JICCE Journal of information and communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 14(2): 106-114, Jun. 2016

Regular paper

Touch Position Recovery Algorithm for Differential Sensing Touch Screen

Ji-Ho Kim, Dong-Min Won, and HyungWon Kim*, ***Member, KIICE***

Department of Electronic Engineering, Chungbuk National University, Cheongju 28644, Korea

Abstract

Differential sensing methods are more effective in alleviating panel noise than single-line sensing, and thus have been increasingly used in the touch screen industry. However, they have a drawback: they tend to cancel out multiple touches and need touch position recovery algorithms. This paper introduces a novel algorithm of touch position recovery for differential sensing, which is a low-complexity but high-accuracy approach for determining multiple touch positions. We have implemented the proposed method in a touch screen controller system on a chip. In the simulation experiments using realistic touch screen models and a differential sensing circuit, the algorithm exhibited a high detection performance of a signal-to-noise ratio gain of up to 52.21 dB. Therefore, we can conclude that the proposed method is substantially more accurate than the previous method. Further, the proposed method incurs little or no overhead in terms of the detection speed and the chip size.

Index Terms: Differential sensing, Recovery algorithm, Touch controller, Touch screen panel

I. INTRODUCTION

Recently, there is an increasing demand for large touch screen panels (TSPs) in the touch screen industry. With the availability of relatively large touch screens, TSPs have been adopted for various applications such as television screens, digital signage, and medical devices. Owing to the development of these devices, most of the recent touch applications require additional functions such as a multi-touch function. Further, since a touch screen can serve as both a display terminal and an input device, it often replaces a mouse or a keyboard [1-3].

Various types of TSPs have been developed to improve the detection performance and to support a wide range of applications. Nowadays, projected-capacitance type TSPs have been increasingly adopted in most touch screen applications. Fig. 1 shows the general architecture of a projected-capacitance TSP with a human finger [4].

Projected-capacitance TSPs are further divided into the self-capacitance and the mutual-capacitance types [5-7]. Self-capacitance TSPs use rows and columns of conductors overlaid in a grid pattern, and mutual-capacitance TSPs use an array of capacitors located at each intersection of the conductor grid. Fig. 2 shows the structure of self-capacitance and mutual-capacitance TSPs. The detection method of projected-capacitance TSPs involves the measurement of the capacitance changes inferred by touched objects. In the case of self-capacitance panels, a controller detects the changes in the capacitance for each row and column. The self-capacitance TSPs, however, suffer from ghost point problems in the case of multiple touches, which is a serious drawback [8]. Mutual-capacitance TSPs, on the other hand,

Received 06 January 2016, **Revised** 19 February 2016, **Accepted** 30 April 2016

***Corresponding Author** HyungWon Kim (E-mail: hwkim@cbnu.ac.kr, Tel: +82-43-261-2399)
Department of Electronics Engineering, Chungbuk National University, 1 Chungdae-ro, Seowon-gu, Cheongju 28644, Korea.

Open Access http://dx.doi.org/10.6109/jicce.2016.14.2.106 print ISSN: 2234-8255 online ISSN: 2234-8883

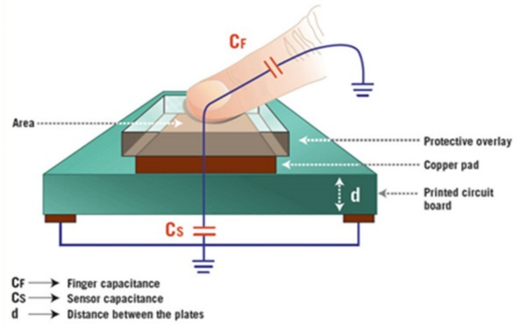


Fig. 1. Architecture of projected-capacitance TSP.

have two conductive layers for driving and sensing the signal. Since their controller measures the mutual capacitance between the two layers, mutual-capacitance type panels can better distinguish multiple touches.

To sense an RX signal from a mutual-capacitance TSP, either a single-line sensing or a differential sensing method can be used for recognizing the changes in the mutual capacitance throughout the TSP. While the early techniques usually drive TX lines sequentially, the latest techniques often drive the TX lines concurrently. As an example of the concurrent driving techniques, frequency division concurrent sensing (FDCS) [9] has been reported to provide a fast sensing speed for large touch screens.

