Article

Design and Implementation of DSLMS Algorithm Based Photoelectric Detection of Weak Signals

Yang Wang [1,*], **Min Wang** [1], **Zishuo Song** [2] **and Weihao Bian** [1]

1 School of Control Science and Engineering, Tiangong University, Tianjin 300387, China; wangmin@tiangong.edu.cn (M.W.); 2331091152@tiangong.edu.cn (W.B.)
2 School of Information Engineering, Ningxia University, Yinchuan 750021, China; song56569@163.com
* Correspondence: 2331091154@tiangong.edu.cn; Tel.: +86-1661-161-6300

Abstract: Accurately extracting weak signals is extremely important for overall performance and application in optoelectronic imaging and optical communication systems. While weak signals are susceptible to noise, adaptive filtering is a commonly used noise removal method. Still, its convergence speed is slow, the steady-state error is large, and the anti-interference ability is weak. To solve the above problems, this paper proposes a new type of variable-step-length adaptive filtering algorithm (DSLMS) based on the minutiae function, which effectively reduces the noise component in error through its combination with the pair cancelation system, utilizing the low correlation property of the noise signal, to improve the anti-noise interference ability of the adaptive filter. Using FPGA and Matlab (2018b) for experimental verification, the results show that this algorithm shows significant advantages in noise suppression, accelerated algorithm convergence, and low steady-state error, and it has effectiveness and application potential for the optoelectronic detection of weak signal processing.

Keywords: photoelectric detection technology; weak signal extraction; LMS algorithm; counteracting system; FPGA; noise suppression

1. Introduction

In the actual signal acquisition environment, the effective signals of some scenes could be stronger and accompanied by a large amount of environmental noise. This often makes the acquired signals non-smooth and increases the difficulty of accurately extracting the effective signals from the noisy environment. Researchers have proposed the adaptive filtering technique to improve the signal-to-noise ratio of weak signals in complex noise environments. Reference [1] introduces an adaptive time delay estimation algorithm for low-signal-to-noise-ratio environments that can accurately estimate the filtering results. Reference [2] proposed a method combining LMS and RLS filters, which is able to obtain results close to Kalman filtering performance with low computational complexity. Reference [3] described the principle of active noise reduction and submitted a patent application. Reference [4] proposed a new normalized LMAT (NLMAT) algorithm that outperforms existing algorithms in various noise environments. Reference [5] proposed a fast and stable normalized least mean quadrature (FSNLMF) algorithm with faster convergence. This technique enhances the optimization of the signal-to-noise ratio in the signal-processing process by dynamically adjusting the filtering parameters to adapt to environmental changes. It improves the accuracy and reliability of signal detection in complex noise backgrounds. Therefore, adaptive filtering techniques have been widely used and thoroughly studied in the fields of radar, sonar, communication and navigation systems, etc. [6]. Widrow and his colleagues introduced the least mean square error (LMS) algorithm in the early 1980s, simplifying the computational process, reducing the difficulty of implementation, and enhancing adaptability [7]. This algorithm has been applied in numerous fields due

to its superior performance, such as radar beam formation, adaptive interference noise cancelation, and next-generation mobile communication technologies [8–11].

Traditional least mean square error (LMS) algorithms use a fixed-step-size approach (hereafter referred to as fixed-step-size LMS algorithms), which limits their ability to achieve fast convergence and maintain low steady-state errors. Researchers have conducted extensive studies to address this issue and optimize the performance of fixed-step-size LMS algorithms. A variable-step-size LMS algorithm that employs an S-function to regulate the step size is introduced in the literature [12]. This method dynamically adjusts the step size through the S-function by assigning a more significant step size to quickly approach the optimal solution and decreasing the step size to improve the accuracy as the convergence point approaches. This variable-step-size strategy significantly improves the convergence speed and reduces the steady-state error compared to the fixed-step-size LMS algorithm. However, the steep variation in the S-function near the zero point causes the algorithm to change the step size too fast as it approaches convergence, which increases the error at the steady state of the algorithm. In the literature [13], by applying translation and flip-flop transformations to the S-function, coupled with introducing new parameters to optimize the change at the bottom of the function, the researchers succeeded in improving the performance of the variable-step-size LMS algorithm. Although this approach brought about a performance improvement, the complexity of the algorithm model adversely affected its flexibility. Subsequently, reference [14] explored a new variable-step-size LMS algorithm that draws on the Q-function with the properties of the S-function curve and adjusts the step size using a compensating term inter-correlation function for the relative error. Due to the inherent characteristics of the Q-function, this algorithm faces the same challenge of significant steady-state errors as it approaches the convergence point. Reference [15] employs a gradient statistical averaging-based approach to modulate the step-size factor, aiming to speed up the algorithm's convergence and optimize its steady-state error performance. However, the drawback of this approach lies in the weak immunity to interference caused by the introduction of a judgment threshold that increases the algorithm's complexity. Reference [16] introduced a variable-step-size LMS algorithm based on exponential functions by using exponential functions. Still, the algorithm also suffers from high complexity due to the frequent execution of exponential operations. To solve this problem, reference [17] improved on [16] by applying the variable-step-size method to the partial update of the filter weight coefficients, improving the convergence speed and effectively reducing the algorithm's complexity. However, in low-signal-to-noise-ratio environments, the algorithm has a more significant step size when approaching convergence, and the steady-state performance is poor.

In summary, the existing variable-step-size LMS algorithms cannot simultaneously solve the problems of noise interference, slow convergence speed, and higher steady-state error. In order to effectively reduce the influence of environmental noise, speed up the convergence of the algorithm to the optimal solution, and achieve a lower error level after reaching the steady state, this paper proposes a variable-step-length adaptive filter based on the improved skip-link function, in which the DSLMS adaptive filtering algorithm is used, with the following main contributions:

(1) A new variable-step-size adaptive filtering algorithm is proposed based on the minutiae function, which is combined with the pairwise elimination system to effectively reduce the noise component in the error by utilizing the characteristic of low correlation of the noise signal and improve the anti-noise-interference capability of the adaptive filter.

(2) The adaptive filter effectively improves the convergence speed and significantly reduces the steady-state error by segmentally adjusting the step factor of the weight coefficients.

(3) Using FPGA technology as an experimental platform, this study modularizes the design of the DSLMS algorithm and compares it with the traditional LMS algo-

rithm through experiments. The results show that the proposed algorithm has better performance.

2. Related Work

2.1. Theoretical Analysis of Adaptive Filters

Adaptive filters deal with weak signals without knowing the statistical characteristics of the input signal and noise; the filter itself can learn or estimate the statistical characteristics of the signal during the working process [18] and adjust its parameters based on this to achieve the optimal filtering effect under a specific cost function [19]. LMS algorithmic filters are a kind of adaptive filter widely used in noise cancelation, echo cancelation, spectral line enhancement, channel equalization, system identification, etc. [20–22].

