scientific reports

Check for updates

OPEN

A smart pen prototype with adaptive algorithms for stabilizing handwriting tremor signals in Parkinson's disease

Jéssica Cristina Tironi[1,2], Anita Fernandes[1], Renata Coelho Borges[2], Luis Augusto Silva[3]✉ & Wemerson Delcio Parreira[4]✉

This study presents a smart pen prototype designed to dynamically mitigate hand tremors, thereby enhancing writing quality and user comfort for individuals with conditions such as Parkinson's disease. The device employs an accelerometer for real-time tremor detection, a microcontroller for rapid data processing, and a vibration motor to counteract tremor effects. Adaptive algorithms—including Fx-LMS, Fx-NLMS, a combined Fx-LMS/NLMS approach, RLS, and the Kalman Filter—were evaluated using signals from the NewHandPD dataset. Simulation results revealed that although the RLS algorithm achieved the lowest mean square error, the Kalman Filter converged approximately eight times faster, a finding that was confirmed through microcontroller tests and further validated on an orbital shaking table under constant and variable tremor conditions. These outcomes underscore the potential of the Kalman Filter as a non-invasive, adaptive solution for real-time tremor mitigation in assistive writing devices. Future improvements may include integrating additional sensors and further optimizing microcontroller performance to enhance overall adaptability and accuracy.

Keywords Kalman Filters, Parkinson's disease, Pen prototype, Tremor mitigation, Adaptive control

Abbreviations

DBS	Deep Brain Stimulation
ET	Essential Tremor
Fx-LMS	Filtered-x Least Mean Square
Fx-NLMS	Filtered-x Normalized Least Mean Square
HDBS	Higher-Dimensional Biodynamics System
IMU	Inertial Measurement Unit
LDBM	Lower-Dimensional Biodynamic Model
LMS	Least Mean Square
MSE	Mean Square Error
PD	Parkinson's Disease
PID	Proportional-Integral-Derivative
PVC	Polyvinyl Chloride
RLS	Recursive Least Square

The brain, the most complex organ in the human body, is commonly characterized as the epicenter of the organism's control, influencing practically all facets of life. Neurological disorders, such as Parkinson's Disease and Essential Tremor (ET), affect millions of people globally, causing them to have significant challenges performing daily tasks[1,2]. These tremors, characterized by rhythmic and involuntary oscillations, vary in frequency and intensity depending on their origin[3]. ET, frequently observed in elderly people, manifests itself as a bilateral postural tremor, mainly affecting the upper limbs. Parkinsonian tremor (PD) occurs predominantly

[1]Polytechnic School, University of Vale do Itajaí (UNIVALI), Itajaí 88302-901, SC, Brazil. [2]Present address: Graduate Program in Biomedical Engineering (PPGEB), Federal University of Technology – Paraná (UTFPR), Curitiba 80230-901, PR, Brazil. [3]Department of Computer Science, Faculty of Science, Universidad de Salamanca, Salamanca 37008, Spain. [4]Faculty of Electrical Engineering, Polytechnic School, Pontifical Catholic University of Campinas (PUC-Campinas), Campinas 13087-571, SP, Brazil. ✉email: luisaugustos@usal.es; wemerson@puc-campinas.edu.br

at rest, with a frequency of 3 to 7 Hz, and is episodic and can be intensified or suppressed by motor or cognitive activities. It is worth noting that even healthy people can experience physiological tremors, characterized by bilateral muscle oscillations with a frequency of 8 to 12 Hz, which can be exacerbated by fatigue, stress, anxiety, and certain medications[4,5].

Treatment for these tremors varies depending on the underlying cause but usually includes a combination of medications, physical therapy, and, in more severe cases, surgical interventions, such as Deep Brain Stimulation (DBS)[6]. While these approaches can be effective, they also have significant limitations. Medications can lose effectiveness over time and cause unwanted side effects, while surgeries are invasive and do not always guarantee permanent results. Furthermore, many of these options do not address the need for more personalized and less invasive solutions that allow patients to maintain a degree of autonomy in their daily lives[7,8].

In this scenario, the development of assistive technologies appears as a promising alternative. Devices utilizing sensors, actuators, and adaptive algorithms have been widely researched to suppress tremors and help individuals regain control over their daily activities[9,10]. In the last few years, some research in assistive technologies has explored several approaches to controlling tremors in patients with diseases such as Parkinson's. Several studies have proposed mechanical or wearable devices for tremor suppression. Van and Ngo (2021)[11] developed a wearable device based on gyroscopes and actuators, which was able to attenuate tremors with an effectiveness of up to 92.6%. Furthermore, Chen Liu (2024)[12] proposed a wearable orthosis for suppressing finger tremors, utilizing a stacked polyvinyl chloride (PVC) gel linear actuator, force sensors, and an inertial measurement unit (IMU). The device employs the Least Mean Square (LMS) algorithm for vibration control, achieving a tremor suppression rate exceeding 65% at the fingertips.

Other approaches explored control methods such as PID controllers and fuzzy logic systems. Studies such as those by Zamanian and Richer (2019)[13] and Jamaludin et al. (2021)[14] demonstrated significant reductions in tremors using permanent magnet motors, notch filters, fuzzy controllers, and actuators, achieving suppression of up to 97% and 88.39%, respectively. Chandra et al. (2024)[10] addressed tremor suppression by developing a mathematical model of the human hand that aids in selecting suitable actuators and controllers. They focused on the design of a Proportional-Integral-Derivative (PID) controller derived from a Lower-Dimensional Biodynamic Model (LDBM) that is applied to a Higher-Dimensional Biodynamics System (HDBS). The approach emphasized model simplification and error minimization to achieve effective tremor suppression, without indicating that it adapts in real-time to changing conditions or parameters. This means the control strategy presented is not characterized as adaptive. In contrast, Araujo et al. (2023)[15] investigated LMS-based algorithms using simulated tremor signals. Although the results were promising, the simulations relied on synthetic signals, and the algorithms were not tested on embedded platforms or validated with real-world tremor data, which limits their practical applicability.