The sensed signal from the RX lines is passed to an analog-to-digital converter (ADC) followed by a microcontroller unit (MCU), where software algorithms filter noise and calculate the touch positions [1, 2, 10-12]. The software algorithms running on MCUs, however, suffer from a critically low processing speed. Therefore, conventional software-based algorithms are often unacceptable for large TSPs. For example, the 23-inch TSP used for our experiments has 44 TX lines and 78 RX lines, and therefore, the number of touch points is 3,432. The complexity of a software algorithm rapidly increases with an increase in the number of touch points, which in turn increases with an increase in the TSP size. The speed of the touch position calculations of a TSP is, in general, calculated using Eqs. (1) and (2). Eqs. (1) and (2) express the resulting frame scan rate and report rate, respectively. Here, N_{tx} and N_{rx} denote the number of TX lines and the number of RX lines, respectively. T_{drive} represents the period of the TX driving signal. Eq. (1) shows that the time required to finish one frame scan is a function of ($N_{tx} \times N_{rx}$). Furthermore, $T_{algorithm}$ indicates the time taken to process the touch position calculation algorithm, which usually is the software running in an MCU. Eqs. (1) and (2) show that the scan rate and the report rate tend to rapidly increase with an increase in the size of the touch screen.

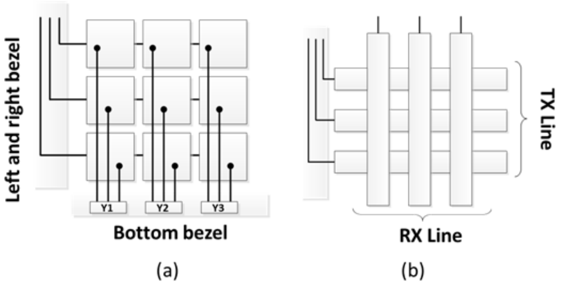


Fig. 2. Structure of (a) self-capacitance TSP and (b) mutual-capacitance TSP.

$$\text{Scan rate} \propto \frac{1}{T_{drive} \times N_{tx} \times N_{rx}}, \tag{1}$$

$$\text{Report rate} \propto \frac{1}{T_{algorithm} \times N_{tx} \times N_{rx}}. \tag{2}$$

On the other hand, although the TSP size is being continuously increased because of the market demand, the target report rate requirement has mostly remained the same or in some cases, increased to meet the tight performance goal. To reduce the processing time and increase the report rate, we need to develop new high-speed algorithms or hardware-based methods.

In this paper, we propose a touch position recovery algorithm with a differential sensing circuit. The rest of this paper is organized as follows. In Section II, we briefly discuss the existing touch screen sensing methods, and the result of sensing a TSP. The proposed algorithm and the experimental results are described in Sections III and IV, respectively. Finally, we present the conclusion in Section V.

II. EXISTING TOUCH SCREEN SENSING METHODS

A. *Single-Line Sensing Method*

The touch positions in mutual-capacitance touch screens are determined by measuring the change in the mutual capacitance. The single-line sensing method has been most widely used thus far because of its simplicity. In this method, the drive circuit applies stimulus signals to a driving (TX) line. As its stimulus signals, it often employs chopping signals, pulses, or sinusoidal signals. Then, the sensing circuit sequentially reads out the sensing (RX) lines to measure the changes in capacitance. Most of the single-line sensing circuits convert the charge in the mutual capacitors to a voltage by integrating the sensed RX signal. An ADC translates the voltage into digital values that directly indicate the mutual capacitance variation, as shown in Fig. 3 [13]. Therefore, the touch position recovery algorithm of single-line sensing is as simple as noise filters. However, this method has a disadvantage: it is very difficult to remove the common-mode noise, a dominant noise in most TSPs.

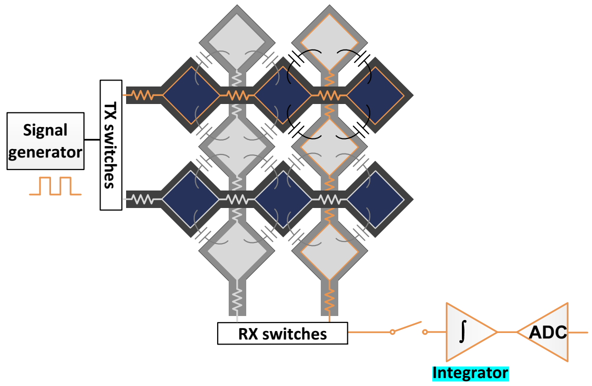


Fig. 3. Single-line sensing scheme.

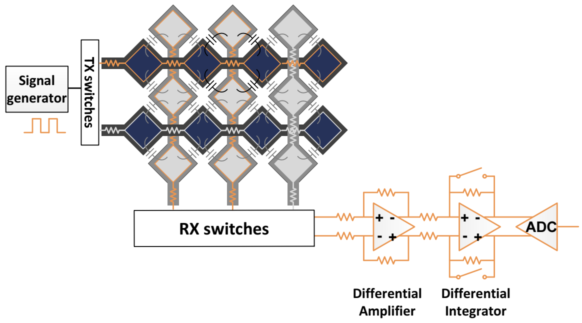


Fig. 4. Differential sensing scheme.