As shown in Figure 1, the LMS algorithm includes two essential parts: filtering and adaptive processes. The filtering process consists of two key steps: Firstly, in the adaptive filtering process, the input signal, $i(n)$, first undergoes a multilevel structure consisting of delay units $\left(Z^{-1}\right)$ and weights, $w_0(n), w_1(n), \cdots, w_{M-1}(n)$, which is responsible for implementing the signal weighting and accumulating it to form the output signal, $o(n)$ [23]. The output of each stage consists of the product of the corresponding weights and the delayed version of the input signal, which is summarized into the final output of the filter.
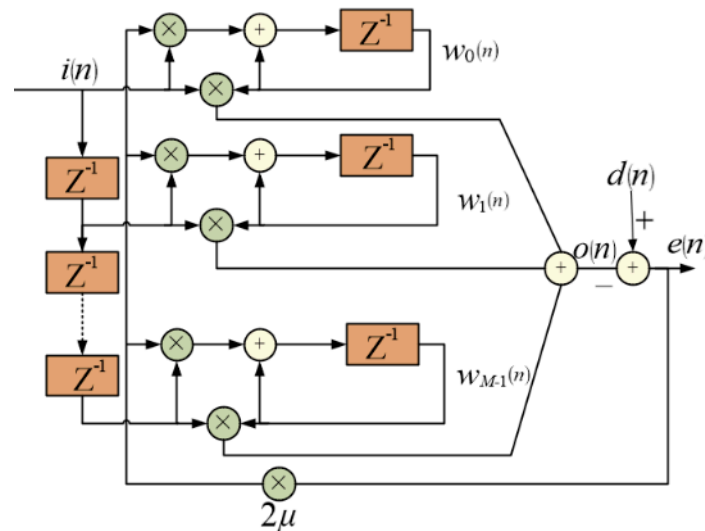

Figure 1. Schematic diagram of the LMS algorithm.

The adaptive adjustment mechanism is based on the error, $e(n)$, between the output signal, $o(n)$, and the desired signal, $d(n)$. This error signal is used to guide the updating of the weight parameters according to a gradient descent algorithm, where the weight adjustment is proportional to the product of the error signal and the input signal, and a step factor (µ) controls the update amplitude. Such a weight-updating process aims to gradually reduce the mean square error between the output and the expectation, thus optimizing the overall performance of the filter [24].

The horizontal filter input signal vector, $i(n)$, is denoted as $i(n) = [i(n), i(n+1), \cdots, i(n-M+1)]^T$, where M is the order of the filter. The weight vector, $w(n)$, of the filter is $w(n) = [w_0(n), w_1(n), \cdots, w_{M-1}(n)]^T$, in this case, the output, $o(n)$, of the filter is calculated as shown in Equation (1):

$$o(n) = w^T(n)i(n) \tag{1}$$

Equation (2) is the estimation error at the nth moment.

$$\begin{aligned} e(n) &= d(n) - o(n) \\ &= d(n) - w^T(n)i(n) \end{aligned} \tag{2}$$

Equation (3) is the lateral filter's mean square error criterion cost function.

$$J(n) = E\left\{|e(n)|^2\right\} = E\left\{\left|d(n) - w^T(n)i(n)\right|^2\right\} \tag{3}$$

The gradient descent method is applied to adjust the values of the adaptive filter's weight coefficient vectors along the performance surface's negative gradient direction. Equation (4) is the iterative formula for calculating the weight vector, w.

$$w(n+1) = w(n) + \mu(-\nabla J) \tag{4}$$

where μ is the step factor, and $\mu > 0$.

The transversal filter is optimized by seeking a weight vector that minimizes the mean square error cost function. For a generalized smooth process when the number of iterations tends to infinity, the expectation of this weight vector estimate is close to the Wiener optimal solution, w_{opt} [25].

Although the traditional least mean square error (LMS) adaptive filtering algorithm performs well in many aspects, the algorithm suffers from several shortcomings in applications dealing with weak signals. First, the traditional LMS algorithm uses the product of the instantaneous error and the weight vector, $e(n)i(n)$, as an approximation of the gradient vector, $\nabla J(n)$. Since this approximation does not include the computation of the expectation value in the weight-updating process, its iterative step lacks robustness under the influence of noise. Second, the step factor (μ) plays a decisive role in the weight-updating process, which directly affects the response speed of the filter parameter tuning as well as the rate of minimizing the error. In a boisterous signal environment, too small a value of μ slows down the system's adaptation to changes in the signal environment. It prolongs the time for the system to converge to the desired signal. In less noisy signal environments, larger values of μ speed up the weight update and cause over-adjustment and oscillations in the system. Such oscillations not only lead to convergence problems of the filter near the global optimum solution but also degrade the system's performance, leading to signal distortion and increased steady-state error.

2.2. Principle of the Pair of Eliminators

Noise-counteracting techniques originated in the mid-1960s by a team of researchers at Stanford University and are now widely used in various environments. The basic principle of noise counter-cancelation is shown in Figure 2 [26–28].
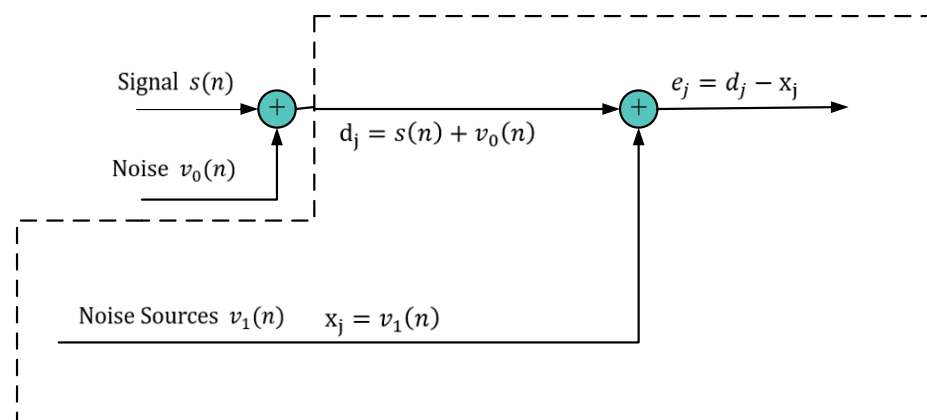

Figure 2. Noise canceller schematic diagram.

Noise cancelation systems are a vital signal-processing technique that operates through two main inputs: the primary input, d_j, and the reference input, x_j. The primary input signal, d_j, combines the desired signal, $s(n)$, and the accompanying noise, $v_0(n)$. The

reference input signal, $v_1(n)$, which is usually obtained from an ambient noise source, is associated with the noise component, $v_0(n)$, in the primary input but is independent of the desired signal, $s(n)$. In a noise cancelation system, a reference signal, $v_1(n)$, generates an output designed to match and neutralize the primary signal's noise component, $v_0(n)$. The error signal $e_j = d_j - x_j$ near-clean signal $s(n)$ is obtained by subtracting the estimated noise, y_j, from the primary input signal, d_j.

The noise pair cancelation technique can identify and cut down unwanted noise components employing a reference noise signal. Although the original signal, d_j, is interfered with by the noise, $v_0(n)$, the pair cancelation system ensures that the final error signal (e_j) maximally reflects the desired signal, $s(n)$, by accurately estimating and cutting the reference noise, $v_1(n)$, which improves the signal-to-noise ratio of the desired signal [29].