Our work focuses on implementing an adaptive approach to tremor suppression, leveraging adaptive algorithms such as Fx-LMS, Fx-NLMS, Fx-NLMS&LMS, RLS, and Kalman, selected based on their effectiveness and demonstrated performance in practical vibration control scenarios within the literature[15–17]. In software simulation, we evaluated these algorithms using real input signals collected from patients with Parkinson's disease, providing a realistic contrast to previous studies relying on synthetic signals[15]. Additionally, inspired by prior research, the development of the pen prototype required a careful selection of electronic components. The chosen hardware, comprising a microcontroller, DC actuators, and 3-axis accelerometers, enables precise signal acquisition and processing. The microcontroller interprets signals from the accelerometers to generate an adaptive output that suppresses tremors, enhancing writing accuracy and legibility.

While these aspects highlight the system's practical design, it is worth noting that most prior works rely on synthetic tremor data or fixed control strategies that do not adapt to individual variability. Therefore, this work addresses a critical gap by combining real patient-derived signals, adaptive filtering, and real-time embedded implementation for tremor mitigation — a configuration not yet explored in the literature. The technical challenges of developing such devices are significant, particularly regarding algorithm selection and component integration to ensure effective and efficient performance. So, this work aims to advance in the assistive technologies area by providing an innovative and less invasive solution for patients experiencing tremors, significantly improving their quality of life.

The remainder of this paper is organized as follows. In Section 2.1, we present a comprehensive review of adaptive filtering, focusing on the algorithms relevant to our analysis. In Section 3, we describe the proposed hardware developed specifically for tremor cancellation, covering its design and key operational aspects. Section 4 provides the results and offers a detailed discussion of our findings concerning the effectiveness of tremor mitigation. Finally, we conclude the paper in Section 5, summarizing our study's contributions.

The primary contributions of this study are as follows:

i) implementation and comparison of five adaptive algorithms (Fx-LMS, Fx-NLMS, LMS&Fx-NLMS, RLS, Kalman) using real tremor signals from PD patients;
ii) deployment of these algorithms in a microcontroller-based smart pen prototype;
iii) experimental validation of the prototype using an orbital shaking table under controlled and variable conditions; and
iv) demonstration of the Kalman filter's superior convergence speed in embedded systems, supporting its viability for assistive applications.

Materials and methods

According to[18], adaptive algorithms are widely employed in applications involving signals with unknown or time-varying statistical properties, unlike fixed coefficient algorithms, which have predefined internal parameters and

do not adapt to their applied environment; adaptive algorithms can automatically adjust their internal parameters (coefficients or learning rate) based on the characteristics of the input signal, thus optimizing their performance.

The adaptive algorithms chosen for the development of this work included Fx-LMS, Fx-NLMS, Fx-NLMS&LMS, RLS, and the Kalman filter. We have selected Fx-LMS, Fx-NLMS, and Fx-NLMS&LMS algorithms based on work[15], where the efficiency of these adaptive algorithms in tremor control was demonstrated and confirmed through simulations using a mathematical model to generate hand tremors.

The simulation process utilizes the NewHandPD dataset[19,20]. The NewHandPD is a dataset of 736 JPEG images of spirals and meanders drawn by 92 individuals (18 healthy controls, 74 Parkinson's patients) at São Paulo State University for Parkinson's disease research. Although the NewHandPD dataset has been primarily used in the context of Parkinson's disease diagnosis – particularly in tasks involving handwriting classification and disease severity estimation – our study introduces a novel application by repurposing the time-series motion data captured during the drawing of spirals and meanders. These signals, recorded using the Biometrical Smart Pen (BiSP), contain dynamic tremor-related information such as velocity and acceleration, which we extract and use as inputs for real-time adaptive filter simulations. To the best of our knowledge, this is the first study to employ these real patient-derived signals within a closed-loop adaptive control framework for tremor mitigation, representing a meaningful shift from diagnostic analysis to active intervention.

Additionally, RLS was selected due to its superior performance in time-varying signals and its faster convergence speed compared to the aforementioned algorithms. Lastly, the Kalman filter was chosen for its efficiency in predicting future cases and its ability to handle real-time changes effectively.

Adaptive filters

Adaptive algorithms are extensively used in scenarios involving signals with unknown or time-varying statistical properties, as they can automatically adjust internal parameters (e.g., coefficients or learning rate) to optimize performance, unlike fixed-coefficient algorithms with predefined parameters[18].

Least-Mean-Square – LMS

The LMS algorithm proposed by[21], is the most widely used adaptive algorithm, due to its low computational complexity, convergence proof in stationary environments, unbiased mean convergence to the Wiener solution, stability when implemented with finite precision arithmetic, and simplicity, with various modifications proposed by researchers over the years to improve filter efficiency[18]. The $(n+1)$-th weight-vector coefficients $\mathrm{w}(\cdot)$ is given by

$$\mathrm{w}_{n+1} = \mathrm{w}_n + 2\mu e_n \mathrm{x}_n \tag{1}$$

in which μ is the step-size, x_n is the input signal and n-th error is e_n and is given by the difference of output adaptive filter signal y_n and desired signal d_n,

$$e_n = d_n - y_n. \tag{2}$$

The desired signal is given by,

$$y_n = \mathrm{w}_n^{\top} \mathrm{x}_n. \tag{3}$$

Normalized Least Mean Square – NLMS

The NLMS algorithm addresses this problem by using a time-varying step size, which is particularly useful when the statistics of the input signals change rapidly. There are different approaches to developing the NLMS algorithm, such as formulating it as a constrained optimization problem, or recursively applying the conventional LMS algorithm multiple times for each new input sample[18,22]. Consider the LMS algorithm, (1) where the learning step μ is often defined as a fixed constant. However, in the NLMS algorithm, this value varies over time. Consequently, the corresponding equation is adjusted to incorporate this temporal variation. Since the learning step is time-varying, it is necessary to compute the a posteriori error, here denoted as e^{+}. For this purpose, the following equation is used:

