Position Estimation and Smooth Tracking With a Fuzzy-Logic-Based Adaptive Strong Tracking Kalman Filter for Capacitive Touch Panels

Chih-Lung Lin, *Member, IEEE*, Yi-Ming Chang, Chia-Che Hung, Chun-Da Tu, and Cheng-Yan Chuang

***Abstract*—This paper presents a novel 7-in capacitive touch panel (CTP) system with a smooth tracking algorithm that accurately estimates the position where the panel is touched and tracks the trajectory of touch. The proposed CTP system consists of a microcontroller unit, a sensor IC, and an interface board. When a user draws at different speeds, the measurement noise caused by the sensor IC induces an error in the touched position and zigzag trajectory, especially when the motion is slow. The fuzzy-logic-based adaptive strong tracking Kalman filter method is implemented in a CTP system to mitigate the effect of measurement noise and provide a smooth tracking trajectory at different speeds. Moreover, the approach effectively measures and quantifies the "smoothness" of the touched trajectory. Experimental results indicate that the proposed method reduces the measurement noise and decreases the mean tracking error by 85.4% over that achieved using the moving average filter.**

***Index Terms*—Capacitive touch panel (CTP), fuzzy-logic system, Kalman filter (KF), strong tracking Kalman filter (STKF).**

I. Introduction

TOUCH panel technology, which is based on resistive, capacitive, acoustic-wave, and infrared methods, has been widely commercialized in mobile phones, digital cameras, navigation systems, TVs, and tablet PCs [1]–[5]. A resistive touch panel consists of two thin, transparent, and conductive layers, such as indium tin oxide films, which are separated by a narrow gap. Despite considerable efforts to develop such panels in order to create a balance between cost and performance [1], [2], the complexity involved in identifying multiple touching events severely limits its range of applications. Therefore, the capacitive touch panel (CTP) has attracted a significant amount of interest and achieved considerable penetration of the consumer electronic product market in recent years owing to its sensitivity, excellent durability, and multitouch functionality [3]. However, CTP is easily affected by noise produced by finger trembling, environmental magnetic interference, or process variation, leading to the inaccurate prediction of touch positions and a zigzag output. Hence, reducing the noise by using hardware or software algorithms is essential to CTP applications [6], [7]. Although Kwon *et al.* [7] proposed a touch controlling circuit based on a differential sensing method to reduce sensing errors and increase the dynamic range of the sensing voltage, the hardware implementation to mitigate the noise is more complex than when using a software algorithm. Therefore, the moving average filter (MAF) method is frequently used to eliminate the measurement noise and locate the touched position [8]–[12]. However, to achieve an acceptable performance, MAF needs a large number of points in a specific interval to filter out a significantly high-frequency noise, leading to amplitude decay and signal delay [13]. To mitigate the noise problem, several researchers have utilized the Kalman filter (KF) to reduce the noise and estimate the state vectors accurately that carry sensor information, including position, velocity, angle, and orientation [14]–[28]. By using KF, Golnaraghi *et al.* estimated the position and velocity based on a hybrid system of a position sensor and an inertial measurement unit [14]. Li *et al.* designed KF to reduce noise and calculate the position and angle for a walking biped robot by using a force sensor and an accelerometer [15]. Based on KF, Miñarro *et al.* estimated the position and velocity in the INS and GPS systems [16]. Despite the effectiveness of KF in suppressing the noise and accurately evaluating the touched positions in CTP [17]–[19], the performance of the KF method depends on precise prior knowledge of the process noise covariance matrix (Q) and the measurement noise covariance matrix (R) [20], [21]. Under touch conditions, the measurement noise varies with the movement speed of the touch: a high speed produces less noise because few sampling points are found in a fixed distance and vice versa. Uncertain speed of touched movement degrades the estimated ability of the KF method with fixed Q and R matrices [22]. Moreover, KF requires exact knowledge of both the dynamic process and measurement models [22], [23]. A modeling error may occur when the theoretical behavior of a filter and its actual behavior do not correlate well with each other [24], [25]. In the CTP system, accurately modeling the

Manuscript received May 5, 2014; revised June 12 2014 and October 13, 2014; accepted December 15, 2014. Date of publication January 30, 2015; date of current version June 26, 2015. This work was supported in part by the Advanced Optoelectronic Technology Center at National Cheng Kung University, the Ministry of Education, the Ministry of Science and Technology (MOST) of Taiwan under Grant 101-2221-E-006-221-MY3, and Cando Corporation.

C.-L. Lin is with the Department of Electrical Engineering and the Advanced Optoelectronic Technology Center, National Cheng Kung University, Tainan 701-01, Taiwan (e-mail: cllin@ee.ncku.edu.tw).

Y.-M. Chang, C.-D. Tu, and C.-Y. Chuang are with the Department of Electrical Engineering, National Cheng Kung University, Tainan 701-01, Taiwan.

C.-C. Hung is with the AU Optronics Coporation, Hsinchu 300-78, Taiwan.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2015.2396874

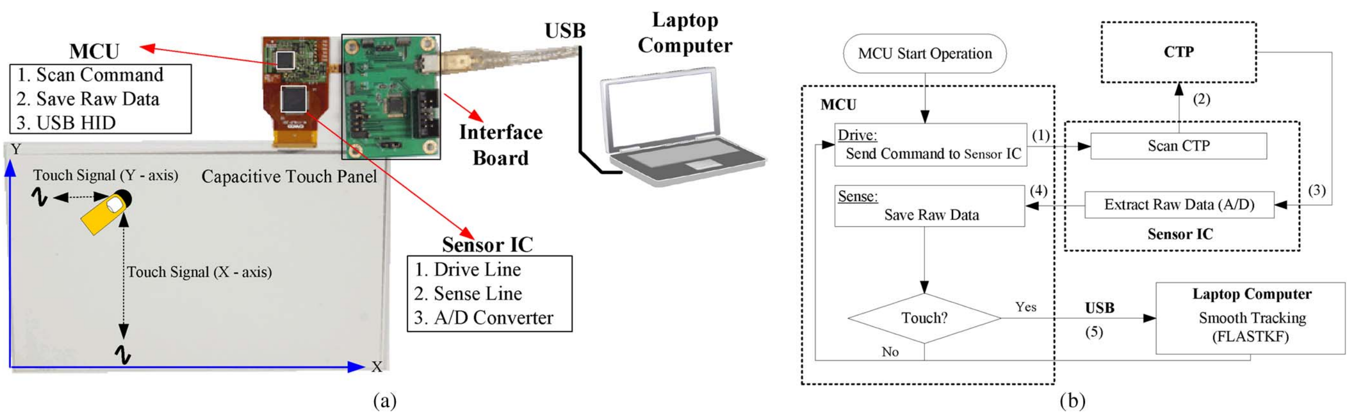


Fig. 1. (a) Photograph of the experimental CTP system. (b) Block diagram of the CTP system.

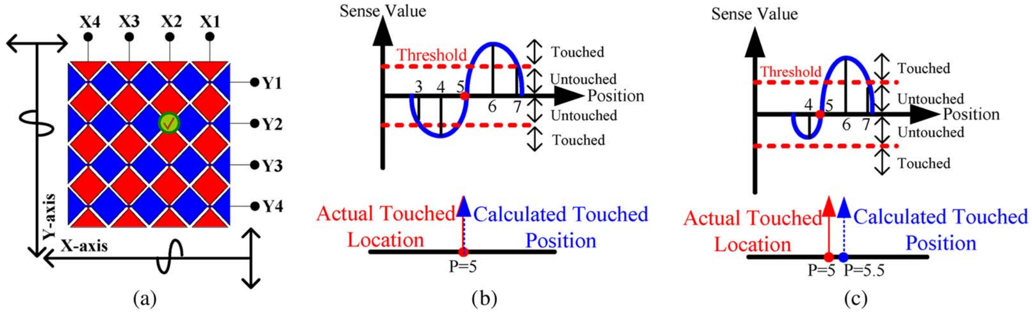


Fig. 2. Actual touched location and calculated touched position using the self-capacitance sensing method. (a) Detection of touched location. (b) Accurate signal generated by the touch of a single finger. (c) Signal generated by touch of a single finger with reduced accuracy owing to noise and the slight change in capacitance of the touched point.

fast and nonlinear movement of a touching finger is extremely difficult. Thus, when the KF models do not fit the behavior of fast and nonlinear movement of a touching finger, the modeling error of KF causes an error in estimating the touched position.