Assuming that the signals $s(n)$, $v_0(n)$, and $v_1(n)$ are smooth, the expectation of the square of the output error mode $|e_j|$ of the filter is

$$E\left[|e_j|^2\right] = E\left[|s + v_0 - v_1|^2\right] = E\left[|s|^2\right] + E\left[|v_0 - y_j|^2\right] + 2E\left\{Re\left[s \times (v_0 - y_j)\right]\right\} \quad (5)$$

Since the useful signal $s(n)$ is uncorrelated with the noise signal $v_0(n)$ and the reference signal $v_1(n)$, the

$$E\left[|e|^2\right] = E\left[|s|^2\right] + E\left[|v_0 - y_j|^2\right] \quad (6)$$

Because the power of the sound signal is not affected by the filter power vector, the moment when $E\left[|v_0 - y_j|^2\right]$ reaches its minimum value is the moment when $E\left[|e|^2\right]$ reaches its minimum value. When $v_0 - y_j = 0$, i.e., when $v_0 = y_j$, the minimum value is obtained.

$$E_{min}\left[|e_j|^2\right] = E\left[|s|^2\right] \quad (7)$$

Equations (5)–(7) show that an ideal noise counter-cancelation system can eliminate the difference between the estimated noise and the actual noise to a large extent, thus leaving only the effective signal. However, in real dynamically changing noise environments, the counter-cancelation system may not effectively respond to changes in noise characteristics due to its lack of real-time adjustability, which results in incomplete noise cancelation and prevents the system from adapting quickly when noise characteristics change.

3. DSLMS Adaptive Filter

In the field of weak signal processing, although the traditional fixed-step-size LMS algorithm is widely used due to its simplicity, its fixed and unchanging weight-vector step factor makes it difficult to simultaneously achieve the improvement of convergence speed and the reduction in steady-state error when dealing with signals and noises in a variable environment [30].

To solve the above problems, this paper proposes a variable-step-size adaptive filter based on the improved micro tangent trace function, which adopts the DSLMS algorithm, and by optimizing the step-size adjustment mechanism, it speeds up the convergence speed, reduces the steady-state error, and effectively improves the processing performance of weak signals. By combining the DSLMS algorithm with the pair cancelation system, the signal can be separated from the noise more effectively, the anti-interference ability of the system can be improved, and the accuracy and efficiency of signal processing can be improved through more accurate signal reconstruction. The principle of the DSLMS adaptive filter is shown in Figure 3.
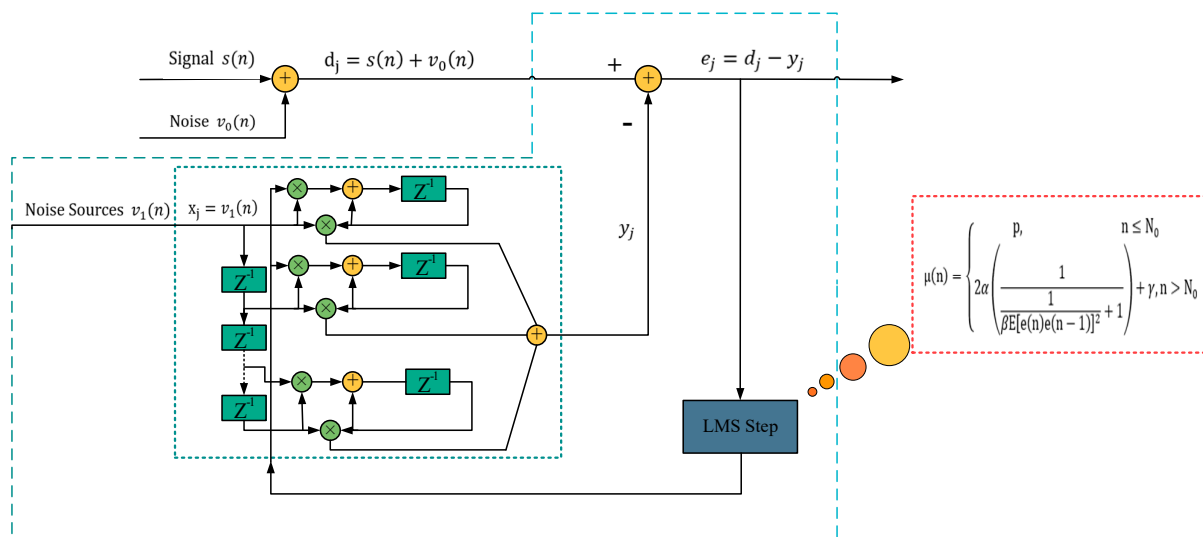
Figure 3. DSLMS filter structure schematic diagram.

3.1. Algorithmic Principles

This study proposes a variable-step-size adjustment strategy based on the minutiae function by taking the inverse of the independent variable (x) of the minutiae function to obtain a new step-size factor-taking function, as shown in Equation (8).

$$\mu(x) = 1 - \frac{1}{1 + x^2} \tag{8}$$

where x denotes the autocorrelation estimate based on the current error versus the previous moment's error, bringing e_j from Figure 3 into the equation.

From Figure 4a, it can be seen that the μ(x) curve conforms to the step-size adjustment principle of the adaptive filtering algorithm in the absence of noise interference or relatively small noise interference in the early stage of the algorithm (the initial stage of algorithmic convergence). The step-size adjustment function provides the algorithm with a more significant value of the step size (μ) to improve the convergence speed of the algorithm and to enable it to quickly transit to the stage of the algorithm's convergence completion. In the convergence completion phase, a smaller step size value (μ) is offered to keep the algorithm more stabilized in that phase.
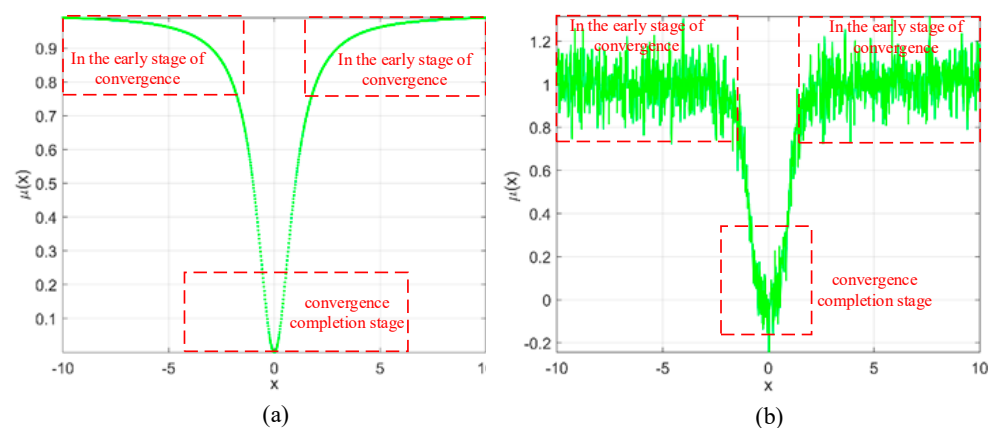

Figure 4. Convergence performance analysis graph. (**a**) Algorithm convergence performance with no or relatively low noise interference (**b**) Algorithm convergence performance in case of severe noise interference.