$$e_n^{+} = d_n - \mathrm{w}_{n+1}^{T}\, \mathrm{x}_n. \tag{4}$$

Substituting (1) into (4), we obtain:

$$e_n^{+} = \left(1 - 2\mu \mathrm{x}_n^{T} \mathrm{x}_n\right) e_n. \tag{5}$$

The value of the time-varying learning step μ is given by:

$$\mu_n = \frac{1}{\xi + \|\mathrm{x}_n\|^2} \tag{6}$$

where $\|\cdot\|^2$ is the L_2 norm, $\xi \in \mathbb{R}$ is a small constant ($\gamma \approx$ 1E-10) that aims to prevent division by zero and is chosen by the user. By substituting the necessary equations and applying the appropriate simplifications, the following equation for the coefficient update can be obtained:

$$\mathrm{w}_{n+1} = \mathrm{w}_n + \frac{1}{\xi + \|\mathrm{x}_n\|^2} e_n \mathrm{x}_n. \tag{7}$$

Filtered-X Least Mean Square (Fx-LMS) In real-time active noise control applications, the Least Mean Squares algorithm is often not ideal due to its inherent delay, which can lead to instability as the noise at the system input may interfere with the adaptation process. To address this issue, Widrow[23] proposed an alternative solution by introducing an algorithm that estimates the input signal. In this approach, the reference signal is adaptively filtered and compared with the input signal to generate an estimated version of the primary noise signal. The convergence behavior of the Fx-LMS algorithm when processing stochastic input signals is analyzed in[24], specifically in the context of a moving average process. In this model, $P(z)$ represents the primary path transfer function, while $\widehat{\mathrm{S}}(z)$ denotes the estimated value of the secondary path transfer function $\mathrm{S}(z)$. Thus, the estimate of $\mathrm{S}(z)$ is given by

$$\widehat{\mathrm{S}}(z) = \sum_{q}^{Q-1} \frac{\mathrm{S}_q}{Z^q}. \tag{8}$$

The equation that represents the error of the system, in the time domain, can be described as

$$e_n = d_n - \sum_{q}^{Q-1} \mathrm{S}_q y_{n-q}. \tag{9}$$

Assuming the output $y(n) = \mathrm{w}_n^\top \mathrm{x}_n$, as well as in (2), substituting in (9), we obtain an error equation

$$e_n = d_n - \sum_{q}^{Q-1} \mathrm{S}_q \mathrm{x}_{n-q}^\top \mathrm{w}_{n-q}. \tag{10}$$

Using a methodology analogous to Wiener's optimal solution, an expression to the coefficients with filtered values is obtained, given by

$$\mathrm{w}_{\mathrm{opt}} = {\mathrm{R}_{f,n}}^{-1} \mathrm{p}_{f,n} \tag{11}$$

where $\mathrm{w}_{\mathrm{opt}}$ denotes the optimal vector of weights, $\mathrm{R}_f(n)$ denotes the correlation matrix of filtered entries, and $\mathrm{p}_f(n)$ represents the cross-correlation between the filtered input and the desired signal. The expression for the input $\mathrm{x}_f(n)$ is obtained by filtering $\mathrm{x}(n)$ using $\widehat{\mathrm{S}}(n)$:

$$\mathrm{x}_f(n) = \sum_{q}^{Q-1} \mathrm{S}_q \mathrm{x}_{n-q}. \tag{12}$$

The optimal value of the error measured by substituting (11) and (12) in (10), is given by

$$e_{\mathrm{opt}} = d_n - \mathrm{w}_{\mathrm{opt}}^\top \mathrm{x}_{f,n}. \tag{13}$$

From (13) the weight vector that updates the Fx-LMS algorithm is

$$\mathrm{w}_{n+1} = \mathrm{w}_n + \mu e_n \mathrm{x}_{f,n} \tag{14}$$

where μ, as in the LMS, is the convergence step of the algorithm. The Fx-NLMS algorithm extends the Fx-LMS approach by incorporating normalization of the input signal, similar to the NLMS algorithm ("Normalized Least Mean Square – NLMS"), which helps maintain stability and ensures efficient adaptation in real-time applications.

Recursive Least Squares – RLS

The RLS and LMS algorithms are widely used in signal processing and other data processing applications. Although both algorithms are popular, they exhibit significant differences in convergence speed and computational complexity[18,25]. Although the LMS algorithm is relatively simple and easy to implement, the RLS algorithm offers faster convergence at the cost of higher computational complexity. The RLS algorithm performs exceptionally well in environments with time-varying characteristics[18]. During the execution of the RLS algorithm, the covariance matrix is updated at each iteration using the following equation:

$$\mathrm{P}_{\lambda,n}^{-1} = \lambda^{-1} \left(\mathrm{P}_{\lambda,n-1}^{-1} - k_n \left[\mathrm{x}_n^T \mathrm{P}_{\lambda,n-1}^{-1} \right] \right) \tag{15}$$

where $\mathrm{P}_{\lambda,n}^{-1}$ denotes the inverse covariance matrix, λ represents the forgetting factor, x_n is the system's input signal, and k_n is the Kalman gain. The Kalman gain can be obtained by substituting (16) into (17), yields:

$$u_n = \mathrm{P}_{\lambda,n-1}^{-1} \mathrm{x}_n \tag{16}$$

$$k_n = \frac{1}{\lambda + \mathrm{x}_n^T u_n} u_n \tag{17}$$

To perform the adaptive filtering of the input signal, the filter utilizes the input vector x_n and the coefficient vector w_n, updated at each iteration of the algorithm, using:

$$y_n = \mathrm{w}_{n-1}^T \mathrm{x}_n \tag{18}$$

After computing the output of the filter, y_n, and giving the desired signal d_n, the filter error can be obtained by calculating the difference between these two signals, as illustrated in (2). To update the coefficient vector of the adaptive filter, the Kalman gain k_n and the error e_n are used, resulting in the following equation:

$$\mathrm{w}(n) = \mathrm{w}_{n-1} + k_n e_n. \tag{19}$$

The RLS filter uses (19) to find the best coefficients for the adaptive algorithm. Its goal is to minimize the mean square error (MSE) between the desired output and the filter output. As a result, the filter can adjust to changes in the input signal.

