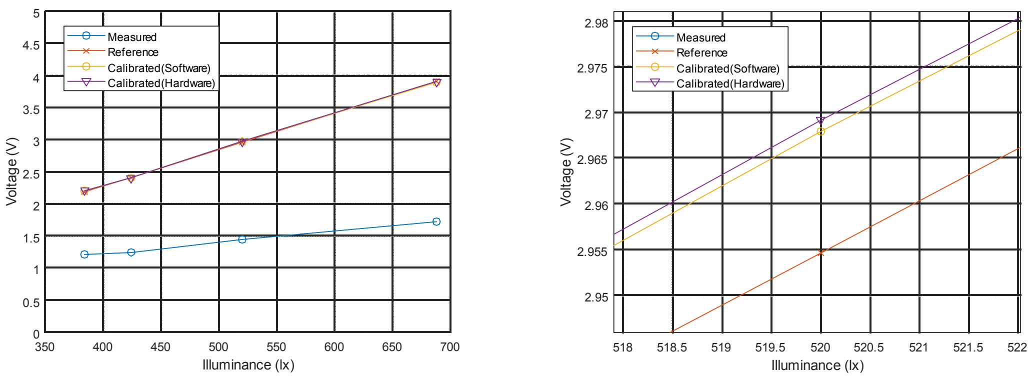
Fig. 14 Sensor board for the multimodal sensor platform

Ⅴ. Experimental Results

To verify the multimodal sensor platform, we designed the sensor board including temperature (LM75A), MEMS MIC (SPH0645), digital accelerometer (ADXL345), and ambient light sensor (TEMT6000) sensors as shown in Fig. 14. These sensors are accessible through SPI, I2C, I2S, and GPIO interfaces.

Fig. 15 shows the compensated results for an ambient light sensor (TEMT6000) sensor, where x-axis is illuminance of the immediate vicinity perceived by the sensor and y-axis is measured voltage from the sensor. To measure the reference illumination intensity, we used a light lux meter (DT-1300). As shown in Fig. 15 (a), there is the difference between the measured and reference data due to offset and nonlinearity errors. The genetic algorithm processor finds the optimal coefficients and compensates the measured data. The compensated results are almost same as the reference data as shown in Fig. 15 (b).

We verified a sensor module for PnP technology. Fig. 16 shows a firmware for the multimodal sensor platform. The yellow (or red) box indicates that the related sensor is connected (or disconnected). When a sensor module is connected on the multimodal sensor platform, a firmware



(a) (b)

Fig. 15 Compensated Results for Ambient Light Sensor (TEMT6000)

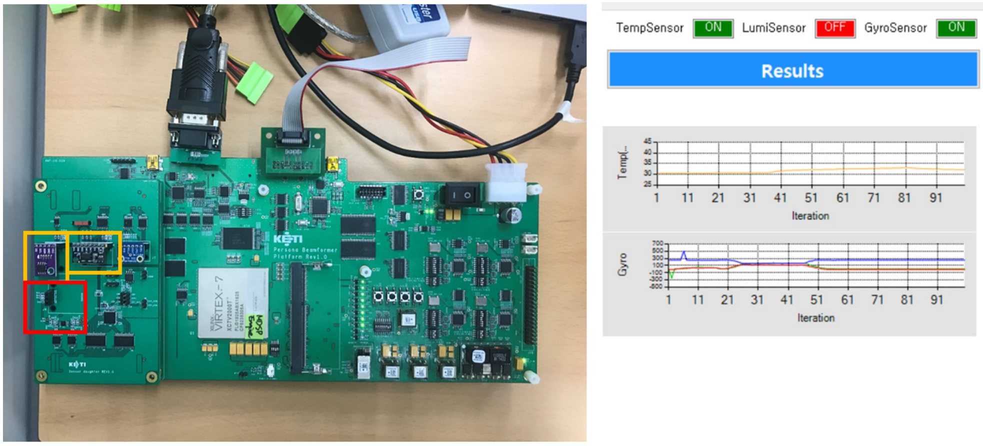


Fig. 16 Firmware for multimodal sensor platform

turns on a button (green light) on the right upper side in Fig. 16, and reads a sensor identification. Otherwise, the switch changed a red light. In Fig. 16, a temperature sensor and a gyro sensor are connected to the multimodal sensor platform. These sensors are automatically recognized and read data.

We simulated the third-degree polynomial to compensate measured data. It can reduce the measured error to 96.84% and complete computation within 25.43ms shown in Table 1.

Table 1. Simulation Results using Genetic Algorithm

Test case	Degree (order)	CPs (ea.)	Errors (%)	Calibration time (ms)
Worst	3	4	3.16	25.43
Typical	2	3	0.001	10.98

Ⅵ. Conclusion

In this paper, we developed a multimodal sensor planform that supports various bus interfaces and the genetic algorithm processor to compensate sensor data. In addition, it provides PnP technology to simply plug sensors without any software configuration. The genetic algorithm processor calibrates various sensors concurrently and it achieves reasonable performance using only a few data points with reducing 97% error in the worst case.

감사의 글

본 연구는 2019년도 중소벤처기업부의 기술개발사업 지원에 의한 연구임 [S2521302]

References

[1] Y. Qu, G. Zhang, Z. Zou, Z. Liu, and J. Mao, “Active Multimodal Sensor System for Target Recognition and Tracking,” *Sensors*, vol. 17, no. 7, June 2017, pp. 1-22.