This paper presents a fuzzy-logic-based adaptive strong tracking KF (FLASTKF) algorithm implemented in a CTP system, which is used to solve the aforementioned KF-related problems. The modeling error is eliminated using the strong tracking KF (STKF). In particular, the proposed method integrates fuzzy logic into the STKF to improve the tracking ability of the CTP system. This paper provides a novel method to measure and quantify the “smoothness” of a touched trajectory. Several experiments were conducted to evaluate the effectiveness of the proposed approach. The experimental results indicate that the proposed FLASTKF method successfully achieves the accurate estimation of touched position and also provides a smooth tracking trajectory over the MAF, KF, and STKF methods.

The rest of this paper is organized as follows. Section II describes the hardware components of a CTP system. Section III then presents the FLASTKF algorithm. Next, Section IV displays the several experiments for estimating the touched position. The conclusion is finally drawn in Section V.

II. Overview of Hardware Capacitive Touch System

Fig. 1(a) shows the CTP system consisting of a controller system, a 7-in projected CTP, and a laptop computer. The controller system, consisting of a microcontroller unit (MCU C8051F34D), a sensor IC (CR 31060 with an 8-b A/D converter), and an interface board, is used to detect the variation in the capacitance of CTP and generate raw data to the laptop computer. CTP comprises 30 × 18 ITO diamond-type electrodes, in which the diamond-patterned electrode is 5 mm in length and 1.1 mm in thickness. The sensor IC receives commands from the MCU to scan CTP by using a self- or mutual capacitance sensing method [7], [18]. The sensor collects analog signals, which are transformed into digital data via an internal 8-b A/D converter, before transmitting these data to the MCU. Capable of generating raw data from digital data, the MCU also determines whether CTP has been touched. The laptop computer receives raw data from the USB interface board at a reporting rate of 100 Hz when a touching event occurs. Additionally, the proposed method is implemented in Visual C++ to accurately estimate the touched position on a display panel. Fig. 1(b) shows the block diagram of the hardware system for CTP.

III. Smooth Tracking Algorithm

A. Signal Preprocessing

Signals that are sensed by a sensor IC must be preprocessed because they are always contaminated by noise and finger trembling. In this paper, noise of the sensed signals is reduced by using the weighted average method in order to determine the touched position. Fig. 2(a) shows the location of a touch by the self-capacitance sensing method, in which the self-capacitance of each channel is measured sequentially in a repeated cycle

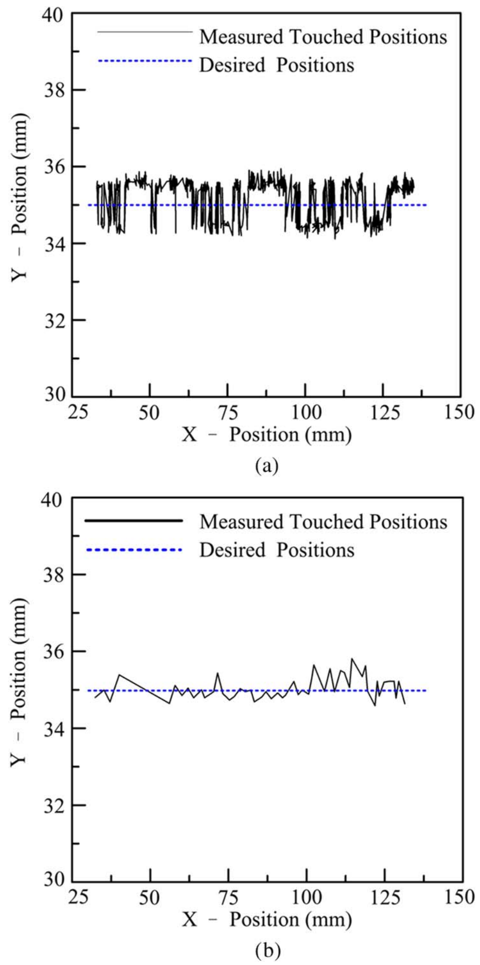


Fig. 3. Trajectory determined from measured touched positions. (a) Touched points during slow movement. (b) Touched points during fast movement.

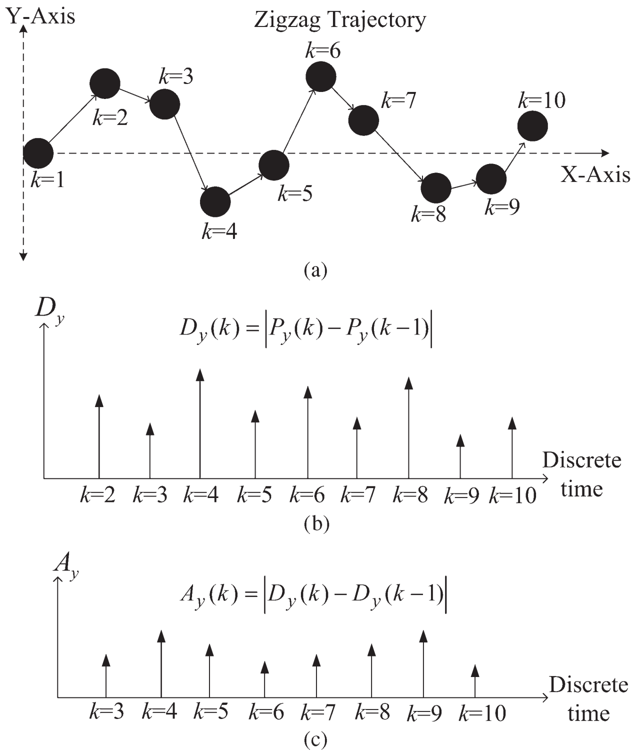


Fig. 4. Example of using the proposed method to characterize a trajectory. (a) Zigzag trajectory. (b) Magnitude of change of position in the y-axis. (c) Magnitude of change of D_y.

[7], [18]. When the touched point is sensed from $X1$ to $X4$ in the X-axis and then $Y1$ to $Y4$ in the Y-axis, a touched location $(X2, Y2)$ of a single finger is recognized by measuring each row and column channel. Hence, two sine waveforms from the X-axis and Y-axis are extracted from the sensor IC to locate the touched point by the weighted average method [17], [18]. Fig. 2(b) plots a complete sine wave of the sensed signal, from which the touched position can be calculated accurately. The actual touched location is the same as the calculated touched position. Fig. 2(c) shows the contaminated sensed signal in the actual touch condition, as affected by the noise and the slight change in capacitance of the touched point. Additionally, the weighted average method yields only a slight error. Nevertheless, apart from accurate detection of the touched position, Fig. 3(a) and (b) describes the measured touched position with a solid line and desired position denoted by a dashed line. In this paper, a user is instructed to draw a straight line along the dashed line at slow and fast speeds, subsequently generating the measured touched position without a filter to ameliorate the noise. Experimental results demonstrate that the sampled signal is significantly more contaminated when the finger movement is slower, explaining the lower accuracy of the calculated touched position. Fig. 3(a) reveals that finger movement at a low speed results in a critical error in the touched position that produces a zigzag trajectory. Fig. 3(b) shows only small errors in the touched position, associated with the movement at a high speed. Additionally, this paper attempts to determine whether the trajectory is smooth or zigzag, by providing quantifiable equations to evaluate the trajectory smoothness as follows. Equation (1) is the magnitude of the change in position, which represents the velocity (speed) along the trajectory without direction in the y-axis, where P_y is the position along the y-axis. Equation (2) is the magnitude of change of D_y

$$D_y(k) = |P_y(k) - P_y(k-1)| \tag{1}$$

$$A_y(k) = |D_y(k) - D_y(k-1)|. \tag{2}$$

If a smooth trajectory is drawn along the x-axis with a nearly constant speed, its magnitudes of change of D_y should approximate to zero. Fig. 4(a) displays the zigzag trajectory along the x-axis, including ten touched positions. Fig. 4(b) and (c) plots the signals associated with (1) and (2). The degree of zigzag trajectory at each touched position can be determined by using (2), explaining why a seriously zigzag trajectory leads to greater $A_y(k)$. Similarly, (3) is the magnitude of change of D_x, where D_x denotes the absolute values of change of position in the x-axis. Equation (4) represents summation of A_x and A_y, and Equation (5) is its average value, where N refers to the last sampling point. Additionally, $S(k)$, which denotes the magnitudes of change of trajectory velocity without direction, is defined as trajectory smoothness in order to represent the amount of noise or zigzag degree. Although some robust discrete differentiators or Euclidean method could be used, it may increase the computational cost over that determined from the proposed $S(k)$, which sufficiently captures the smoothness of the touched trajectory. Therefore, a smoother trajectory is associated with a lower $S(k)$ for each discrete-time index. However, since the noise can be eliminated by using the KF method when the characteristics of noise conform to those of Gaussian white noise, the proposed KF-based method can reduce the noise and estimate the touched position

$$A_x(k) = |D_x(k) - D_x(k-1)|. \tag{3}$$

$$\text{Trajectory smoothness} \equiv S(k) = A_x(k) + A_y(k). \tag{4}$$

$$\text{Average trajectory smoothness} \equiv S_{\text{avg}} = \frac{1}{N-2}\sum_{k=3}^{N} S(k). \tag{5}$$