B. Differential Sensing Method

Another approach is a differential sensing method, which can effectively reduce the common-mode noise and has thus received more attention. Differential sensing removes the common-mode noise affecting pairs of adjacent cells (touch points) on the same TX line, which is the most prominent noise type in TSPs [7]. Fig. 4 illustrates a conventional differential sensing circuit. A differential amplifier subtracts the RX signals from each pair of adjacent cells. A differential integrator accumulates the signal difference while cancelling high-frequency noise. Once the differential sensing is carried out for all the RX line pairs, a driving signal is applied to the next TX line, and then, the abovementioned process is repeated. The differential sensing method usually provides a substantially higher signal-to-noise ratio (SNR) than the single-line sensing method. However, it suffers from the multi-touch cancellation problem. If two differential RX lines are both touched, the two touched signals cancel each other and thus, may not be detected correctly.

A differential sensing method called the delta (Δ)-integration scheme [14, 15] has been reported to detect the touched position effectively. It reads two RX lines that are Δ distance apart rather than two adjacent RX lines. However, this method treats a touched signal or an untouched signal as a binary value and counts the binary values to form a wide hill of touched areas. Therefore, it is only effective to detect a wide touched area. Further, it is not suitable for highly accurate multi-touch detection methods, which must distinguish multiple touch points having small areas and many levels of touch strength.

In this paper, we propose a new method that can accurately detect multiple touches of very small areas and a range of touch strength levels. It employs a recovery algorithm that recovers all the original touch points accurately even in the case of the multi-touch cancellation problem.

C. Frequency Division Concurrent Sensing Method

While the differential sensing method improves the SNR, it usually yields the same frame scan rate and the single-line sensing method. Recently, in order to improve the frame scan rate, concurrent sensing methods have been proposed, which drive multiple TX lines concurrently.

For example, a concurrent sensing method called FDCS [16] concurrently drives multiple sine waves of orthogonal frequencies onto all the TX lines. Then, its sensing circuit reads out each RX signal, which is the sum of all the sine waves propagated from all the TX lines. Thereafter, it determines the touch positions by analyzing the fast Fourier transform (FFT) of the sine waves [16]. Fig. 5 illustrates the operation of the FDCS method.

In addition to its high speed, the concurrent sensing method can employ differential sensing to achieve a high SNR. However, for such a concurrent differential sensing method, the multi-touch cancellation problem described above still remains as the cause of a critical TSP malfunction, a false-negative detection.

Therefore, the proposed touch position recovery method is an effective approach to both differential sensing and concurrent differential sensing methods.

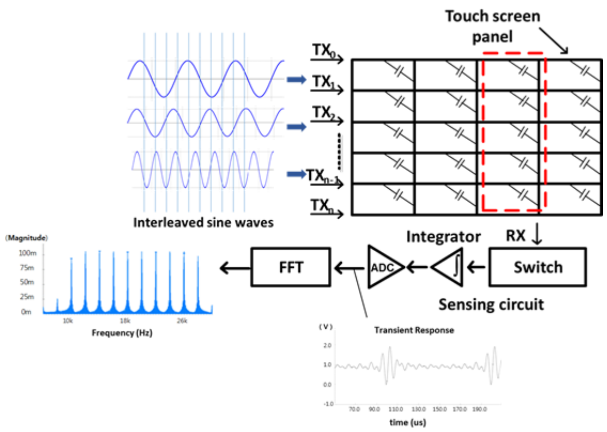


Fig. 5. FDCS scheme.

III. PROPOSED ALGORITHM

When a human finger touches a screen, not only the touched point but also its adjacent points experience changes in the mutual capacitance. This follows from the fact that TSPs usually have a small spacing between the crossing points for a high resolution. For example, Fig. 6(a) illustrates a touch point sensed using the single-line sensing method. The yellow circle indicates the center point of the touch position. The orange and the red squares denote the adjacent points affected by touch. The mutual capacitors of the adjacent points experience smaller capacitance changes than the center point. The blue squares shown in Fig. 6(a) represent untouched points. Fig. 6(b) shows the amplitude variation at the center and the adjacent points of the touch position. Fig. 6(b) has only one summit, and therefore, the software algorithm can easily determine the touch position by searching for the highest value. However, this method would not remove the noise.

Fig. 7(a) shows the result of the differential sensing method. Depending on which RX line was touched, its differential integration result can be either a positive or a negative voltage. The amplitude variation shown in Fig. 7(b) appears as one positive and one negative summit. Therefore, in contrast to the single-line sensing method, the differential sensing method needs a special algorithm to accurately recover the touch positions. One such algorithm is proposed in this section. Fig. 8 depicts a flow chart of the proposed algorithm. The algorithm starts with a differential sensing step, which iteratively scans all RX lines of the TSP. Each sensed data sample is stored in the memory after the ADC. Once the differential sensing step completes the scanning of one frame, the recovery algorithm step begins.

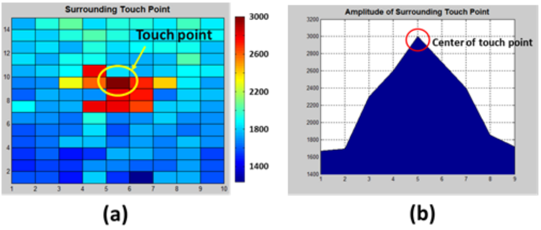


Fig. 6. (a) Part of single-line sensing result and (b) amplitude shape of touch point and adjacent pixel.