However, in the environment with more serious noise interference, as shown in Figure 4b, the error, $e(n)$, contains the signal component, $s(n)$, and the residual noise, $v_0(n) - v_1(n)$, after being processed by the counter-cancelation system. Despite the use of the cancelation system, if the residual noise is still considerable, it will cause the error, $e(n)$, to be more significant all the time, so that $\mu(x)$ cannot go to a minimal value and the adaptive algorithm is too complex to reach the optimal solution. It can only fluctuate around the optimal solution. At this time, the weight-vector step factor must be improved to eliminate the noise interference.

$$\mu(n) = \begin{cases} p, & n \leq N_0 \\ 2\alpha\left(\dfrac{1}{\frac{1}{\beta E[e(n)e(n-1)]^2}+1}\right) + \gamma, & n > N_0 \end{cases} \tag{9}$$

The DSLMS algorithm proposed in this paper employs an improved weight-vector step-size factor tuning function, shown in Equation (9), which determines the step size based on the current number of iterations, n, and a predefined threshold value, N_0. When the number of iterations n is less than or equal to N_0, the step size is set to a constant, p. This provides a more significant step size in the initial phase of the algorithm, which speeds up the initial convergence.

When the number of iterations (n) exceeds N_0, the step-size adjustment becomes a dynamic value dependent on the correlation of the error signal. Fine control of the step size is achieved by introducing the parameters α, β, and γ. α is a scaling factor that controls the overall amplitude, determining the baseline size of the step size change. β acts as a scaling factor, which controls how much the correlation of the error signal affects the step size. By adjusting this, we can control how quickly the algorithm responds to dynamic changes in the error signal, which involves the algorithm's adaptability and sensitivity. γ ensures that the algorithm has a minimum step size value, which means that the step size does not drop to zero even when the error correlation is minimal, guaranteeing that the algorithm can continue updating the weights.

In summary, the formula for the variable-step-size adaptive filtering algorithm in the adaptive pair cancelation filter proposed in this paper can be summarized as

$$e_j = d_j - y_j = s(n) + v_0(n) - w^T(n)x_j(n) \tag{10}$$

where e_j denotes the error signal, the difference between the desired signal and the filter output. d_j denotes the actual signal, the superposition of the desired output signal and the noise signal. y_j is the filter's actual output signal. $s(n)$ denotes the desired output signal. $v_0(n)$ is the noise signal. $v_1(n)$ represents the reference signal. $w^T(n)$ is a transposition of the filter weight vector.

$$w(n+1) = w(n) + 2\mu e_j x_j \tag{11}$$

where $w(n+1)$ denotes the weight vector at the next moment. $w(n)$ is the weight vector at the current moment. μ is the step factor, a positive constant that controls the speed of weight adjustment.

$$\mu(n) = \begin{cases} p, & n \leq N_0 \\ \beta\left(\dfrac{1}{\frac{1}{\alpha E[e(n)e(n-1)]^2}+1}\right) + \gamma, & n > N_0 \end{cases} \tag{12}$$

3.2. Algorithm Performance Analysis

To better analyze the specific impact of the parameters in the step adjustment function model on the algorithm's performance, this paper will focus on the volatility of the step coefficients as well as the role of parameters α and β on the convergence speed of the algorithm. On this basis, we will discuss in detail the selection principle of each parameter and its scope of application. When analyzing the step-size adjustment strategy, we will choose the specific condition of $\gamma = 0$ to explore the influence of α and β parameters.

Figure 5a shows the step coefficient, $\mu(n)$, and error (e_j) adjustment curve when α is different and β is the same; Figure 5b shows the step coefficient, $\mu(n)$, and error (e_j) adjustment curve when α is the same and β is different; and Figure 6 shows the fluctuation in algorithmic weight vectors when the parameters α and β are fixed and γ is taken to other values.

(**a**)

(**b**)
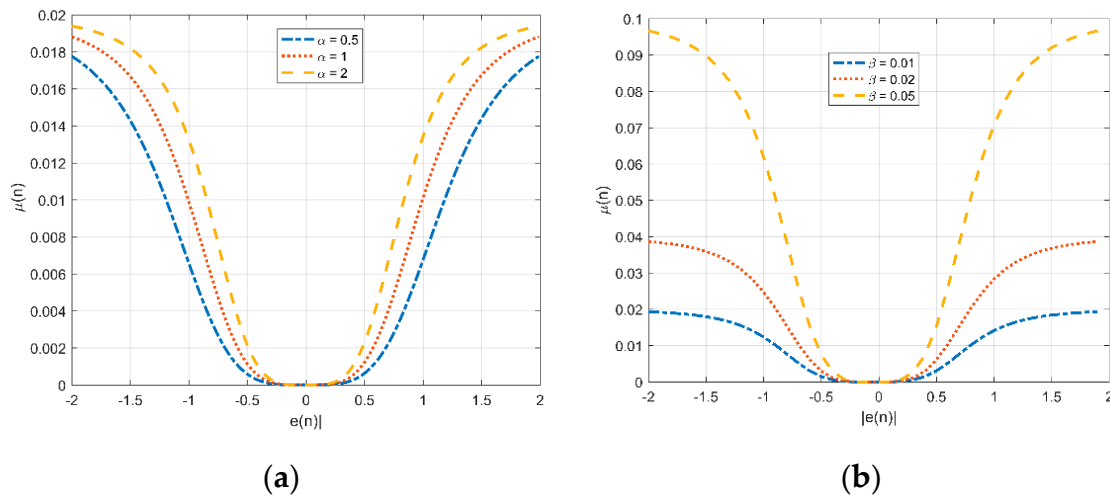
Figure 5. Effect of α and β on the convergence speed and stability of the algorithm: (**a**) β is the same and α is different; (**b**) α is the same and β is different.
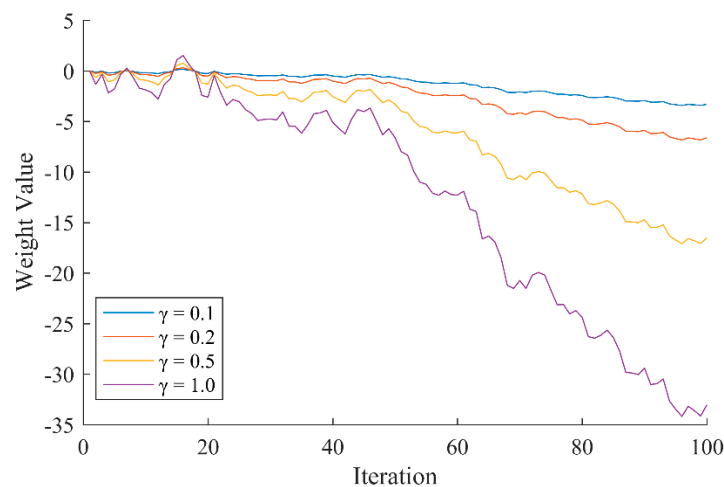

Figure 6. Fluctuation of weight vectors of the γ-different-time algorithm.

The three curves in Figure 5a are the corresponding step-size adjustment curves when α is taken as 0.5, 1, and 2 (β = 0.010), respectively, from which it can be seen that the larger the value of α, the larger the value of step size provided for the algorithm in the early stage of convergence, but the algorithm decreases in stability in the stage of convergence completion. The smaller the value of α, the more stable the algorithm is in the stage of convergence completion, but the algorithm's speed of convergence is slower; because of the equilibrium stability and the speed of convergence, α = 2 is taken in this paper. The three curves in Figure 5b are the corresponding step-size adjustment curves when β is 0.01, 0.02, and 0.05 (α = 2), respectively. It can be seen that the smaller the value of β, the smaller the change in the step-size factor in the error close to zero, but in the early stage of algorithm convergence, the algorithm is not able to provide a more significant value of step size and it slower; when β is taken with a more substantial value, it can provide the algorithm with a more significant value of step size to make it converge more slowly. When the value of β is more critical, it can give the algorithm a faster convergence speed. Still, at the same

time, the step factor will be affected by the error when the error is close to zero. The value of β should be in the range of 0.005~0.500; the value of β is taken as $\beta = 0.01$ in this paper, and the algorithm's performance in terms of convergence speed and steady-state error performance is better.