B. KF

Based on KF, measurement noise is eliminated, and the position of each touched point is estimated. The trajectory model can be expressed using both a process model and a measurement model as follows:

$$\begin{aligned} &\text{Process model}: X(k+1) = AX(k) + w(k)\\ &\text{Measurement model}: Z(k) = CX(k) + v(k) \end{aligned} \tag{6}$$

where

$$X = \begin{bmatrix} x \\ v_x \\ y \\ v_y \end{bmatrix}, \; A = \begin{bmatrix} 1 & \alpha & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \alpha \\ 0 & 0 & 0 & 1 \end{bmatrix},$$
$$Z = \begin{bmatrix} x_m \\ y_m \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

X process state vector of position and velocity;
A transition matrix;
v_x & v_y velocity in x and y coordinates;
α empirical value ($\alpha = 0.8$ used in this paper);
C measurement design matrix;
Z vector of the updated measured position;
w & v zero-mean Gaussian white noise vectors.

The α value in this paper is set to around 0.8, which is determined by experimental measurements and is not a universal value for all applications. Both vector $w(k)$ and vector $v(k)$ are zero-mean Gaussian white sequences with a zero cross correlation with each other [14], [24], [25]

$$E\left[w_k w_i^T\right] = \begin{cases} Q, & i = k \\ 0, & i \neq k \end{cases}$$
$$E\left[v_k v_i^T\right] = \begin{cases} R, & i = k \\ 0, & i \neq k \end{cases}$$
$$E\left[w_k v_i^T\right] = 0, \text{ for all } i \text{ and } k \tag{7}$$

where
Q process noise covariance matrix;
R measurement noise covariance matrix.

As a recursive algorithm designed to estimate the state vector, KF is divided into *time prediction* and *measurement update*. *Time prediction* calculates the forward-time transition of the states in the current epoch $(k-1)$ to those in the next epoch (k). Meanwhile, *measurement update* incorporates the new measurement value in the *a priori* state estimate to yield an improved *a posteriori* state estimate. The equations for *time prediction* and *measurement update* are as follows.

Time prediction:

$$\hat{X}(k|k-1) = A\hat{X}(k-1|k-2) \tag{8}$$
$$P(k|k-1) = AP(k-1|k-1)A^T + Q \tag{9}$$

where
$\hat{X}(k|k-1)$ predicted state vector;
$P(k|k-1)$ prediction covariance matrix.

Measurement update:

$$K(k) = P(k|k-1)C^T\left[R + CP(k|k-1)C^T\right]^{-1} \tag{10}$$
$$\hat{X}(k|k) = \hat{X}(k|k-1) + K(k)\left[Z(k) - C\hat{X}(k|k-1)\right] \tag{11}$$
$$P(k|k) = (I - K(k)C)\,P(k|k-1) \tag{12}$$

where
$K(k)$ Kalman gain matrix;
$\hat{X}(k|k)$ estimated state vector;
$P(k|k)$ estimated covariance matrix.

The Kalman gain matrix is a function of noise statistics. The KF approach update engine is triggered when CTP receives a raw data input, and the measurement update equations can eliminate the measurement noise to estimate the touched position precisely. However, KF depends on exact knowledge of the process and measurement models. When KF is applied with inadequate knowledge of these models, the performance of KF in estimating the touched position is seriously degraded. Even a mismatch may occur between the actual trajectory and the estimated trajectory when the movement is fast and nonlinear, owing to the modeling error [24].

C. STKF

According to a previous study, STKF avoids model uncertainties for the modeling error in the KF method [25]. By applying a suboptimal scaling factor, STKF adjusts the prediction covariance matrix in real time. Hence, in this paper, the STKF method, which provides reliable real-time tracking ability, estimates the touched position with satisfactory precision when movement is fast and nonlinear, subsequently yielding a predicted trajectory that matches the actual trajectory.

The prediction covariance matrix of the conventional KF method can be modified as

$$P(k|k-1) = \lambda_k AP(k-1|k-1)A^T + Q_k \tag{13}$$

where

$$Q_k = \begin{bmatrix} q_k & 0 & 0 & 0 \\ 0 & q_k & 0 & 0 \\ 0 & 0 & q_k & 0 \\ 0 & 0 & 0 & q_k \end{bmatrix}.$$

The value q_k is an element of a diagonal matrix Q_k that is determined using the described fuzzy logic. λ_k is a suboptimal scaling factor applied to the time-varying filter gain value, which is given by

$$\lambda_k = \begin{cases} c_k, & c_k \geq 1 \\ 1, & c_k < 1 \end{cases} \tag{14}$$

where

$$c_k = \frac{\mathrm{tr}[N_k]}{\mathrm{tr}[M_k]} \tag{15}$$

$$N_k = V_k - R - CQ_kC^T \tag{16}$$

$$M_k = CAP(k|k)A^TC^T \tag{17}$$

$$V_k = \begin{cases} v_0v_0^T & k = 0 \\ \frac{\rho V_{k-1} + v_k v_k^T}{1+\rho} & k \geq 1. \end{cases} \tag{18}$$

ρ is the forgetting factor. This paper evaluated the seven values of the forgetting factor, namely, 0.1, 0.3, 0.5, 0.7, 0.8, 0.95, and 1. As demonstrated in Fig. 16 of the experimental results and discussion section, the modeling error can be reduced by adjusting λ_k in a timely manner to obtain almost the same mean tracking errors for the STKF with different forgetting factors, which is also indicated by other studies [29], [30]. A forgetting factor of 0.95 in the STKF is commonly used [25], [31], [32], explained why this value is used here. Innovation sequence parameter v_k is the error between the measured touched position z_k and the predicted touched position $\hat{z}_k$.

The suboptimal scaling factor λ_k in the STKF method increases the accuracy of the process and measurement models, according to the extent determined by the information in the innovation sequence. When a fast and nonlinear movement produces $\lambda_k > 1$ due to the magnitude innovation sequence, STKF provides an appropriate factor λ_k to improve the process and measurement models, subsequently yielding an accurate estimation of touched position. When movement at a normal speed yields $\lambda_k = 1$ owing to accuracy of the process and measurement models, STKF becomes the standard KF, yielding an accurate estimation of the touched position. However, if movement at a low speed yields a significant amount of measurement noise, then the fixed Q and R matrices of the KF and STKF methods fail to capture the measurement noise accurately, subsequently degrading the filter performance and yielding errors when estimating the touched position.

D. FLASTKF

Fuzzy logic is incorporated into the STKF method to adjust q_k adaptively for the change of the measurement noise. The fuzzy rule models the unstable measurement information during each measurement interval. Therefore, this paper develops a novel FLASTKF method to increase the accuracy of determining the trajectory, yielding a smooth tracking trajectory, at all speeds. The FLASTKF method is discussed in detail as follows.

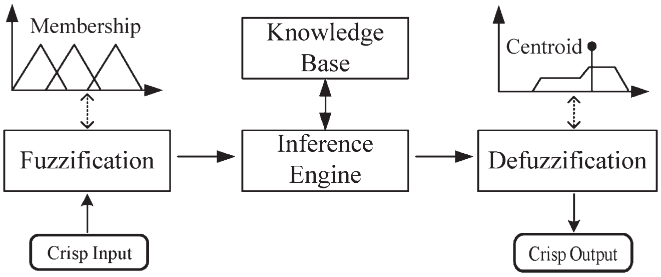


Fig. 5. Fuzzy-logic system.