Kalman filter

The Kalman filter was first described in 1960 by Rudolf Emil Kalman[26], and it has since been widely applied in fields such as autonomous and assisted navigation. This filter can provide a recursive solution to the problem of optimal linear filtering in both stationary and non-stationary environments[27]. According to[28], the updated state estimates are calculated based on the previous estimate and the new input data, making the process more efficient as it avoids the need to compute the estimate directly from all past observed data at each filtering step. Furthermore, the Kalman filter enables estimating the past, present, and future states, even when the nature of the modeled system is unknown. According to[27], a discrete linear system can be described by:

$$\hat{y}_n = \mathrm{F} y_{n-1} + \mathrm{B} u_{n-1} + \mathrm{c}_{n-1} \tag{20}$$

$$\hat{z}_n = \mathrm{H}_n y_n + \mathrm{v}_n \tag{21}$$

where *y* represents the system state, F is the state transition matrix that correlates control to the state, B is the control matrix that correlates control to the state, c is the vector that transmits system error sources, v transmits measurement error sources, and H is the observation matrix that relates the system state to the measurement. The variables *c* and *v* are random, independent, and represent process and measurement noise, respectively. The inclusion of these variables allows for the determination of the covariance matrices, given by:

$$p(c) \approx N(0, \mathrm{Q}) \tag{22}$$

$$p(v) \approx N(0, \mathrm{R}) \tag{23}$$

where Q is the process noise covariance matrix, and R is the measurement noise covariance matrix. The Kalman filter consists of two main stages: *(i)* the prediction stage and *(ii)* the correction stage. In the prediction stage, the state estimate and error covariance propagation are computed using,

$$\hat{y}_n^- = \mathrm{F}_{n,n-1} \hat{y}_{n-1}^- \tag{24}$$

$$\mathrm{P}_n^- = \mathrm{F}_{n,n-1} \mathrm{P}_{n-1} \mathrm{F}_{n,n-1}^T + \mathrm{Q}_{n-1} \tag{25}$$

where $\hat{y}_n$ is the updated state estimate at the *n*-th time step (given the measurement at the *n*-th time step), F_n is the state transition matrix relating the current state to the next state, $\hat{y}_{n-1}$ is the state estimate at the *n*-th time step (given the information available at the $(n-1)$-th time step), P_n is the error covariance matrix of the predicted state, and Q_{n-1} is the process noise matrix. The second stage of the Kalman filter is the correction stage. In this stage, the Kalman gain is given by:

$$k_n = \mathrm{P}_n^- \mathrm{H}_n^T \left[\mathrm{H}_n \mathrm{P}_n^- \mathrm{H}_n^T + \mathrm{R}_n\right]^{-1} \tag{26}$$

where H_n represents the observation matrix that relates the system state to the measurement at the *n*-th time step, and R_n is the measurement noise covariance matrix. After computing the Kalman gain, this value is used to update the state estimate and error covariance, using

$$\hat{y}_n = \hat{y}_n^- + k_n(\mathrm{x}_n - \mathrm{H}_n \hat{y}_n^-) \tag{27}$$

$$\mathrm{P}_n = (\mathrm{I}_n - k_n \mathrm{H}_n) \mathrm{P}_n^- \tag{28}$$

where k_n is the Kalman gain, responsible for determining the weight assigned to the current measurement in the state estimate update, x_n represents the measurement at time step *n*, and I_n represents the identity matrix.

Proposed solution

The project aims to develop a prototype adaptive pen to mitigate hand tremors while writing, using a system based on adaptive devices and algorithms. The pen uses a Teensy 4.1 microcontroller to process the signals captured by the ADXL345 accelerometer, which detects the three-dimensional oscillations of tremors. Based on these signals, a vibration motor is activated to provide tactile feedback to the user, promoting the attenuation of tremors.

The Teensy 4.1 microcontroller, chosen for its high processing capacity and efficiency in embedded projects, is responsible for executing an adaptive algorithm in real-time. This algorithm adjusts to the individual characteristics of each user's tremors, identifying patterns, and controlling the vibration motor to minimize unwanted oscillations. The adaptability of the system is one of the main differentiators of the prototype, allowing it to adjust to different tremor profiles. The ADXL345 accelerometer was selected due to its accuracy and low power consumption, which are crucial features for effective detection of tremors in the three directions (X, Y, Z). This sensor uses a micromachined polysilicon surface suspended on a silicon wafer, providing high resistance and accuracy in measuring deflections caused by acceleration. Communication with the microcontroller is carried out through digital interfaces such as I2C or SPI, ensuring efficient integration into the system.

The 3V vibration motor, commonly known as a "coin motor", is integrated into the system to provide robust haptic feedback to the user. With a rotation of up to 9000 rpm, the motor operates efficiently within a voltage range of 2.5V to 4V. Figure 1 illustrates a demonstration of the adaptive pen prototype, highlighting the arrangement of the main components. The combination of its compact dimensions and high energy efficiency makes it ideal for applications in portable devices such as the proposed pen.

Results and discussion

In this section, the results of the tests conducted to validate the functionality of the prototype and demonstrate its effectiveness in controlling tremors will be presented. It is important to note that, up to this point, validation tests have been conducted in a laboratory setting and have not involved human participants.

Initially, computational simulations were carried out to evaluate the performance of the selected adaptive algorithms. To perform these tests, the hand tremor signals of 31 PD patients, available in the NewHandPD dataset, were used. The handwritten dynamics were derived from the dynamic trajectory data recorded by the BiSP device during spiral drawing tasks, as demonstrated by Pereira et al. (2019)[19], using a smart pen BiSP Although the dataset's public description highlights the image set, the full acquisition included motion data recorded at 200 Hz.