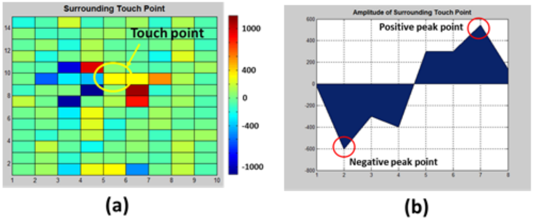


Fig. 7. (a) Part of differential sensing result and (b) amplitude shape of touch point and adjacent pixel.

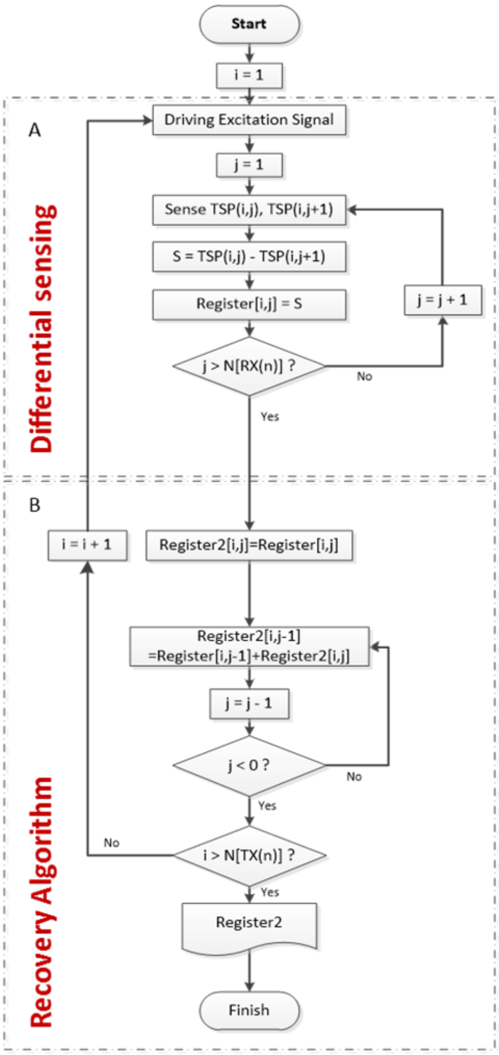


Fig. 8. Flow chart of proposed algorithm.

The remaining steps of the proposed algorithm are to find the peak values and to determine their positions as the touched points. The proposed algorithm iteratively processes two data samples and accumulates the intermediate results throughout all the RX differential data over each TX line. One of the two data samples is the intermediate result calculated in the previous iteration, and the other data sample is the next differential data sample read from the memory. The key recovery process of the algorithm involves adding these two data samples, as expressed in Eq. (3). It recovers the original touch points traversing backwards from the final differentially sensed data towards the first differentially sensed data.

$$\text{Recovery}(M_{i,j}) = \begin{cases} Diff(M_{i,j}), & if(j=n), \\ Diff(M_{i,j})+\text{Recovery}(M_{i,j+1}), & if(0\le j<n)\ . \end{cases} \qquad (3)$$

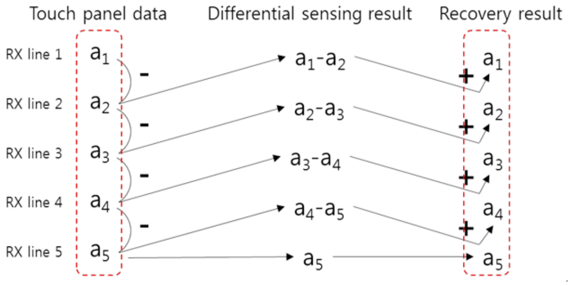


Fig. 9. Proposed touch point recovery method.

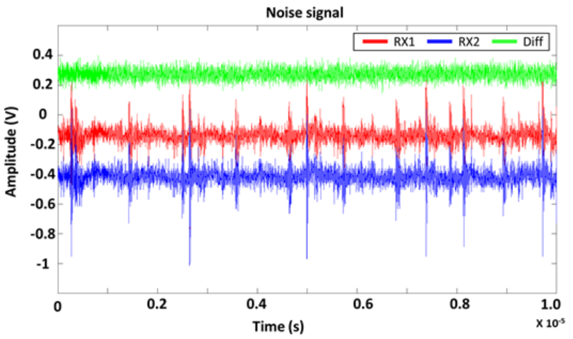


Fig. 10. Noise waves measured from real TSP.