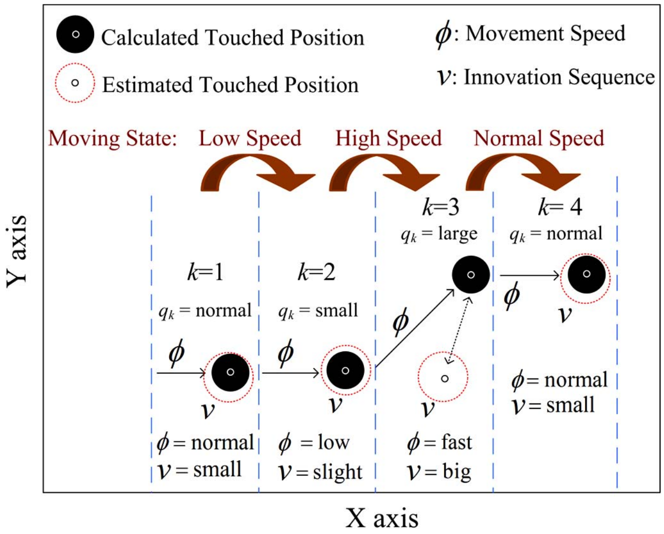


Fig. 6. Definitions of speed of touch movement and innovation sequence.

A typical fuzzy-logic system consists of four processes: fuzzification, inference engine, knowledge base, and defuzzification (Fig. 5). The fuzzification process establishes a mapping process between a crisp input value and a defined fuzzy set. The inference engine calculates the membership value by using the knowledge base that collects the different knowledge about the behavior in the form of a fuzzy "IF-THEN" rule. The defuzzification process then converts fuzzy set into a crisp value by using the established mapping [24], [25], [33].

Fig. 6 displays the calculated and estimated dynamic touched positions, where ϕ is the movement speed of a finger and v denotes the innovation sequence. A fuzzy logic is implemented by monitoring the ϕ and v parameters and adjusting the value of q_k according to these inputs as follows. If the movement speed of a finger is normal (i.e., if ϕ is M) and the innovation sequence is small (i.e., if v is S), then q_k is designed to a normal value (i.e., then q_k is MS) to accurately estimate the touched position. If the movement speed of a finger is low (i.e., if ϕ is S) and the innovation sequence is small (i.e., if v is S), then the value of q_k is set to a smaller value (i.e., then q_k is SS) because a smaller value of q_k can reduce the noise effect. If the movement speed of a finger is fast (i.e., if ϕ is L) and the innovation sequence is large (i.e., if v is L), then the value of q_k is set to a larger value (i.e., then q_k is LL) since a larger

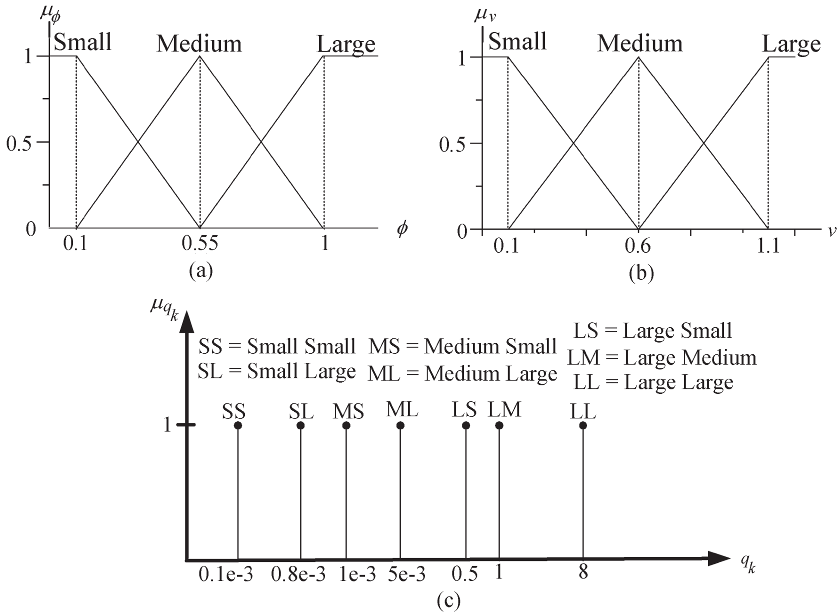


Fig. 7. (a) Membership function of input variable ϕ. (b) Membership function of input variable v. (c) Membership function of output variable q_k.

TABLE I
FUZZY INFERENCE RULE TABLE

v \ q_k \ ϕ	S	M	L
S	SS	MS	LS
M	SL	ML	LM
L	SL	LS	LL

value of q_k supports faster tracking. If the movement speed of a finger is normal (i.e., if ϕ is M) and the innovation sequence is normal (i.e., if v is M), then the value of q_k is designed to a normal value (i.e., then q_k is ML). Correspondingly, the two-input–one-output fuzzy logic can be derived to adjust q_k. Variables ϕ and v (i.e., the input linguistic variables of finger speed and the innovation sequence) characterize the movement; in addition, the variable q_k is used as the output linguistic variables in various touch scenarios.

Human experts are used to generate the fuzzy rulebase herein so no training or testing data set was collected. According to the human experts and experimentally obtained numerical data, Fig. 7(a) and (b) shows the three fuzzy partitions with triangular membership functions of each input variable for linguistic description. To determine the element q_k as a consequent fuzzy set, numerous experiments were conducted to characterize the CTP and STKF. Based on the experimental data and experiences of the human experts, Fig. 7(c) shows the seven fuzzy partitions with singleton-type membership functions of the output variable. Since the singleton fuzzy set in the consequent of a fuzzy rulebase provides a simple computation, its implementation is favored in resource-limited hardware, such as a touch IC or a MCU. Therefore, the processing time in this paper is reduced using the singleton-type membership function. Table I lists the fuzzy inference rules. The weighted average method is used as the fuzzy defuzzification strategy [34].

Integration of fuzzy logic into the STKF: The aforementioned methods are combined to adjust the dynamic q_k based on movement speed and the innovation sequence in order to increase the estimation accuracy of the touched position, regardless of speed. Therefore, this paper develops the FLASTKF method and then applies it to CTP systems. Fig. 8 shows its flowchart. The dashed line indicates a strong tracking loop, in which the suboptimal scaling factor is calculated in four steps. The block on the right-hand side represents fuzzy logic for adjusting q_k. By adjusting both the suboptimal scaling factor and q_k, the FLASTKF method outperforms the KF and STKF methods in both tracking capability and estimation accuracy.

IV. Experimental Results and Discussion

The raw data sampled at 100 Hz, in which an 8-b A/D converter is used, are filtered using the proposed method to determine the touched position accurately. The proposed method is implemented on a laptop computer with a Pentium 4 1.7-GHz processor and 512-MB RAM. Fig. 9 shows a block diagram of data processing in a CTP system. In the proposed method, the movement speed and innovation sequence are determined from raw data by using fuzzy logic. Additionally, the element of the matrix Q_k in the STKF method is calculated. The estimated touch position is then displayed. Fig. 10 shows the experimental system, which is mounted on the display of a laptop computer. The graphical user interface is designed to confirm the accuracy of the analytical results. Experimental results indicate that the proposed method provides smooth trajectory without an observable delay.

This paper also demonstrates the effectiveness of the proposed FLASTKF algorithm by designing and undertaking experiments on trajectory smoothness and tracking. Each experimental outcome is compared with the corresponding outcomes of MAF, KF, and STKF. Notably, the aforementioned filters are optimized to ensure an objective comparison. The number of points in MAF is ten, and the Q/R ratio associated with KF and STKF is set to 0.01.