Figure 6 illustrates the effect of the parameter γ on the convergence performance of the algorithm's weight vectors. A higher value of γ accelerates the convergence of the weight vectors in the early stages of the algorithm. Still, it leads to more significant fluctuations in the weight vectors when the algorithm reaches the steady state. On the contrary, a lower value of γ makes the convergence speed slower but is conducive to reducing the fluctuation at the steady state and enhancing the algorithm's stability. Therefore, choosing the γ value requires a trade-off between convergence speed and strength, depending on the requirements of the application scenario. In this study, after comparative analysis and considering the convergence speed and weight fluctuation at steady state, we choose $\gamma = 0.2$ to ensure that the algorithm achieves good noise suppression while maintaining reasonable performance.

The DSLMS algorithm proposed in this paper uses time series correlation analysis to quantify the correlation of the error signal at two consecutive moments by calculating $E[e(n)e(n-1)]^2$. Utilizing the non-correlated nature of noise, the method can effectively distinguish between noise and valuable signals in the signal. The DSLMS algorithm can identify noise variations and reduce their effects when adjusting the step size, significantly enhancing the adaptive filter's anti-interference capability. In addition, the algorithm employs a dynamic step-size adjustment mechanism that connects the step size to the correlation of the error signal, allowing the algorithm to automatically reduce the step size as it approaches the optimal solution in response to the reduction in error. Combining the DSLMS algorithm with the pair eliminator system and fine-tuning the segmented step-size factor, the methodology used in this study effectively improves the noise rejection capability. It balances the convergence speed with the system stability while ensuring the algorithm's performance.

4. DSLMS Adaptive Filter

FPGAs (Field Programmable Logic Gate Arrays) excel in parallel processing power, fast data computation, on-demand reconfigurable flexibility, and strong customization adaptability, which make them an ideal hardware platform for implementing efficient and complex DSLMS algorithms [31,32].

The LMS algorithm requires many matrix and vector operations, and the FPGA's ability to perform multiple computational tasks, weighted updates, and error calculations in parallel ensures smooth processing of the data stream through the filter design, significantly accelerating the data processing rate [33]. In addition, the reconfigurability of FPGAs allows researchers to tailor the hardware logic to the LMS filter requirements, which not only helps to optimize the computational tasks but also allows the logic to be adjusted to the characteristics of the input data, which in turn improves the processing efficiency [34].

For the DSLMS algorithm proposed in this paper, FPGAs can achieve precise and rapid step-size adjustments, thus enhancing the stability and performance of the algorithm. Given these significant advantages of FPGAs, this study chooses to use them as the platform for DSLMS filter design. In this design, the filter is divided into four modules: a filter fir module, an error calculation (error_n) module, a step size update (mu_update) module, and a weight update (w_update) module, which realizes filter coefficient updating, error data updating, and filter calculation functions. With these four modules, the following five-step operation is realized:

(1) Filter calculations: $dy_j = w_j \times x_j (j = 0 \sim 11)$.

(2) Calculate the filter result and output $yout = \sum_{j=0}^{11} dy_j$.

(3) Calculation of errors: $e_j = d_j - yout$.

(4) Calculation of new weights: $dw_j = 2\mu x_j \times e_j$.

(5)   Updating weights: $w_j = w_{j-1} + dw_j$.

In the data filtering module, after receiving the signal pulse, the weighting of the input data is performed, and the results are truncated for output. Due to the feedback mechanism in the DSLMS algorithm optimized for noise correlation characteristics, the filtering process can lead to an overflow of computation results. Proper decimal point alignment will be ensured in the addition module to prevent computational errors due to overflow. In the face of integer bit-width mismatches, sign bit padding is performed on shorter-bit-width digits to match longer-bit-width digits; zero padding is performed on decimal bit-width mismatches to align them. In the module design, the saturation truncation strategy is implemented before the module outputs the results to ensure the accuracy of the calculation results, avoid data overflow, and retain as many valid bits as possible for the critical weight.

For the error calculation module, the module summarizes the filtering results from the previous stage and performs eight rounds of addition operations. After completing these operations, a saturation truncation process is applied, and the computed results are compared with the target signal to calculate the difference. Such a process aims to ensure computational accuracy while reducing distortion caused by numerical limits.

In the weight update module, a 16-bit multiplier with 16-bit multiplication calculates the weight update, producing a 32-bit result. This result then needs to be combined with the existing filter coefficients, which involves a truncation process to accommodate the bit width of the coefficients. To deal with this bit-width mismatch and avoid data overflow, a saturated truncation method is used in the system design, whereby when the multiplication result exceeds the maximum value that the target bit width can represent, the output is limited to the maximum value that can be defined, thus maintaining the validity of the result. This module adapts to signal variations and noise characteristics by precisely adjusting the filter coefficients, ensuring high output accuracy and stability.

Figure 7 represents a schematic diagram of the RTL structure of the DSLMS algorithm after generating the synthesized Verilog code, representing the implementation of Equations (10)–(12), respectively.
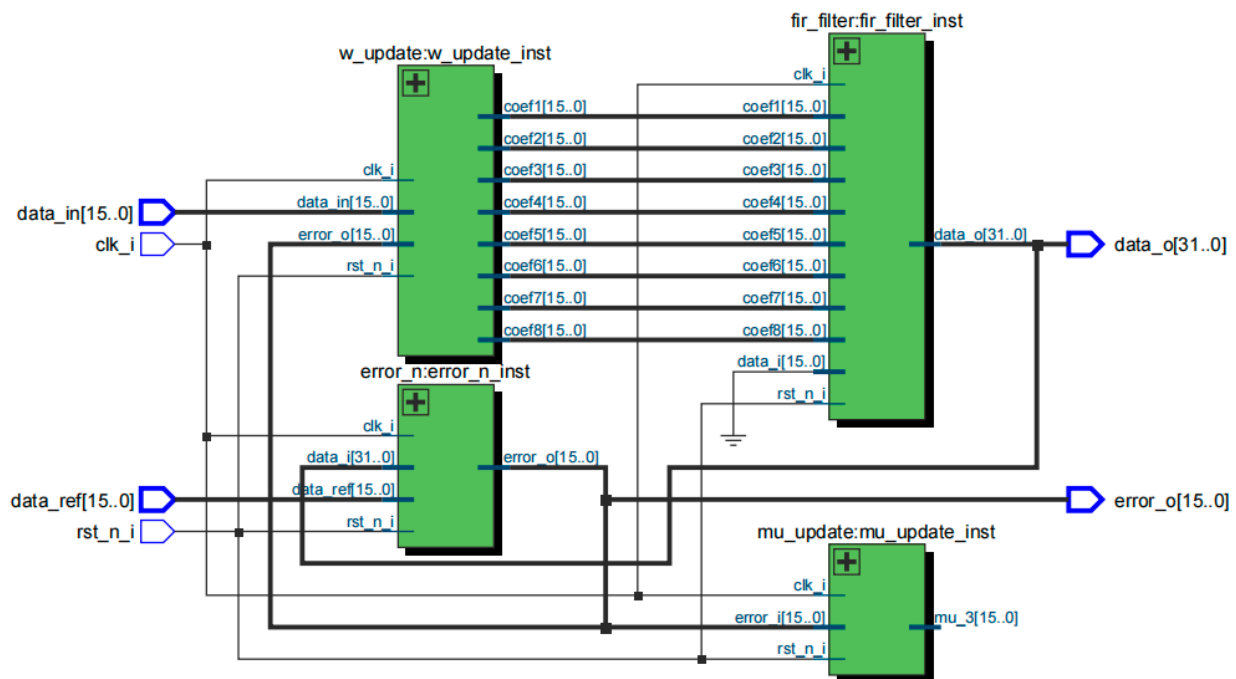