For example, Fig. 9 illustrates how the proposed algorithm recovers the original touched points in a TSP. The first column shows a set of raw RX data for one TX line. The second column gives the intermediate results of the recovery process of Eq. (3). The third column shows the recovered RX touched data corresponding to the raw data. It is observed that the recovered results are not exactly the same as the raw data, but they have enhanced touch data with lower noise than the raw data. This enhancement is the consequence of the following advantages of the proposed method. While its differential sensing step can eliminate most of the common-mode noise, the accumulation process of its recovery step can further suppress the high-frequency random noise. An example is given in Section IV.

In addition, the proposed algorithm employs only simple accumulation operations; therefore, it can be easily implemented using low-cost hardware or possibly even software. Furthermore, it enables the simultaneous processing of an unlimited number of multiple touches with little overhead on the processing time. This is a significant advantage over the previous differential sensing approaches.

IV. EXPERIMENTAL RESULTS

The proposed algorithm has been implemented on the top of our touch screen controller system on a chip (SoC) design employing a differential sensing circuit. The differential sensing circuit in the SoC consists of a differential integrator, a rotating auto-zeroing offset cancellation circuit [17], and an ADC supporting 8 bits. The rotating auto-zeroing offset cancellation circuit can eliminate the mismatch of the differential amplifier and thus, allow an accurate measurement of the difference between two adjacent cells.

This sensing circuit has been implemented using a Magnachip/SK Hynix 0.18-µm CMOS process. To accurately evaluate its performance, we applied to the simulation the actual noise signals measured from a 23-inch commercial TSP. Fig. 10 illustrates the measured noise, where the blue and the red signals represent the noise signals measured from the RX lines RX1 and RX2, respectively. The green signal represents the differential signal obtained by subtracting the blue and the red signals.

To evaluate the operation and the performance of the proposed method, we conducted extensive simulations by using an accurate TSP model with 9 TX lines and 16 RX lines. We drove pulse waves onto one TX line, and read out each pair of RX lines sequentially. We considered two touch cases:

1) TSP has one touch point and 8 surrounding points as shown in Fig. 11(a).

2) TSP has two touch points and 18 surrounding points as shown in Fig. 11(b).

In each iteration of the differential sensing step of the proposed algorithm, the ADC receives 15 differential data samples since there are a total of 16 RX lines. In the case of Fig. 11(a), the following four differential pairs are expected to observe large differential values, while others show very small values: RX(5,6), RX(6,7), RX(7,8), and RX(8,9). Similarly, in Fig. 11(b), nine differential pairs are expected to show large differential values.

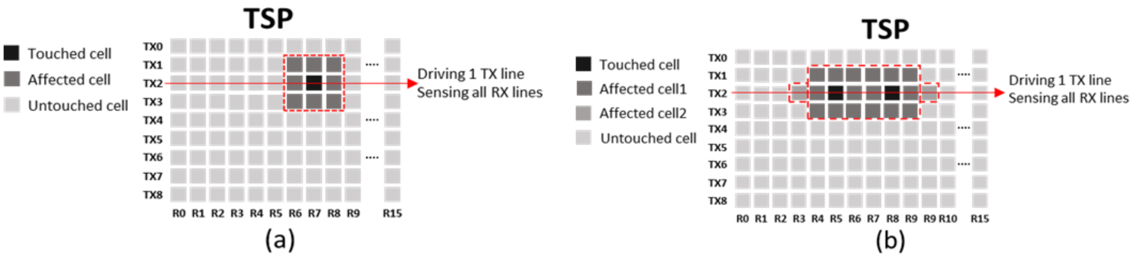


Fig. 11. Simulation experiments. (a) TSP with 1 touch point and 8 affected points and (b) TSP with 2 touch points and 18 affected points.