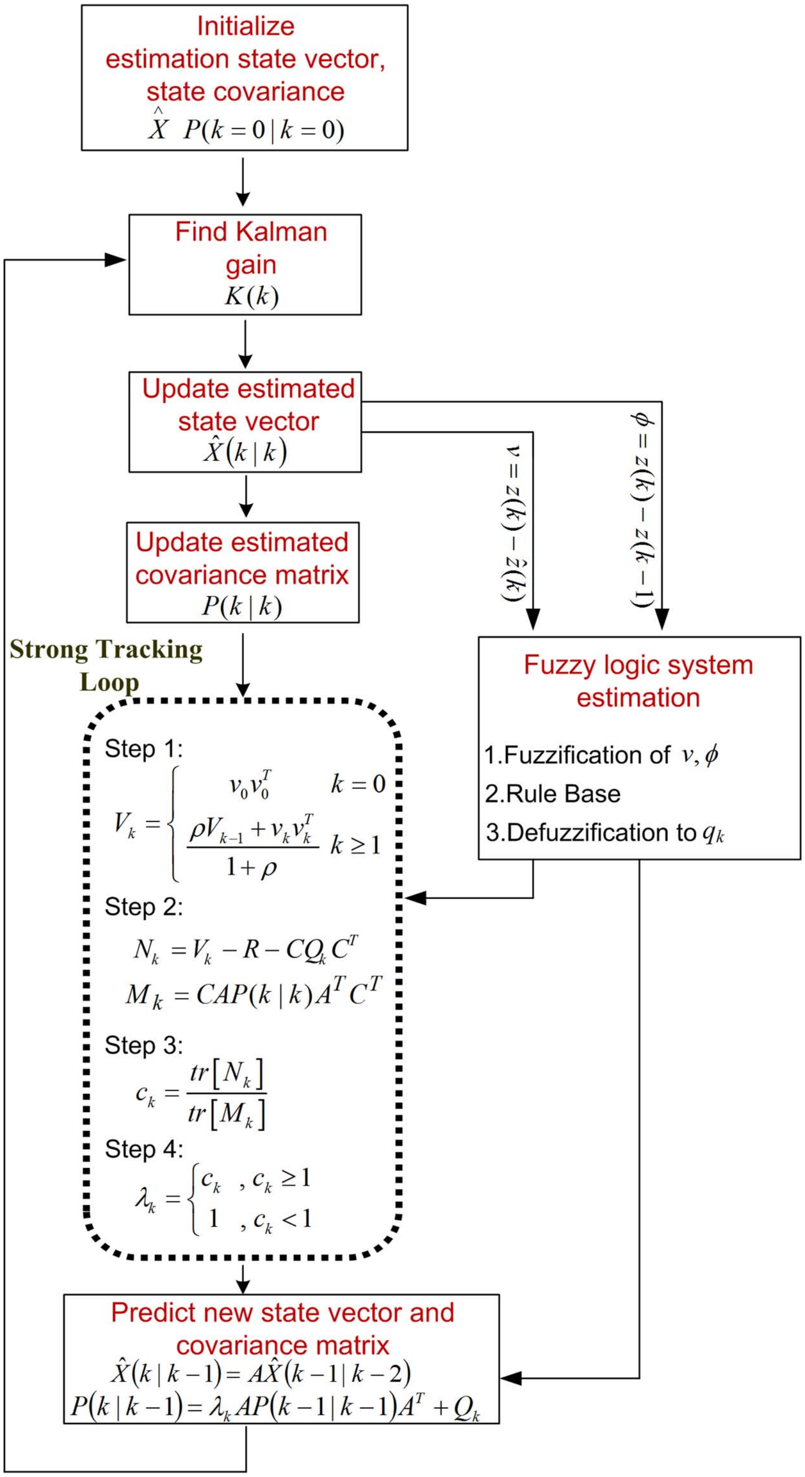

Initialize estimation state vector, state covariance
$\hat{X}$ $P(k=0|k=0)$
Find Kalman gain
$K(k)$
Update estimated state vector
$\hat{X}(k|k)$
Update estimated covariance matrix
$P(k|k)$
$v = z(k) - \hat{z}(k)$
$\phi = z(k) - z(k-1)$
Fuzzy logic system estimation
1.Fuzzification of v, ϕ
2.Rule Base
3.Defuzzification to q_k
Strong Tracking Loop
Step 1:
$V_k = \begin{cases} v_0 v_0^T & k=0 \\ \frac{\rho V_{k-1} + v_k v_k^T}{1+\rho} & k \geq 1 \end{cases}$
Step 2:
$N_k = V_k - R - CQ_k C^T$
$M_k = CAP(k|k)A^T C^T$
Step 3:
$c_k = \frac{tr[N_k]}{tr[M_k]}$
Step 4:
$\lambda_k = \begin{cases} c_k & , c_k \geq 1 \\ 1 & , c_k < 1 \end{cases}$
Predict new state vector and covariance matrix
$\hat{X}(k|k-1) = A\hat{X}(k-1|k-2)$
$P(k|k-1) = \lambda_k AP(k-1|k-1)A^T + Q_k$

Fig. 8. Flowchart of FLASTKF.

Fig. 11 compares the smooth trajectories for circular motion at a low speed (1.53 cm/s) determined by using the raw data, MAF, KF, STKF, and FLASTKF. The $S(k)$ of the trajectory as previously mentioned in (4) is used to quantify the trajectory smoothness. The smoothness values obtained using MAF, KF, STKF, and the proposed method show improved results over the raw data without the filter. The MAF only eliminates the high-frequency noise [13], yielding a less smooth trajectory than those of KF, STKF, and the proposed method. This figure also reveals that using the KF-based methods can reduce the noise effect when the uncertain noise of raw data includes Gaussian white noise. The STKF and KF methods obtain the same smoothness of a trajectory because STKF reduces to the standard KF when the finger motion is slow, such that $\lambda_k = 1$. Estimating the touched position using the proposed method yields a favorable trajectory smoothness when q_k is adjusted to eliminate the variation of measurement noise. Thus, according to Fig. 11, a smoother trajectory using the proposed

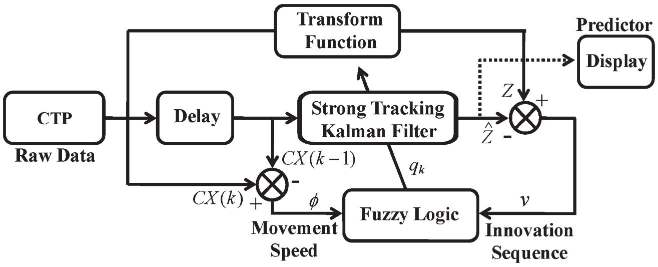


Fig. 9. FLASTKF process in a CTP system.

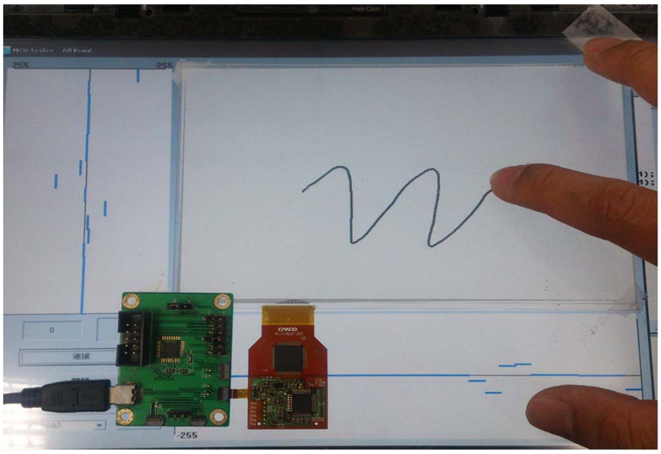


Fig. 10. Experiments of smooth trajectory using the FLASTKF method in a CTP system.

method can yield a lower S_{avg} of 0.022 than the MAF, KF, or STKF methods, which yield values of 0.112, 0.066, and 0.066, respectively. Fig. 12 shows the results of writing the nine digits from "1" to "9," as well as the calculated S_{avg} for writing each digit. According to this figure, the S_{avg} in the writing of each digit obtained using the proposed method is lower than those obtained from the MAF, KF, and STKF methods since the mean of S_{avg} in the proposed method is about 0.0402 lower than those obtained using the STKF, KF, and MAF methods, which are 0.0704, 0.0704, and 0.1151, respectively. Therefore, according to Figs. 11 and 12, the proposed method eliminates noise effectively and provides favorable trajectory smoothness for the CTP.