For the simulations, we have used a plant set to the dimension of 2×1, i.e., $N = 2$. Filter design parameters were defined by exhaustive trials, considering the best results in terms of convergence speed and MSE. Based on the simulation results, the algorithms yielding the lowest MSE and fastest convergence times were subsequently implemented on the microcontroller, as detailed in Sections "Performance analysis of the RLS and Kalman Algorithms on the microcontroller" and "Real-time performance analysis of Kalman Filter on the microcontroller". Figure 2 represents the overview of the tests to validate our prototype.

Comparative analysis of adaptive algorithms

The adaptive algorithms selected for the simulation tests were based on the research carried out by[15]. The Fx-LMS, Fx-NLMS, and Fx-NLMS&LMS algorithms were chosen due to their demonstrated effectiveness in tremor attenuation. While the mentioned study employed synthetic tremor signals generated from mathematical models, this work applies real signals from patients with Parkinson's disease. Additionally, the RLS and Kalman Filter algorithms were included due to their proven ability in temporal series reconstruction and convergence speed. To establish a fair comparison between the algorithms, they were all adjusted to have the maximum MSE in steady-state, according to[15,29]. For the tested algorithms we have considered, μ_1 and $\mu_2 \in \{1E{-}4, 2E{-}4, \ldots, 1\}$, $\delta = 0.01$ and $\lambda = 0.999$. Table 1 presents a summary of the performance of the tested algorithms – Fx-LMS ($\mu_1 = 5E - 4$), Fx-NLMS ($\mu_2 = 8E - 3$), Fx-NLMS&LMS ($\mu_1 = 5E - 3$ and $\mu_2 = 8E - 3$) and RLS ($\lambda = 0.999$) and Kalman.

The MSE behavior of the algorithms can also be seen in Fig. 3. They were all normalized so that they started at the same point. It is important to highlight that, although all algorithms showed a decreasing trajectory,

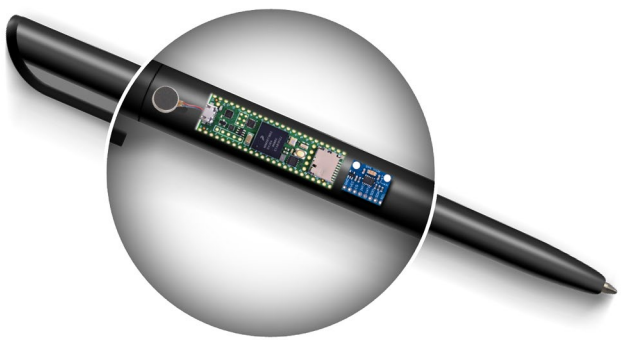

Fig. 1. Illustration of the adaptive pen prototype.

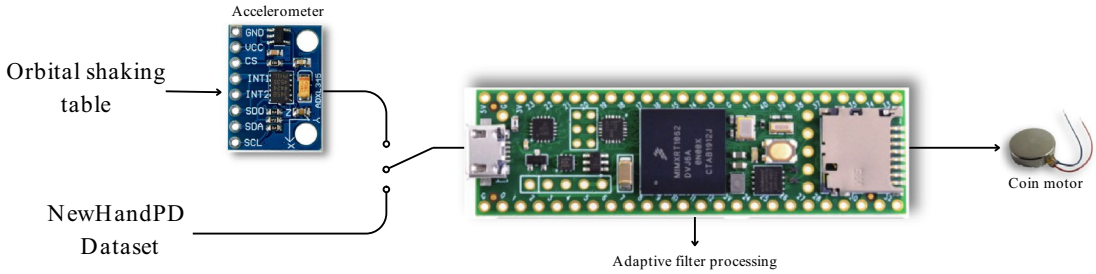

Fig. 2. Summary of Testing on the Teensy 4.1 Microcontroller.

Algorithm	MSE_{∞}	n
Fx-LMS	-38 dB	600
Fx-NLMS	-42 dB	400
Fx-NLMS&LMS	-44 dB	350
RLS	-53 dB	300
Kalman	-50 dB	50

Table 1. Summary of results obtained from computational simulations.

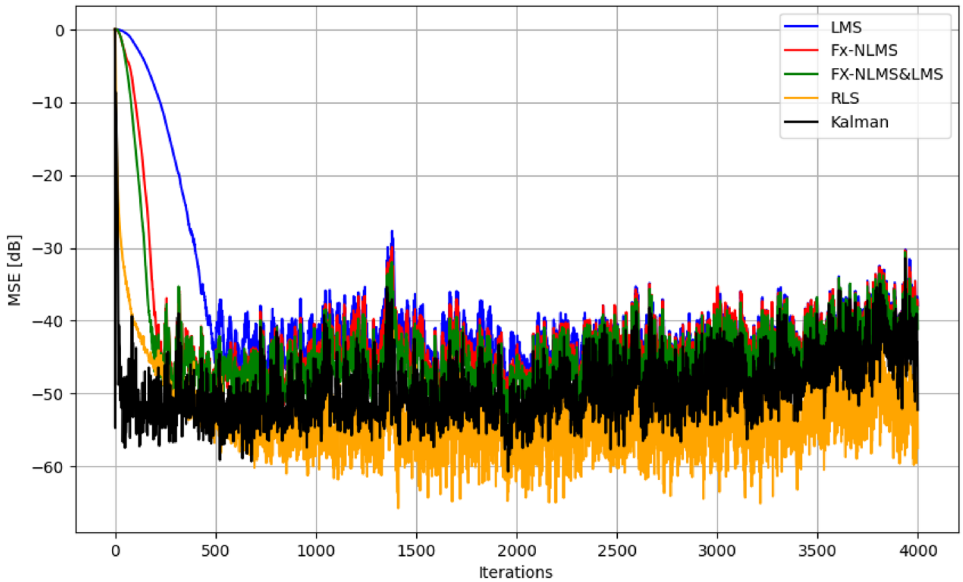

Fig. 3. Comparison of the normalized MSE behavior over 30 runs: Monte Carlo simulation of Fx-NLMS, Fx-LMS&NLMS, RLS, Fx-LMS algorithms, and Kalman filter.

their convergence rates varied. Some curves demonstrated a faster reduction in MSE, indicating a more efficient adaptation, while others showed a more gradual decrease. Based on the information provided, it is highlighted that the RLS algorithm obtained the lowest MSE in steady-state among the algorithms tested. On the other hand, the Kalman filter stood out for its fast convergence rate compared to other algorithms. Another relevant point to be observed is the possibility of oscillations occurring during the adaptation process. These oscillations may arise from specific characteristics of the signal or the complex interaction between the algorithm parameters and the system dynamics.