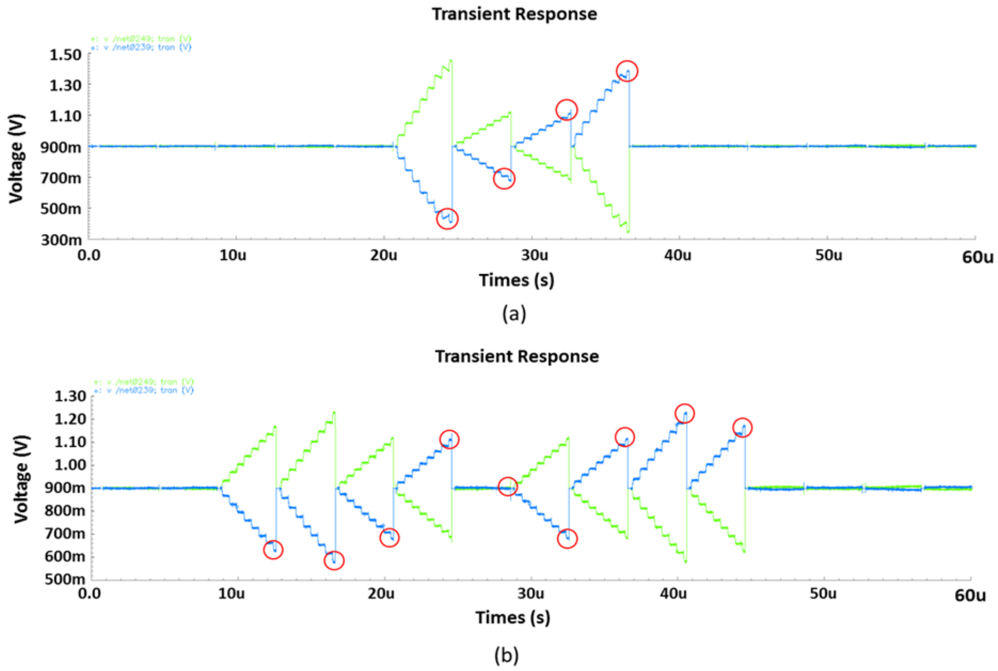
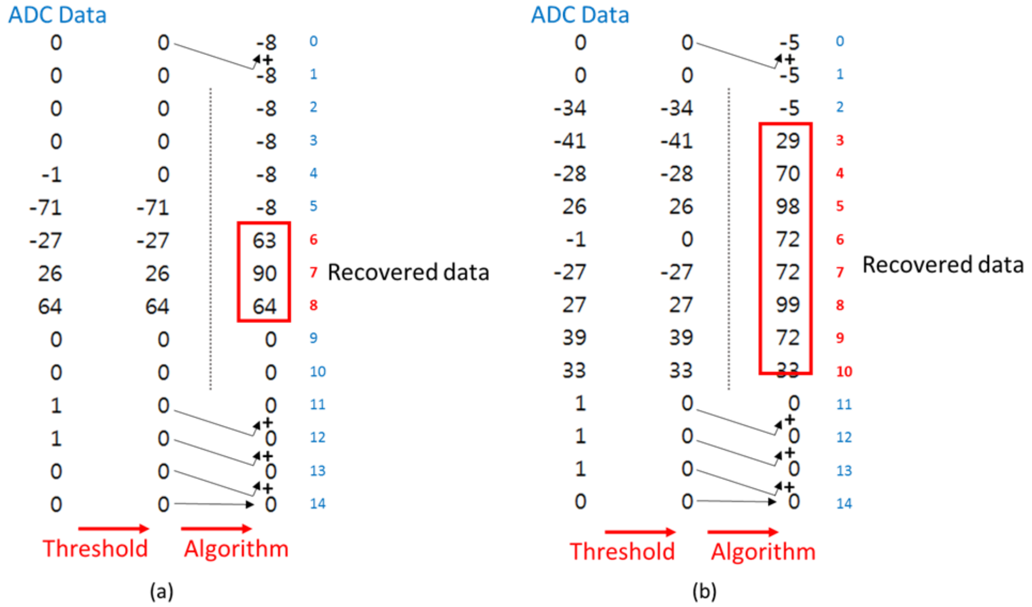


Fig. 12. Differential sensing result. (a) TSP with 1 touch point and 8 affected points and (b) TSP with 2 touch points and 18 affected points.



ADC Data			
0	0	-8	0
0	0	-8	1
0	0	-8	2
0	0	-8	3
-1	0	-8	4
-71	-71	-8	5
-27	-27	63	6
26	26	90	7
64	64	64	8
0	0	0	9
0	0	0	10
1	0	0	11
1	0	0	12
0	0	0	13
0	0	0	14

ADC Data			
0	0	-5	0
0	0	-5	1
-34	-34	-5	2
-41	-41	29	3
-28	-28	70	4
26	26	98	5
-1	0	72	6
-27	-27	72	7
27	27	99	8
39	39	72	9
33	33	33	10
1	0	0	11
1	0	0	12
1	0	0	13
0	0	0	14

Fig. 13. Example calculation results of the proposed algorithm using ADC data samples: (a) for the case of 1 touch point and 8 affected points and (b) for the case of 2 touch points and 18 affected points.

Fig. 12(a) shows the integration results of the simulation experiment illustrated in Fig. 11(a). While the four differential integration values show large values as marked by red circles, all other integration values are negligible, since their RX signal pairs have similar values. Similarly, Fig. 12(b) shows the integration results of the simulation experiment illustrated in Fig. 11(b). There are nine differential integrations that have large values. The resulting differential integration values are sampled by the ADC. Then, the sampled ADC data are processed by the proposed algorithm. Fig. 13 illustrates the calculation process of the proposed algorithm for the two examples illustrated in Fig.

11(a) and (b). In this experiment, we applied a threshold to filter out noise signals that are less than the threshold. Therefore, it can enhance the untouched signals and further improve the detection performance. After the proposed algorithm recovers the original touch strength values, we can select the peak values and their positions as the touched points.

For the example of the recovered values shown in Fig. 13(a), the peak value of 90 is the touched point. The slightly lowered values, 63 and 64 around 90, are the affected points surrounding the touched point. Note that the amplitude of the recovered values look like the amplitude of the single-line sensing case illustrated in Fig. 6. For another example, the recovered values shown in Fig. 13(b) have two peak values, 98 and 99, which indicate the two touched points.