Figure 7. Schematic diagram of the filter RTL structure. fir_filter_inst is the filtering module, error_n_inst is the error calculation module, mu_update_inst is the step update module, and w_update_inst is the weight update module.

data_in[15:0] denotes the input signal, data_ref[15:0] denotes the desired signal, clk_i denotes the clock signal, rst_n_i denotes the reset signal (active, low), error_o[15:0] denotes the error signal, mu3 denotes the step factor mu of Equation (12), and coef1 to coef8 denote the weight update; coef1 denotes the initial weights, which are set to one. data_o[15:0] denotes the final filtered output.

Since the RTL view of each module is too cumbersome, the design schematic is used to show them individually. Figure 8 shows FilterError calculation module and Figure 9 shows Weight update module. Figure 10 shows Error calculation module. Figure 11 shows Step update module.
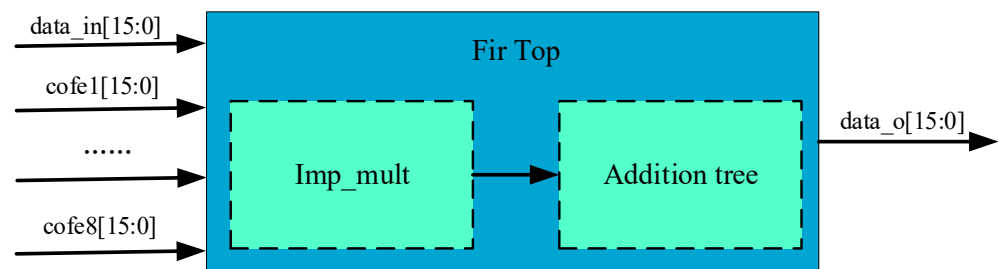
Figure 8. FilterError calculation module.
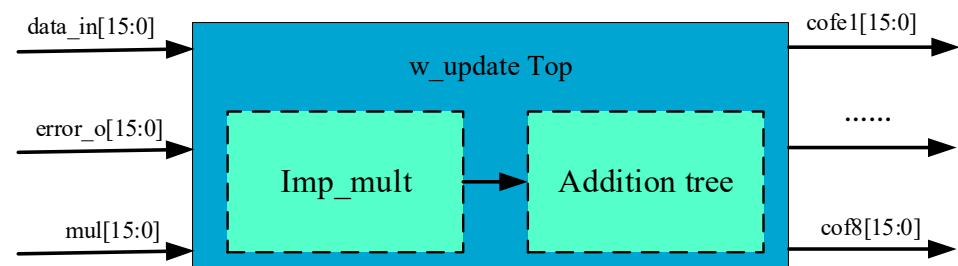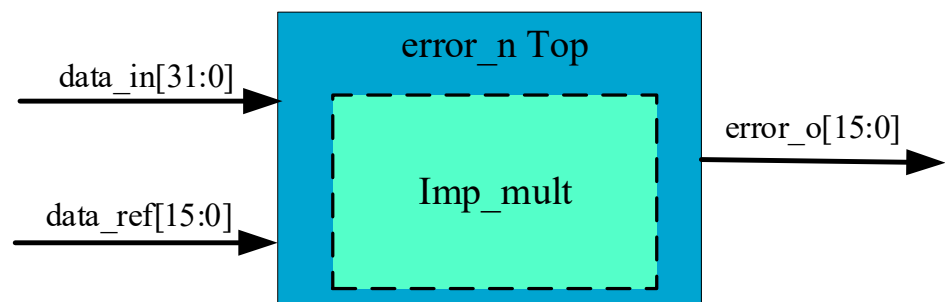
Figure 9. Weight update module.
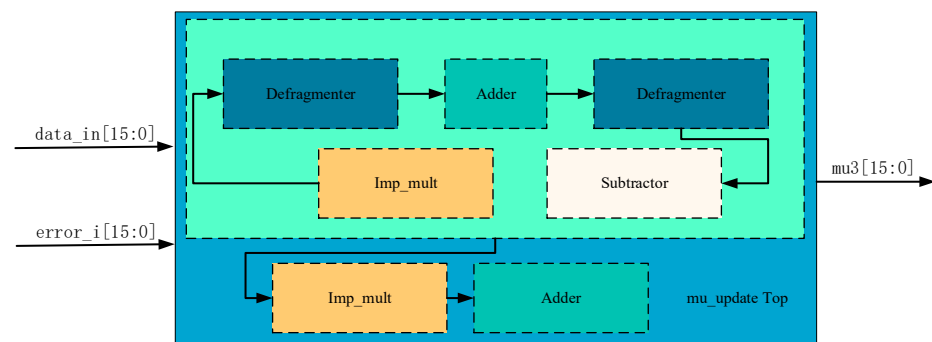
Figure 10. Error calculation module.

Figure 11. Step update module.

5. Experimental Verification

5.1. Experimental Environment

This paper uses the MATLAB 2018b software platform to carry out algorithm verification and simulation analysis. The specific conditions are set as follows: (1) the adaptive filter order is 8; (2) the initial weight coefficient of the filter is 0; (3) according to the assumptions, the parameters of the weak signal, $s(n)$, are set to be $kA = 0.1$, $f_0 + f_d = 50$ Hz; therefore, the weak signal, $s(n)$, can be expressed as

$$s(n) = 0.1cos(2\pi \times 50 \times n) \tag{13}$$

In Equation (13), 0.1 is the signal's amplitude, and 50 is the signal's frequency. To generate the noise, we use a built-in function in MATLAB to generate a Gaussian white noise $v_0(n)$ (n) with zero mean and a signal-to-noise ratio of 5 dB. The relationship between the signal and the noise is additive; i.e., the noise is added to the original signal. Thus, the final signal received by the system's receiving antenna is

$$d_j = s(n) + v_0(n) \tag{14}$$

where d_j is the signal at the original input of the filter, and $v_0(n)$ corresponds to additive Gaussian white noise with an adjustable signal-to-noise ratio. The original signal $s(n)$ is shown in Figure 12, the noise superimposed signal d_j is shown in Figure 13, and the noise signal $v_0(n)$ is shown in Figure 14.
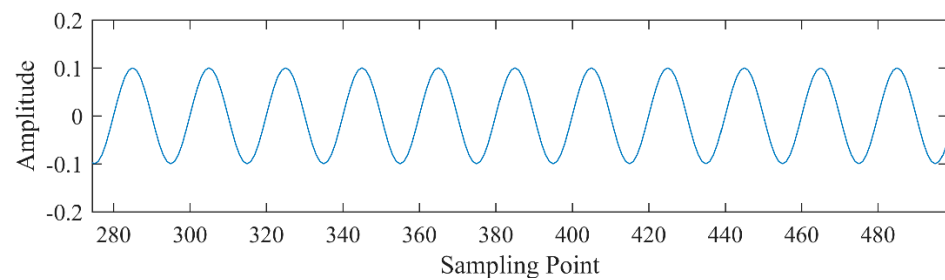