This paper also demonstrates the tracking capability of the proposed method by analyzing fast linear and nonlinear trajectories. In the analysis of linear trajectory, a straight line is drawn fast to compare the trajectory delays associated with MAF, KF, STKF, and the proposed method. Trajectory delay refers to the distance between the end of the line and the N point of the filters. Fig. 13(a) compares the trajectory delays for a straight line (12.7 cm) at a high speed (25.4 cm/s) obtained using raw data and the MAF, KF, STKF, and FLASTKF methods. The trajectory obtained using the MAF method incurs a significant delay owing to the use of insufficient sampling points; in addition, the KF method yields a similar trajectory delay owing to the modeling error. The trajectory delay obtained using the STKF method can be reduced by tuning λ_k to reduce the modeling error. The proposed method has a shorter delay distance than that of MAF, KF, and STKF because λ_k and q_k are simultaneously adjusted. Fig. 13(b) plots the delay distances at various speeds, as obtained using various algorithms. The tested speeds are 8.47, 12.7, 16.9, and 25.4 (cm/s). An increasing finger speed implies a longer trajectory delay in the MAF and KF methods. In the proposed method, the trajectory delay is considerably lower than 4 (mm).

Moreover, a nonlinear trajectory is analyzed by drawing a sine waveform and comparing the trajectories tracked by MAF, KF, STKF, and the proposed method. Excellent trajectory tracking can rapidly track the nonlinear touched trajectory. Since the raw data contain little noise, which is negligible during high-speed movement, the raw data are replaced with the actual touched trajectory as the reference trajectory in order to evaluate the tracking performance. The equation for tracking effectiveness is

$$e_t(k) = \sqrt{e_x(k)^2 + e_y(k)^2} \tag{19}$$

$$e_x(k) = P_{\text{ref\_}x}(k) - P_{\text{fil\_}x}(k) \tag{20}$$

$$e_y(k) = P_{\text{ref\_}y}(k) - P_{\text{fil\_}y}(k) \tag{21}$$

where $e_t(k)$ is the tracking error between the reference point and the filtered point, $e_x(k)$ and $e_y(k)$ are the errors in the x and y-directions, and $(P_{\text{ref\_}x}(k), P_{\text{ref\_}y}(k))$ and $(P_{\text{fil\_}x}(k), P_{\text{fil\_}y}(k))$ refer to the positions of the reference point and the filtered point, respectively.

Fig. 14 compares the applications of the aforementioned methods with a nonlinear trajectory and calculated tracking errors at a speed of 45 (cm/s). Based on the measurements, the trajectories that are obtained using the MAF and KF methods do not match the reference trajectory, resulting in a serious tracking error. The trajectory that is obtained using the STKF method does not track the reference trajectory very closely because only λ_k is adjusted. The proposed FLASTKF method almost matches the reference trajectory. Additionally, the method yields a smaller tracking error than those obtained using the MAF, KF, and STKF methods. Moreover, Fig. 15 compares the mean tracking errors with nonlinear motions at different speeds (i.e., 30, 35, 40, and 45 cm/s). The mean tracking errors obtained using the MAF and KF methods gradually increase with the speeds of the touched movement. Moreover, the average of the mean tracking errors obtained using the proposed FLASTKF is improved by 85.4% over that of the MAF method. Therefore, experimental results indicate that adjusting the q_k in the proposed method both eliminates the unstable measurement noise and enhances the tracking performance to achieve smooth tracking, regardless of the speed.

To discuss how different values of forgetting factor in the STKF method impact the estimated accuracy of the touched position, this paper evaluated the seven values of the forgetting factor, namely, 0.1, 0.3, 0.5, 0.7, 0.8, 0.95, and 1, which is analyzed by drawing the nonlinear trajectory at different speeds of 30 and 45 cm/s. Fig. 16 compares the mean of the tracking errors of the KF with that of STKF using the seven values of the forgetting factor. The mean of the tracking errors obtained using the STKF method is significantly better than that obtained using the KF method. The mean values of the tracking errors at a speed of 30 (cm/s) in the best case ($\rho = 0.95$) and the

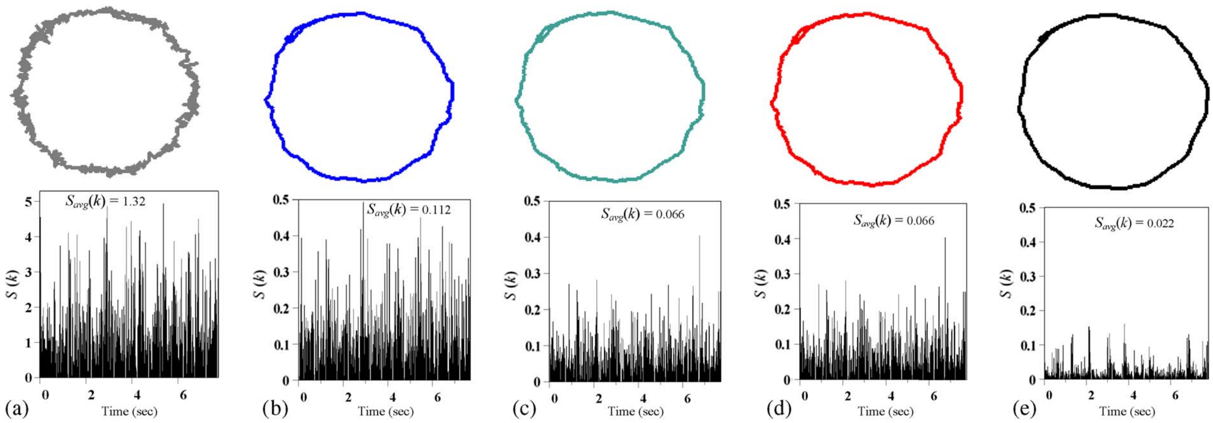
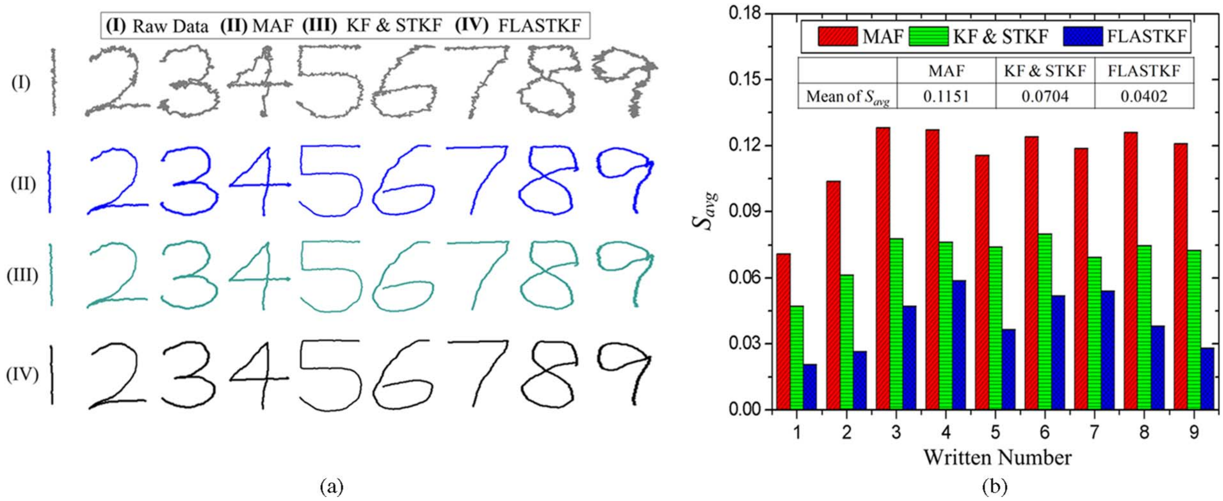


Fig. 11. Comparison of smooth trajectories and $S(k)$ associated with circular motion determined by various methods. (a) Raw data. (b) MAF method. (c) KF method. (d) STKF method. (e) FLASTKF method.



	MAF	KF & STKF	FLASTKF
Mean of S_{avg}	0.1151	0.0704	0.0402

Fig. 12. Comparison of smooth trajectories while writing nine digits slowly. (a) Results of writing digits from one to nine on CTP of LCD. (b) $S_{\rm avg}$ of trajectory in writing nine digits.