It is important to note that a direct comparison with previous studies was not performed, as it would be unfair given that most of those works rely on synthetic or proprietary datasets, unlike our approach based on real, publicly available data.

Performance analysis of the RLS and Kalman algorithms on the microcontroller

In this section, we present the results obtained by implementing the RLS and Kalman Filter algorithms in the microcontroller, using data from the NewHandPD dataset, as in the simulations performed in

Section "Comparative analysis of adaptive algorithms". Tremor signals from 31 patients with Parkinson's disease were used, with 4000 samples per patient for each practice. The tests focused on evaluating the performance of algorithms running on the microcontroller. It is important to emphasize that, at this stage, it was not considered the real tremor signs acquired by the accelerometer or the activation of the vibration motor. During the tests, the adaptive filter code was embedded in the microcontroller and then, the RLS algorithm, followed by the Kalman Filter, was performed. The data were stored in text files on a memory card and later processed on a computer to calculate the MSE. The results, shown in Fig. 4, revealed that the RLS algorithm presented a lower steady-state MSE than the Kalman Filter, as expected. However, the Kalman Filter stood out for converging approximately eight times faster than the RLS, despite having a slightly higher final MSE. This rapid adaptation of the Kalman Filter makes it more suitable for attenuating tremors with fewer iterations, justifying its choice for the continuation of the project.

A decrease in accuracy (increase in MSE) and a greater number of iterations were also observed for the algorithms to converge in steady-state, due to the limitations of the microcontroller compared to the Monte Carlo simulations performed in Section "Comparative analysis of adaptive algorithms". Despite this, both algorithms maintained satisfactory performance, demonstrating effectiveness in adapting to the input signal and attenuating the tremors.

Real-time performance analysis of Kalman Filter on the microcontroller

A new performance validation was performed to assess the system's effectiveness in acquiring vibrations, processing them, and activating the response motor. In this evaluation, the Kalman filter was used exclusively, as it demonstrated a higher adaptation speed than the RLS method in previous tests. This choice is justified by prioritizing convergence time over minimizing the MSE. The prototype was subjected to tests on an orbital shaking table, providing a controlled environment for the execution of the experiment. In the first stage, the shaking table speed was maintained constant, to ensure that, once the adaptive algorithm had learned the characteristics of the input signal, there was no need for relearning. It was made to guarantee the consistency of performance throughout the execution. The mean result, after 31 executions, is shown in Fig. 5, indicating that the algorithm reached a steady state after 60 iterations and MSE of around -37 dB.

In the second stage, a speed variation in the shaking table was included during the execution of the adaptive algorithm, simulating the hand tremors.

We have introduced variations to evaluate the algorithm's tracking capability, which means the algorithm has to relearn the characteristics of the input signal to reach the optimal solution. Figure 6 shows that, despite this need, the algorithm could adapt according to the signal changes, reaching an MSE of around -26 dB. In total, 31 trials were conducted, corresponding to one trial for each patient signal extracted from the NewHandPD dataset. The tests on the orbital shaking table were performed under both constant and variable speed conditions to replicate the dynamics of real tremor signals as captured by the accelerometer. This procedure enabled us to validate the traceability and responsiveness of the adaptive algorithms in processing and reacting to realistic tremor patterns, even though no human participants were directly involved at this stage. Despite this, the evaluation demonstrated that the adaptive algorithm fulfilled its purpose, adjusting to changes and optimizing the response throughout the process.

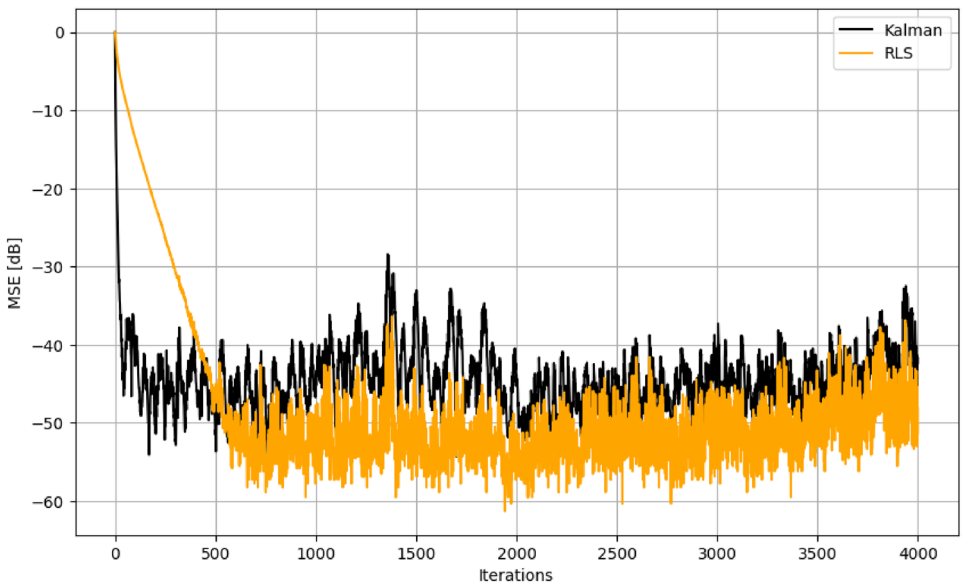

Fig. 4. Normalized MSE behavior of Kalman filter and RLS algorithm running on the microcontroller Teensy 4.1.