We calculate the SNR by using Eqs. (4)–(6). These equations are commonly used for measuring the performance of the touch screen detection methods and circuits. Therefore, using these equations, we calculate the SNR gain of the proposed algorithm to be 52.21 dB (42.17 – (–10.04) dB). Here, 42.17 dB is the SNR of the recovered signals based on the proposed algorithm, while 10.04 dB is the SNR measured at the RX lines of the TSP under experiment.

Table 1 compares a single-line sensing method [13], the delta-integration method [14], and the proposed method conducted an experiment with a real TSP that consists of 44 TX and 78 RX lines. While the previous methods provide the output SNR of 28.86 dB and 39 dB, respectively, the proposed method presents the output SNR of 42.17 dB. The proposed method provides a higher SNR than the single-line sensing scheme, owing to its common noise cancellation ability. On the other hand, the proposed method performs better than the delta-integration method, since the proposed recovery algorithm can filter out the remaining noise and maintain the high-resolution sensing data.

$$SNR(\mathrm{dB}) = 20Log\frac{TouchStrength}{NoisTouched_{RMS100}}, \quad (4)$$

$$\mathrm{TouchStrength} = SignalTouched_{AVG100} - SignalUntouched_{AVG10}, \quad (5)$$

$$NoiseTouched_{RMS100} = \sqrt{\frac{\sum_{n=0}^{n=99}(Signal[n] - SighalTouched_{AVG100})^2}{100}}. \quad (6)$$

Table 1. Performance comparison table

Classification	Single-line sensing scheme [13]	Delta (Δ)-integration scheme [14]	Proposed algorithm
Frame scan rate (Hz)	36.42	120	184.5
Scan rate/Touch point (kHz)	125	139.4	633.1
No. of channels	TX: 44 RX: 78	TX: 27 RX: 43	TX: 44 RX: 78
Touch resolution	3,432	1,161	3,432
$\mathrm{SNR}_{\mathrm{out}}$ (dB)	28.86	39	42.17
$\mathrm{SNR}_{\mathrm{gain}}$ (dB) = $\mathrm{SNR}_{\mathrm{out}} - \mathrm{SNR}_{\mathrm{in}}$	38.9	No data provided	52.21
Multi-touch cancellation problem	Does not occur	Not resolved	Resolved

V. CONCLUSION

Although differential sensing methods provide a better SNR than single-line sensing methods, they often suffer from the multi-touch cancellation problem. In this paper, a simple but highly accurate algorithm was introduced, which recovers the multi-touch values from differential sensing values. It can accurately recover not only the multi-touch positions but also the strength of the touches. The proposed method recovers the touched values by using the high-resolution ADC-sampled data. Therefore, it can provide a substantially higher accuracy than a previous method such as [14] without sacrificing the processing speed and the hardware cost. In addition, it is highly effective in detecting touches with very small touched areas, which was not addressed by the previous work [14].

The proposed method consists of only an adder and a comparator, and therefore, it allows a low-cost chip implementation with a negligible area overhead. Simulation experiments have shown that the proposed recovery algorithm provides a high SNR gain of 52.21 dB even before applying any noise-filtering process.

ACKNOWLEDGMENT

This work was supported by the Center for Integrated Smart Sensors funded by the Ministry of Science, ICT, & Future Planning as Global Frontier Project (CISS-2015 M3A6A6066117).

REFERENCES

[1] C. L. Lin, Y. M. Chang, H. S. Chen, C. Y. Chuang, and T. C. Chu, “Position tracking based on particle filter for self-capacitance single-touch screen panels,” *Journal of Display Technology*, vol. 11, no. 2, pp. 165-169, 2015.

[2] P. Y. Li and Z. W. Li, “Study on calibration algorithm of embedded touch screen,” *Journal of Multimedia*, vol. 9, no. 4, pp. 605-610, 2014.

[3] C. L. Lin, C. S. Li, Y. M. Chang, T. C. Lin, J. F. Chen, and U. C. Lin, “Pressure sensitive stylus and algorithm for touchscreen panel,” *Journal of Display Technology*, vol. 9, no. 1, pp. 17-23, 2013.

[4] S. M. Kim, H. Cho, M. Nam, S. G. Choi, and K. Cho, “Low-power touch-sensing circuit with reduced scanning algorithm for touch screen panels on mobile devices,” *Journal of Display Technology*, vol. 11, no. 1, pp. 36-43, 2015.

[5] N. K. Mada and H. Jagadish, “Waking up a capacitive touch-sensing device with an MCU peripheral,” 2011 [Internet], Available: http://www.embedded.com/design/mcus-processors-and-socs/4218309/Waking-up-a-capacitive-touch-sensing-device-with-an-MCU-peripheral.

[6] I. S. Yang, and O. K. Kwon, “A touch controller using differential sensing method for on-cell capacitive touch screen panel systems,” *IEEE Transactions on Consumer Electronics*, vol. 57, no. 3, pp. 1027-1032, 2011.