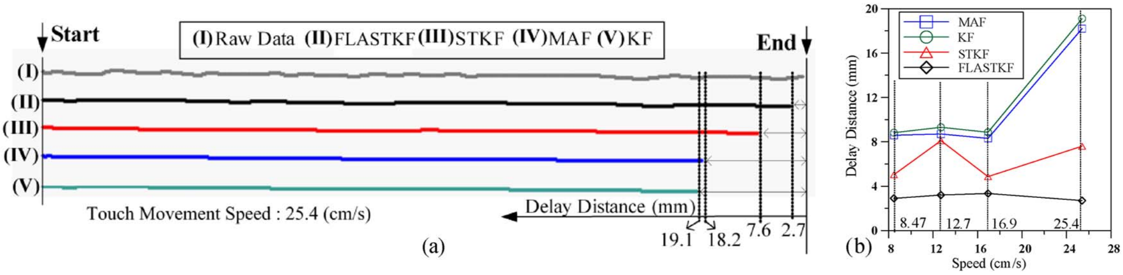


Fig. 13. Comparison of trajectory delays associated with fast horizontal motion. (a) Drawing horizontal line on CTP of LCD. (b) Distance delay of trajectory of motion at four speeds.

worst case ($\rho = 0.8$ or 0.7) are 60.1% and 58.7%, respectively, better than those obtained using the KF method. At a speed of 45 (cm/s), the mean values of the tracking errors in the best case ($\rho = 0.1$) and the worst case ($\rho = 1$) are 65% and 62.6%, respectively, better than those obtained using the KF method. Therefore, the measurements reveal that the degree of improvement in the tracking error obtained using the STKF is almost independent of the forgetting factor in the CTP system. A forgetting factor of 0.95 in the STKF is commonly used [25], [31], [32], so this value is utilized herein.

Although the experimental results indicate that the proposed FLASTKF method outperforms the MAF, KF, and STKF methods, the computational cost is also increased. Owing to the idea that it contains the strong tracking and fuzzy logic, which also has the complex computation, the proposed method requires more computational costs than those of the MAF, KF, and STKF methods. However, the processing time can be reduced by using a faster processor or dedicated solver such as a Java development platform. Additionally, the report rate of the proposed algorithm conforms to that of the Microsoft certificate.

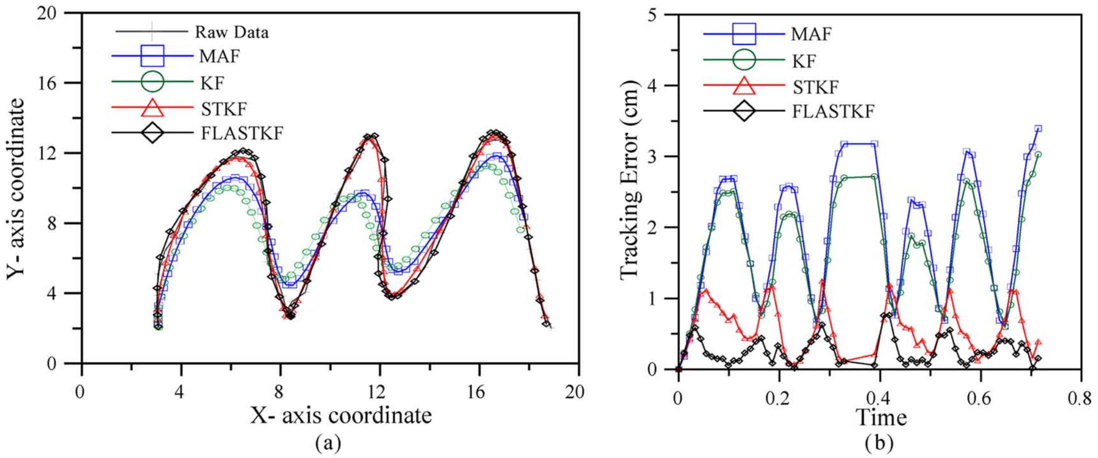


Fig. 14. (a) Nonlinear movement at 45 (cm/s) on CTP. (b) Comparison of errors of tracking trajectory in nonlinear movement at 45 (cm/s).

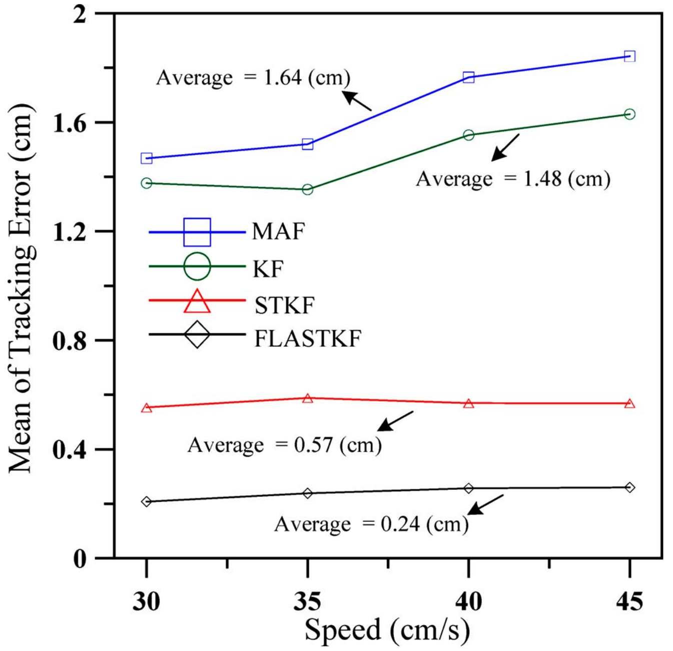


Fig. 15. Comparison of mean tracking errors with nonlinear movement at four speeds.

V. Conclusion

This paper has presented a novel CTP system that consists of an MCU, a sensor IC, and an interface board. Based on a self-capacitance sensing method, the proposed system detects a touching event and senses the associated variation in capacitance. The sensed signal is then calculated using the weighted average method to determine the approximate touched position. The aforementioned procedures for identifying the touched position are implemented in the MCU to control the sensor IC. Additionally, the measurement noise incurred by the sensor IC is solved using the KF method in order to estimate the touched position accurately. Moreover, to determine the touched trajectory accurately, the proposed FLASTKF method is adopted to solve the variation of measurement noise associated with touched movements with speed, as well as offers smoother tracking of the trajectory. Experimental results demonstrate that the proposed method can reduce the impact of unstable measurement noise, subsequently increasing the trajectory smoothness. Furthermore, the mean tracking error using the proposed method is improved by 85.4% over that of the MAF method, explaining why the proposed method performs precise trajectory tracking, regardless of the speed. Therefore, the feasibility of implementing the proposed method, as applied to a CTP system, is verified experimentally.

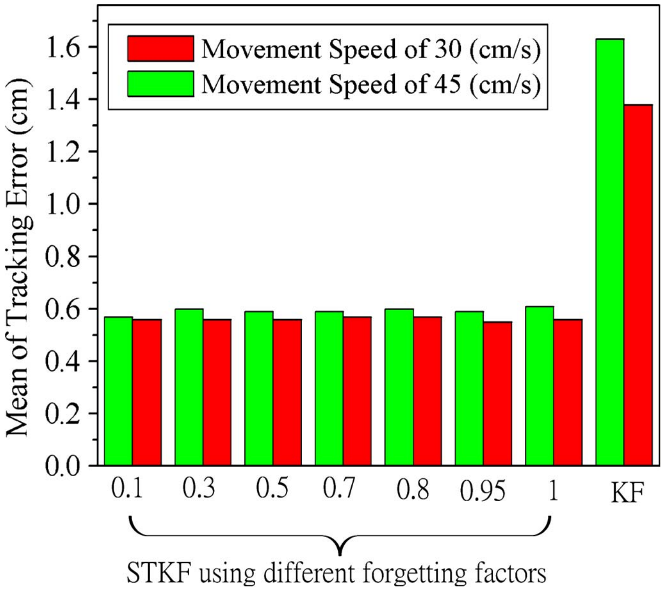


Fig. 16. Comparison of mean tracking errors between KF and STKF using different forgetting factors.

References

[1] R. N. Aguilar and G. C. M. Meijer, "Fast interface electronics for a resistive touch screen," in *Proc. IEEE Sens.*, 2002, vol. 2, pp. 1360–1363.

[2] Y. Park, J. Bae, E. Kim, and T. Park, "Maximizing responsiveness of touch sensing via charge multiplexing in touchscreen devices," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1905–1910, Aug. 2010.