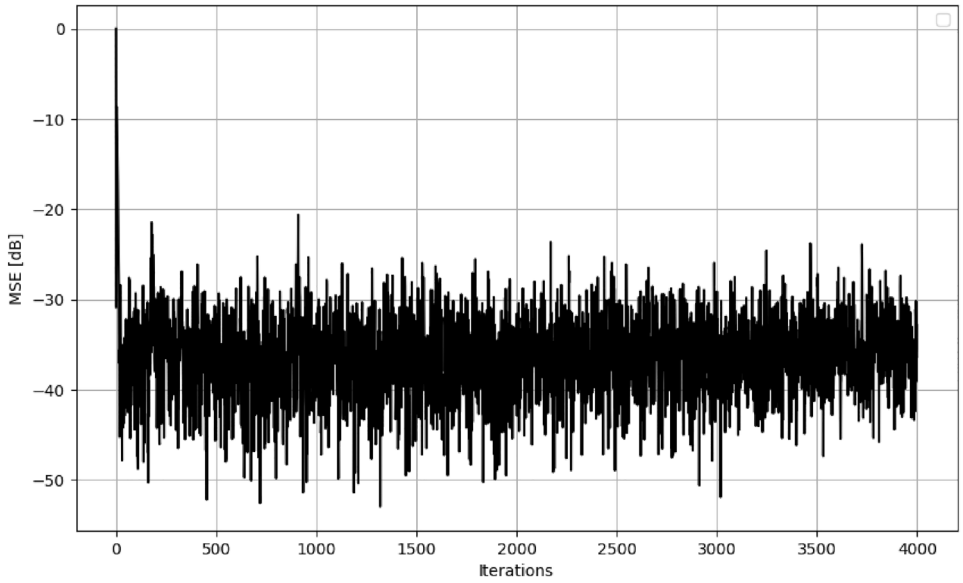

Fig. 5. MSE behavior using Kalman filter: Stationary environment.

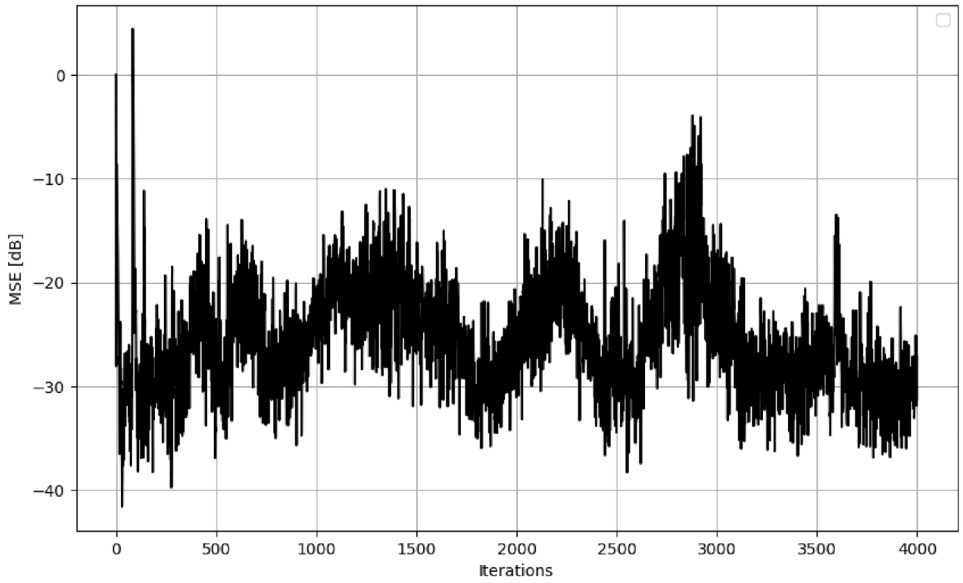

Fig. 6. MSE behavior using Kalman filter: Time-varying environment.

Conclusion

Writing is an essential skill in everyday life, but it can be severely compromised by hand tremors, which can harm the quality and legibility of text. In response to this need, this study consisted of the design and development of a prototype of an adaptive pen, designed to mitigate hand tremors, a problem that significantly compromises writing in professional and personal contexts. The application of adaptive algorithms was effective, with emphasis on the Kalman algorithm, which demonstrated agile and precise adaptation, making it the ideal choice for implementation in the prototype.

The results obtained validate the robustness of the algorithms and the effectiveness of the pen in providing more stable writing, meeting the initial objectives of the project. The possibilities for improvements are vast, including the integration of additional sensors, such as pressure sensors and gyroscopes, which can further improve the accuracy of the system. The adoption of more advanced microcontrollers offers an opportunity to optimize real-time performance, crucial for assistive devices.

Although this study reported average performance metrics demonstrating the feasibility of the adaptive pen prototype, we acknowledge that complementary statistical analyses – including variance, standard deviation, and confidence intervals – would provide a more thorough understanding of the system's robustness and

reproducibility. As the current validation was carried out in a controlled laboratory environment without human participants, future research will extend these analyses to tests involving real users performing handwriting tasks, thereby strengthening the evaluation of the system under natural, subject-specific tremor conditions.

These advances could not only improve the user's experience but also significantly contribute to the field of assistive devices focused on quality of life.

Data availability

The datasets used and analysed during the current study are publicly available in the HandPD repository: https://wwwp.fc.unesp.br/~papa/pub/datasets/Handpd/. HandPD is a collection of 736 JPEG images of spirals and meanders drawn by 92 individuals (18 healthy controls, 74 Parkinson's patients) at São Paulo State University for Parkinson's disease research.