[7] M. G. M. Mohamed, H. Kim, and T. W. Cho, “A fast sensing method using concurrent driving and sequential sensing for large capacitance touch screens,” *Journal of the Institute of Electronics and Information Engineers*, vol. 52, no. 4, pp. 62-70, 2015.

[8] X. Wu, B. W. Lee, C. Joung, and S. Jang, “Touchware: a software based implementation for high resolution multi-touch applications,” in *Proceedings of IEEE 10th International Conference on Computer and Information Technology*, Bradford, UK, pp. 1703-1710, 2010.

[9] I. Seo, T. W. Cho, H. W. Kim, H. G. Jang, and S. W. Lee, “Frequency domain concurrent sensing technique for large touch screen panels,” in *Proceedings of the IEEK Fall Conference*, Seoul, pp. 55-58, 2013.

[10] D. H. Lim, J. E. Park, and D. K. Jeong, “A low-noise differential front-end and its controller for capacitive touch screen panels,” in *Proceedings of the European Solid-State Circuits Conference (ESSCIRC)*, Bordeaux, France, pp. 237-240, 2012.

[11] C. L. Lin, Y. M. Chang, C. C. Hung, C. D. Tu, and C. Y. Chuang, “Position estimation and smooth tracking with a fuzzy-logic-based adaptive strong tracking kalman filter for capacitive touch panels,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 8, pp. 5097-5108, 2015.

[12] S. H. Bae and Y. S. Lee, “Implementation of a high-performance touch controller and differential sensing circuit,” *International Journal of Computer and Information Technology*, vol. 3, no. 6, pp. 1177-1180, 2014.

[13] I. Seo and H. Kim, “Dual sensing with voltage shifting scheme for high sensitivity touch screen detection,” *Journal of the Institute of Electronics and Information Engineers*, vol. 52, no. 4, pp. 71-79, 2015.

[14] J. H. Yang, S. C. Jung, Y. S. Son, S. T. Ryu, and G. H. Cho, “A noise-immune high-speed readout circuit for in-cell touch screen panels,” *IEEE Transactions on Circuits and Systems I*, vol. 60, no. 7, pp. 1800-1809, 2013.

[15] J. H. Yang, S. H. Park, J. M. Choi, H. S. Kim, C. B. Park, S. T. Ryu, and G. H. Cho, “A highly noise-immune touch controller using Filtered-Delta-Integration and a charge-interpolation technique for 10.1-inch capacitive touch-screen panels,” in *Proceedings of 2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, San Francisco, CA, pp. 390-391, 2013.

[16] J. Kim, M. G. A. Mohamed, and H. Kim, “Design of a frequency division concurrent sine wave generator for an efficient touch screen controller SoC,” in *Proceedings of 2015 IEEE International Symposium on Consumer Electronics (ISCE)*, Madrid, Spain, pp. 1-2, 2015.

[17] D. M. Won and H. Kim, “Touch screen sensing circuit with rotating auto-zeroing offset cancellation,” *Journal of Information and Communication Convergence Engineering*, vol. 13, no. 3, pp. 189-196, 2015.

Ji-Ho Kim
received his B.S. in Information and Communication Engineering from Hanbat National University, Daejeon, South Korea, in 2014. He is currently working towards his M.S. degree in Chungbuk National University, Cheongju, South Korea. He developed a real-time image processing algorithm and implemented it on hardware. His current research topics include mixed signal IC design, digital circuit design, SoC design, and touch screen sensing circuit design.

Dong-Min Won
received his B.S. in Electronics Engineering from Chungbuk National University, Cheongju, South Korea, in 2014. He is currently working towards his M.S. degree from the same university. His current research interest includes mixed signal IC design, analog circuit design, SoC design, and touch screen sensing circuit design.

HyungWon Kim (M’95)
received his B.S. and M.S. in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1991 and 1993, respectively, and his Ph.D. in Electrical Engineering and Computer Science from the University of Michigan, Ann Arbor, MI, USA, in 1999. In 1999, he joined Synopsys Inc., Mountain View, CA, USA, where he developed electronic design automation software. In 2001, he joined Broadcom Corp., San Jose, CA, USA, where he developed various network chips including a WiFi gateway router chip, a network processor for 3G communication, and 10-Gb Ethernet chips. In 2005, he founded Xronet Corp., a Korea-based wireless chip maker, whereas CTO and CEO, he managed the company to successfully develop and commercialize wireless baseband and RF chips and software including WiMAX chips supporting IEEE802.16e and WiFi chips supporting IEEE802.11a/b/g/n. Since 2013, he has been with Chungbuk National University, Cheongju, South Korea, as an assistant professor in the Department of Electronics Engineering. His current research focus covers the areas of wireless sensor networks, wireless vehicular communications, mixed signal SoC designs for low-power sensors, and bio-medical sensors.