[3] S. Kim *et al.*, "A highly sensitive capacitive touch sensor integrated on a thin-film-encapsulated active-matrix OLED for ultrathin displays," *IEEE Trans. Electron Devices*, vol. 58, no. 10, pp. 3609–3615, Oct. 2011.

[4] R. Adler and P. J. Desmares, "An economical touch panel using SAW absorption," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 34, no. 2, pp. 195–201, Mar. 1987.

[5] S. H. Bae *et al.*, "Integrating multi-touch function with a large-sized LCD," in *SID Tech. Dig.*, 2008, pp. 178–181.

[6] T. H. Hwang, W. H. Cui, I. S. Yang, and O. K. Kwon, “A highly area-efficient controller for capacitive touch screen panel systems,” *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 1115–1122, May 2010.

[7] I. S. Yang and O. K. Kwon, “A touch controller using differential sensing method for on-cell capacitive touch screen panel systems,” *IEEE Trans. Consum. Electron.*, vol. 57, no. 3, pp. 1027–1032, Aug. 2011.

[8] H. R. Kim *et al.*, “A mobile-display-driver IC embedding a capacitive-touch-screen controller system,” in *ISSCC Dig. Tech. Papers*, Feb. 2010, pp. 114–116.

[9] C. Wen and C. H. Huang, “A paperless fax machine with a single-touch panel,” *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1488–1491, Nov. 2008.

[10] S. Ko *et al.*, “Low noise capacitive sensor for multi-touch mobile handset’s applications,” in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2010, pp. 1–4.

[11] R. Wimmer and P. Baudisch, “Modular and deformable touch-sensitive surfaces based on time domain reflectometry,” in *Proc. User Interface Softw. Technol.*, Oct. 2011, pp. 517–526.

[12] Y. Nakai and N. Matsuo, “Portable Device, Method of Detecting Operation, Computer-Readable Storage Medium Storing Program for Detecting Operation,” U.S. Patent 13/126,075, Aug. 25, 2011.

[13] J. S. Wang, Y. L. Hsu, and J. N. Liu, “An inertial-measurement-unit-based pen with a trajectory reconstruction algorithm and its applications,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3508–3521, Oct. 2010.

[14] S. P. Won, W. W. Melek, Senior, and F. Golnaraghi, “A Kalman/particle filter-based position and orientation estimation method using a position sensor/inertial measurement unit hybrid system,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 5, pp. 1787–1798, May 2010.

[15] T.-H. S. Li, Y. T. Su, S. H. Liu, J. J. Hu, and C. C. Chen, “Dynamic balance control for biped robot walking using sensor fusion, Kalman filter, fuzzy logic,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4394–4408, Nov. 2012.

[16] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Úbeda-Miñarro, and A. F. Gómez-Skarmeta, “High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 491–511, Sep. 2007.

[17] C. L. Lin, Y. M. Chang, U. C. Lin, and C. S. Li, “Kalman filter smooth tracking based on multi-touch for capacitive panel,” in *SID Tech. Dig.*, 2011, pp. 1845–1847.

[18] C. L. Lin *et al.*, “Pressure sensitive stylus and algorithm for touchscreen panel,” *IEEE/OSA J. Display Technol.*, vol. 9, no. 1, pp. 17–23, Jan. 2013.

[19] C. L. Lin, C. S. Li, Y. M. Chang, U. C. Lin, and C. C. Hung, “3D stylus and pressure sensing system for capacitive touch panel,” in *Proc. IEEE Int. Conf. Consum. Electron.*, 2012, pp. 215–216.

[20] R. K. Mehra, “On the identification of variances and adaptive Kalman filtering,” *IEEE Trans. Autom. Control*, vol. AC-15, no. 2, pp. 175–184, Apr. 1970.

[21] K. Nam, S. Oh, H. Fujimoto, and Y. Hori, “Estimation of sideslip and roll angles of electric vehicles using lateral tire force sensors through RLS and Kalman filter approaches,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 3, pp. 988–1000, Mar. 2013.

[22] F. Jiancheng and Y. Sheng, “Study on innovation adaptive EKF for in-flight alignment of airborne POS,” *IEEE Trans. Instrum. Meas.*, vol. 60, no. 4, pp. 1378–1388, Apr. 2011.

[23] A. Chatterjee and F. Matsuno, “A neuro-fuzzy assisted extended Kalman filter-based approach for simultaneous localization and mapping (SLAM) problems,” *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 5, pp. 984–997, Oct. 2007.

[24] W. Abdel-Hamid, A. Noureldin, and N. El-Sheimy, “Adaptive fuzzy prediction of low-cost inertial-based positioning errors,” *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 3, pp. 519–529, Jun. 2007.

[25] D. J. Jwo and S. H. Wang, “Adaptive fuzzy strong tracking extended Kalman filtering for GPS navigation,” *IEEE Sens. J.*, vol. 7, no. 5, pp. 778–789, May 2007.

[26] C. M. Wen and M. Y. Cheng, “Development of a recurrent fuzzy CMAC with adjustable input space quantization and self-tuning learning rate for control of a dual-axis piezoelectric actuated micromotion stage,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 11, pp. 5105–5115, Nov. 2013.

[27] E. DiGiampaolo and F. Martinelli, “Mobile robot localization using the phase of passive UHF RFID signals,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 365–376, Jan. 2014.

[28] T. T. Phuong, K. Ohishi, Y. Yokokura, and C. Mitsantisuk, “FPGA-based high-performance force control system with friction-free and noise-free force observation,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 2, pp. 994–1008, Feb. 2014.

[29] D. H. Zhou, Y. X. Sun, Y. G. Xi, and Z. J. Zhang, “Extension of Friedland’s separate-bias estimation to randomly time-varying bias of nonlinear systems,” *IEEE Trans. Autom. Control*, vol. 38, no. 5, pp. 1270–1273, Aug. 1993.

[30] Z. T. Zhang and J. S. Zhang, “A strong tracking nonlinear robust filter for eye tracking,” *J. Control Theory Appl.*, vol. 8, no. 4, pp. 503–508, Apr. 2010.

[31] X. He, Z. Wang, X. Wang, and D. H. Zhou, “Networked strong tracking filtering with multiple packet dropouts: Algorithms and applications,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 3, pp. 1454–1463, Mar. 2014.

[32] T. Xu, Q. Ge, X. Feng, and C. Wen, “Strong tracking filter with bandwidth constraint for sensor networks,” in *Proc. IEEE Int. Conf. Control Autom.*, Jun. 2010, pp. 596–601.

[33] S. E. Beid and S. Doubabi, “DSP-based implementation of fuzzy output tracking control for a boost converter,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 196–209, Jan. 2014.

[34] H. H. Choi and J. W. Jung, “Discrete-time fuzzy speed regulator design for PM synchronous motor,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 600–607, Feb. 2013.

Chih-Lung Lin (M’05) received the M.S. and Ph.D. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1993 and 1999, respectively.

He is currently a Professor with the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan. His current research interests include pixel circuit design for AMOLEDs, gate driver circuit design for AMLCDs, and flexible display circuits.

Yi-Ming Chang received the B.S. and M.S. degrees in electrical engineering from Feng Chia University, Taichung, Taiwan, in 2007 and 2009, respectively. He is currently working toward the Ph.D. degree in electrical engineering at National Cheng Kung University, Tainan, Taiwan.

His research focuses on the touch interface and tracking algorithms for capacitive touch panels.

Chia-Che Hung received the M.S. and Ph.D. degrees in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 2010 and 2013, respectively.

He is currently with the AU Optronics Corporation, Hsinchu, Taiwan, working on the design of blue phase liquid crystal displays. His research focuses on pixel circuits and display system designs for AMOLEDs.

Chun-Da Tu received the B.S. and M.S. degrees in electronic engineering from Kun Shan University, Tainan, Taiwan, in 2004 and 2006, respectively, and the Ph.D. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 2011.

He is currently a Postdoctoral Researcher in electrical engineering with National Cheng Kung University. His research focuses on scan- and source-driver circuit design on glass for AMLCDs.

Cheng-Yan Chuang received the M.S. degree in mathematical and electrical engineering from National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan, in 2010. He is currently working toward the Ph.D. degree in electrical engineering at National Cheng Kung University, Tainan, Taiwan.

His research focuses on the touch interface and tracking algorithms for capacitive touch panels.