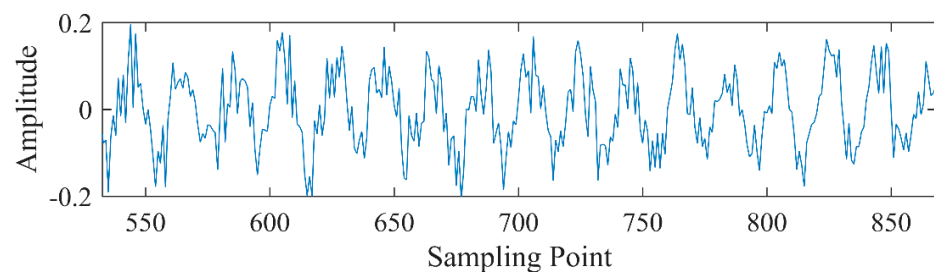**Figure 12.** Original signal, $s(n)$.

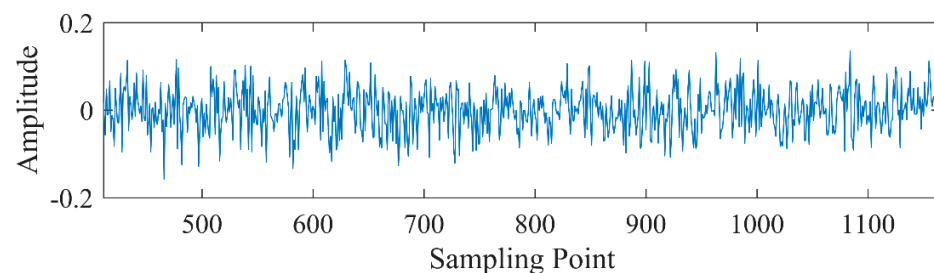**Figure 13.** Noise-superimposed signal, d_j.

Figure 14. Noise signal, $v_0(n)$.

5.2. DSLMS Simulation Parameter Selection and Experimental Analysis

The simulation experiments in this paper set the filter order to 8 orders and the number of sampling points to 3000 points. Three sets of simulations are conducted to compare the noise reduction effect of the fixed-step LMS algorithm, the variable-step LMS algorithm based on the Sigmoid function, and the DSLMS algorithm proposed in this paper.

Before the comparison experiment, each algorithm's optimal parameter selection values are determined. The value of μ of the fixed-step-size LMS algorithm is related to the maximum eigenvalue of the input signal correlation matrix, and based on the literature [35,36], the μ value of the fixed-step-size LMS algorithm is randomly adjusted within the theoretical range for 100 simulation experiments, and the effect of the step size on the error is observed. Analyzing the experimental results, it can be found that the μ value is too large or too small and can not converge stably or keep a small steady-state mean square error. The results of the simulation experiments are synthesized. Under the experimental conditions of this paper, the μ value of the fixed-step-size LMS algorithm should be around 0.15 for the best effect.

For the α and β values of the variable-step-size LMS algorithm based on the Sigmoid function, based on the literature research [37], 100 simulation experiments are carried out, respectively, with randomly selected values within the theoretical range, and the results are statistically analyzed to determine the appropriate parameter values. According to the comprehensive simulation experiment results, under the experimental conditions of this paper, the Sigmoid function-based variable-step-size LMS algorithm with an α value of 6 and a β value of 3 has the best effect.

Figure 15 shows the comparison of the mean square error (MSE) curves of the three algorithms, and Table 1 shows the comparison of the number of iterations of the algorithms under different signal-to-noise ratios, from which it can be seen that the traditional fixed-step LMS algorithm converges the slowest under four noise backgrounds (signal-to-noise ratios of 5, 10, 20 and 30, respectively). The DSLMS algorithm in this paper has the fastest convergence speed. The algorithm in this paper maintains a small mean square error and good stability in the convergence completion stage.
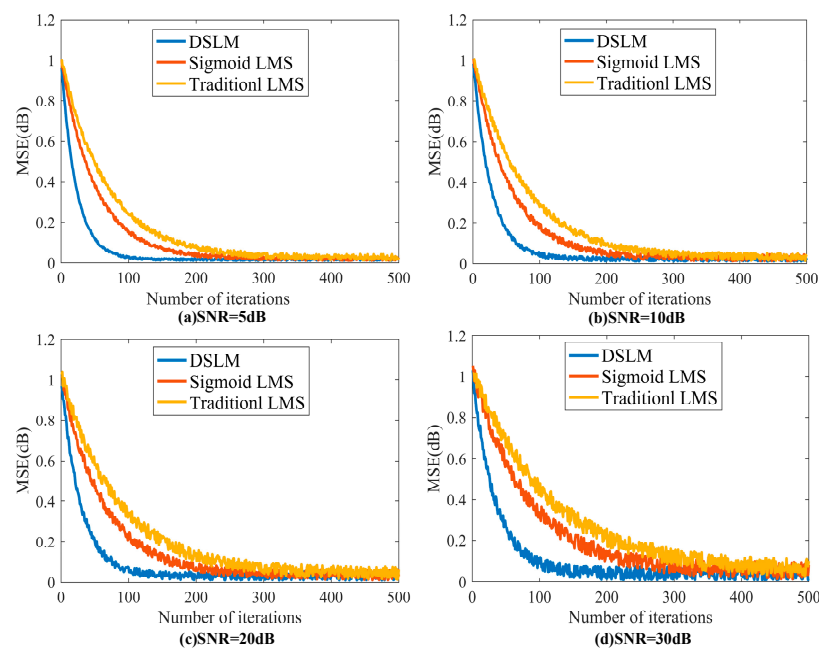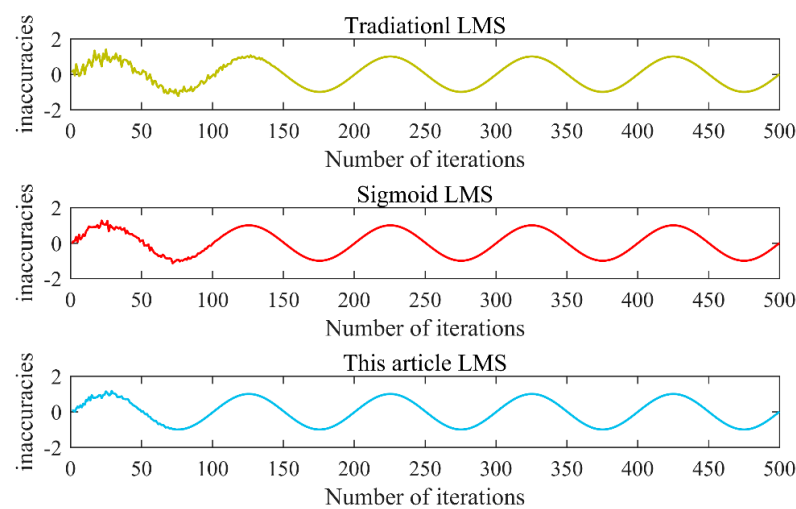

Figure 15. Comparison of MSE of the three algorithms in different SNR contexts. (**a**) SNR = 5 dB; (**b**) SNR = 10 dB; (**c**) SNR = 20 dB; (**d**) SNR = 30 dB.