[2] T. Korthals, M. Kragh, P. Christiansen, H. Karstoft, R. N. Jorgensen, and U. Rückert, “Multi-Modal Detection and Mapping of Static and Dynamic Obstacles in Agriculture for Process Evaluation,” *Front. Robot. AI*, vol. 5, no. 28, Mar. 2018, p.1.

[3] J. Jeonga, T. S. Yoonb, and J. B. Parka, “Multimodal Sensor-Based Semantic 3D Mapping for a Large-Scale Environment,” *Expert Systems with Applications*, *Elsevier*, vol. 105, no. 1, Sept. 2018, pp. 1-10.

[4] J. H. Lee, B. Kim, and D. Kim, “Calibration

Using Genetic Algorithm and PnP Platform Development for Multimodal Sensors," *Proc. of the 7th Int. Conf. on Next Generation Computer and Information Technology (NGCIT)*, Sapporo, Japan, Aug. 2018, pp. 23-25.

[5] S. Park, S. Oh, S. Lee, J. Maeng, and Y. Ko, "A Study on the Temperature and Humidity Control Methodology of Smart Farm ased on Wireless Communication Network," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 13, no. 4, 2018, pp. 851-857.

[6] B. Kim, J. Lee, T. Hwang, and D. Kim, "Design of Lightweight Artificial Intelligence System for Multimodal Signal Processing," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 13, no. 5, 2018, pp. 1037-1042.

[7] W. Oh, "A Fault Detection Scheme in Acoustic Sensor Systems Using Multiple Acoustic Sensors," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 11, no. 2, 2016, pp. 203-208.

[8] J. Rivera, G. Herrera, and M. Chacon, "Improved progressive polynomial algorithm for self-calibration and optimal response in smart sensors," *Measurement, Elsevier*, vol. 42, no. 9, Nov. 2009, pp. 1395-1401.

[9] S. Rahili, J. Ghaisari, and A. Golfar, "Intelligent selection of calibration points using a modified progressive polynomial method," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 9, Apr. 2012, pp. 2519-2523.

[10] H. Shah, "Internet of Things (IoT) Rollouts in the Telehealth Industry," *Telehealth and Medicine Today*, vol. 2, no. 3, 2017, pp. 1-15.

[11] G. Horn and J. L. Huijsing, "Calibration using Digital Signal Processing," *Springer International Series in Engineering and Computer Science book series*, 1998, pp. 173-192.

[12] TTAK.KO-06.0290, *Plug&Play based USN Sensor Access Interface*, 2012.

[13] B. Sklar, *Digital Communications: Fundamentals and Application.*, New Jersey: Prentice Hall, ~~New Jersey,~~ 2001.

[14] Z. Michalewicz, *Genetic algorithms + data structures = evolution programs (3rd ed.)*. Berlin: Springer, 1996.

[15] A. Zalzala and P.J. Fleming, *Genetic Algorithms in Engineering Systems.* The Institution of Engineering and Technology, 1997.

저자 소개

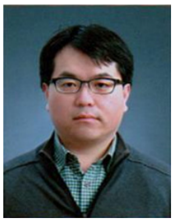
이재학(Jea Hack Lee)

2009년 아주대학교 전자공학과 졸업(공학사)
2011년 아주대학교 대학원 전자학과 졸업(공학석사)
2017년 아주대학교 대학원 전자공학과 졸업(공학박사)
2017년 ~현재 전자부품연구원, SoC 플랫폼센터 선임연구원
※ 관심분야 : VLSI, 뉴럴네트워크, 멀티모달 센서네트워크, 통신시스템

김병수(Byung-Soo Kim)

2006년 인하대학교 전자공학과 졸업(공학사)
2008년 인하대학교 대학원 전자학과 졸업(공학석사)
2013년 인하대학교 대학원 전자공학과 졸업(공학박사)
2014년 ~현재 전자부품연구원, SoC 플랫폼센터 선임연구원
※ 관심분야 : 임베디드 시스템 디자인, 멀티미디어 코덱 디자인, 인공지능 알고리즘, VLSI, SoC 디자인

박현문(Hyun-Moon Park)



2006년 국민대학교 전자통신학과 졸업(공학석사)
2010년 국민대학교 BIT학과 졸업(이학박사)
2010년~2013년 한국전자통신연구원 선임연구원
2014년~현재 전자부품연구원 SoC플랫폼센터 책임연구원
※ 관심분야 : IoT시스템, 에너지하베스팅, 엣지컴퓨팅

김동순(Dong-Sun Kim)

1997년 인하대학교 전자공학과 졸업(공학사)
1999년 인하대학교 대학원 전자학과 졸업(공학석사)
2005년 인하대학교 대학원 전자공학과 졸업(공학박사)
1999년 ~현재 전자부품연구원, SoC 플랫폼센터 수석연구원
※ 관심분야 : 유무선 통신시스템, 무선 센서 네트워크, VLSI, SoC 디자인, 멀티미디어 코덱 디자인, 컴퓨터 구조, 인베디드 시스템 디자인

권진산(Jin-San Kwon)

2011년 고려대학교 컴퓨터정보학과(학사)
2013년 고려대학교 컴퓨터정보학과 전산학(이학석사)
2013년~현재 전자부품연구원 SoC플랫폼센터 전임연구원
※ 관심분야 : 실시간 운영체제, 이기종 컴퓨팅, 뉴로모픽 컴퓨팅