Received: 26 March 2025; Accepted: 29 July 2025
Published online: 06 August 2025

References

1. World Health Organization. Mental health: Neurological disorders (2016). Accessed: June 2023.
2. Kelley, C. R. & Kauffman, J. L. Optimal control perspective on parkinson's disease: Increased delay between state estimator and controller produces tremor. *IEEE Trans. Neural Syst. Rehabil. Eng.* **28**, 2144–2152. https://doi.org/10.1109/TNSRE.2020.3018626 (2020).
3. Louis, E. D. Essential tremor. In *Handbook of clinical neurology*, vol. 196, 389–401 (Elsevier, 2023).
4. Pahwa, R. *et al.* An acute randomized controlled trial of noninvasive peripheral nerve stimulation in essential tremor. *Neuromodulation* 537–545, https://doi.org/10.1111/ner.12930(2019).
5. Pasquini, J. et al. The clinical profile of tremor in parkinson's disease. *Mov. Disord. Clin. Pract.* **10**, 1496–1506 (2023).
6. Meng, J. et al. A detection method for parkinson's hand tremor based on machine learning. In *China Automation Congress (CAC)*, 4105–4109, https://doi.org/10.1109/CAC53003.2021.9728408(2021).
7. Foltynie, T. et al. Medical, surgical, and physical treatments for parkinson's disease. *The Lancet* **403**, 305–324 (2024).
8. Habibollahi, Z. et al. Tremor suppression using functional electrical stimulation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **32**, 3289–3298. https://doi.org/10.1109/TNSRE.2024.3453222 (2024).
9. Lora-Millan, J. S., Delgado, G., León, J. B. & Rocon, E. A review on wearable technologies for tremor suppression. *Front. Neurol.* **12**, https://doi.org/10.3389/fneur.2021.700600(2021).
10. Chandra, G., Gandhi, T. K. & Singh, B. Designing controllers for hand tremor suppression using model simplification. *Biomed. Signal Process. Control* **96**, 106483 (2024).
11. Van, H. P. & Ngo, H. Q. T. Developing an assisting device to reduce the vibration on the hands of elders. *Appl. Sci.* **11**, https://doi.org/10.3390/app11115026(2021).
12. Liu, C. & Zhang, K. A wearable finger tremor-suppression orthosis using the pvc gel linear actuator. *IEEE Robot. Autom. Lett.* **9**, 3854–3861. https://doi.org/10.1109/LRA.2024.3369492 (2024).
13. Zamanian, A. H. & Richer, E. Adaptive notch filter for pathological tremor suppression using permanent magnet linear motor. *Mechatronics* **63**, 102273. https://doi.org/10.1016/j.mechatronics.2019.102273 (2019).
14. Jamaludin, H. B. et al. Active tremor control in human-like hand tremor using fuzzy logic. *Indones. J. Electr. Eng. Comput. Sci.* **24**, 108–115. https://doi.org/10.11591/ijeecs.v24.i1.pp108-115 (2021).
15. Araujo, R. S. A. et al. Analysis of adaptive algorithms based on least mean square applied to hand tremor suppression control. *Appl. Sci.* **13**, https://doi.org/10.3390/app13053199(2023).
16. Madkour, A., Hossain, M. A., Dahal, K. P. & Yu, H. Intelligent learning algorithms for active vibration control. *IEEE Trans. Syst. Man Cybern. Part C (Applications and Reviews)* **37**, 1022–1033. https://doi.org/10.1109/TSMCC.2007.900640 (2007).
17. Toro-Ossaba, A., Tejada, J. C., Rúa, S., Núñez, J. D. & Peña, A. Myoelectric model reference adaptive control with adaptive kalman filter for a soft elbow exoskeleton. *Control Eng. Pract.* **142**, 105774. https://doi.org/10.1016/j.conengprac.2023.105774 (2024).
18. Diniz, P. S. R. *Adaptive Filtering* (Springer, 2012).
19. Pereira, C. R., Weber, S. A. T., Hook, C., Rosa, G. H. & Papa, J. P. Deep learning-aided parkinson's disease diagnosis from handwritten dynamics. In *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, 340–346, https://doi.org/10.1109/SIBGRAPI.2016.054(2016).
20. Pereira, C. R. et al. A new approach to diagnose parkinson's disease using handwriting analysis. *Comput. Methods Programs Biomed.* **136**, 65–76. https://doi.org/10.1016/j.cmpb.2016.08.023 (2016).
21. Widrow, B. & Hoff, M. Adaptive switching circuits. In *1960 IRE WESCON Convention Recor*, 96–104 (1960).
22. Farhang-Boroujeny, B. *Adaptive Filters: Theory and Applications* (Wiley, 2013).
23. Widrow, B. et al. Adaptive noise cancelling: Principles and applications. *Proc. IEEE* **63**, 1692–1716. https://doi.org/10.1109/PROC.1975.10036 (1975).
24. Ardekani, I. T. & Abdulla, W. H. Theoretical convergence analysis of fxlms algorithm. *Signal Process.* **90**, 3046–3055 (2010).
25. Bull, D. R., Wu, M., Chellappa, R. & Theodoridis, S. *Academic Press Library in Signal Processing: Image and Video Compression and Multimedia*. ISSN (Elsevier Science, 2014).
26. Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering* **82**, 35–45. https://doi.org/10.1115/1.3662552 (1960).
27. Welch, G. & Bishop, G. An Introduction to the Kalman Filter (2006).
28. Haykin, S. *Kalman Filtering and Neural Networks* (Signal Processing, Learning, Communications and Control (Wiley, Adaptive and Cognitive Dynamic Systems, 2004).
29. Borges, R. C., Parreira, W. D. & Costa, M. H. Design guidelines for feedforward cancellation of the occlusion-effect in hearing aids. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 607–610 (IEEE, 2019).

Author contributions

J.C.T. carried out the research and drafted the initial manuscript. L.A.S. and R.C.B., along with all other authors, contributed to the critical revision of the manuscript and approved the final version. W.D.P. and A.F. served as project managers, providing overall guidance and oversight for the study.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to L.A.S. or W.D.P.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.