Table 1. Comparison of the number of iterations of the algorithm with different signal-to-noise ratios.

MSE (dB)	Traditional LMS	Sigmoid Variable-Step-Size LMS	The Algorithms in This Paper
15	400	210	85
10	410	235	90
20	425	300	105
30	450	400	120

Figure 16 shows the comparison of the output signals of the traditional fixed-step LMS filter, the variable-step LMS filter with Sigmoid function, and the DSLMS filter in this paper.

Figure 16. Comparison of the filtered outputs of the three algorithms against the same SNR background.

Analyzing the above results, it can be found that the signal-to-noise ratios of the filtered signals are 8~16 dB higher than those of the pre-filtered signals, and the signal portion of the filtered signals that converge stably is 18~37 dB higher than that of the pre-filtered signals. The DSLMS algorithm proposed in this paper can maintain good convergence and stability under high and low signal-to-noise ratios and has a strong anti-noise interference capability when combined with the pair of eliminator systems.

To demonstrate further that the algorithm proposed in this paper performs better, we compared it with the traditional fixed-step LMS algorithm and the variable-step LMS algorithm with the Sigmoid function. We optimized the function equation and parameters that were set. Figure 17a,b shows the comparison between the ideal output signal and the filtered results of each algorithm under the conditions of a signal-to-noise ratio (SNR) of 1, 15, and 30, respectively. The figure shows that the SNR has a more significant effect on the traditional fixed-step LMS algorithm and the variable-step LMS algorithm with the Sigmoid function. In contrast, it has a lesser impact on this paper's DSLMS algorithm. Overall, the curves obtained by this paper's algorithm are closer to the ideal values, and the stability of the curves is reached more quickly. This result shows that the algorithm in this paper shows better results in reducing noise interference, improving convergence speed, and reducing steady-state error.
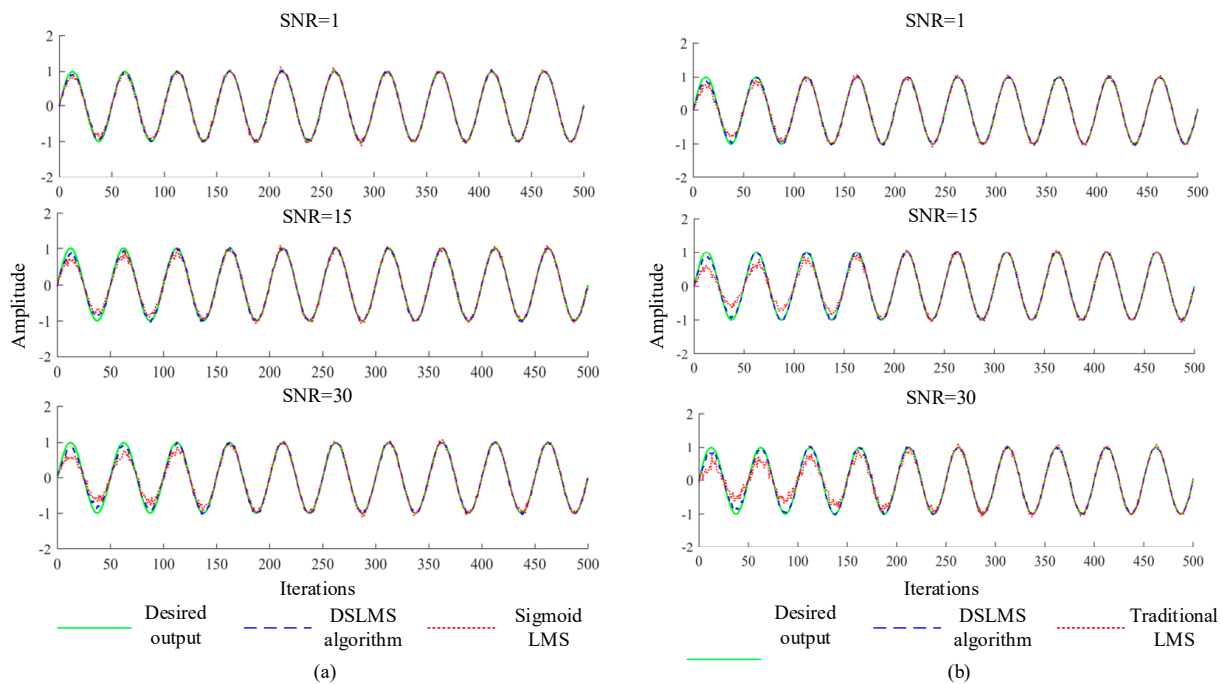
Figure 17. Comparison of traditional fixed-step LMS algorithm, variable-step LMS algorithm with Sigmoid function, and traditional fixed-step filtering with ideal output signal curve. (**a**) Comparison of the variable-step-size LMS algorithm for the Sigmoid function with the ideal output signal profile after filtering with the DSLMS algorithm. (**b**) Comparison of the ideal output signal curve after filtering between the conventional fixed-step LMS algorithm and the DSLMS algorithm.

5.3. FPGA System Test of DSLMS Filter

In order to verify the excellent filtering effect of the DSLMS filter in the FPGA system, we packaged the DSLMS filter into an IP core based on the one designed in Section 4 and completed the Block Design construction of the multi-channel signal acquisition board. Through this experimental platform, we not only realized the physical verification of the adaptive filter at the hardware level but also deeply analyzed its performance in real engineering applications.

Figure 18 shows the multi-channel signal acquisition board. The boards are designed and manufactured in-house to meet the specific needs of the experiments, which include the following main parts: (1) The role of the AD converter is to convert the analog signal generated by the signal generator into a digital signal. (2) The FPGA (Field Programmable Gate Array) is the core part of the experiment, which is used to implement the adaptive filtering algorithm. In the FPGA, the step factor can be adjusted according to the relevant characteristics of the noise to optimize the filtering effect. (3) The ethernet transmission module is used for the communication between the host computer and the FPGA to transmit the filtered data processed by the FPGA to the host computer for storage.

Figure 19 shows the test hardware platform. The platform includes not only a multi-channel signal acquisition board, but also a dedicated signal generator, which is purchased based on performance specifications and whose main function is to generate superimposed signals. The experiments need to simulate a variety of noise conditions under the transmission of the signal; the signal generator can generate a signal with a specific noise characteristic for the multi-channel signal acquisition card to provide input signals.

Finally, the bin file saved by the host computer was analyzed using Matlab data analysis software. In this way, the efficacy of the adaptive filter in removing noise can be evaluated, as well as its adaptability and performance in noisy environments. Preliminary estimates suggest that the uncertainty in the simulation results may be in the range of 3.5% to 5%, depending on the specific noise conditions of the signal and the settings of the filtering algorithms [38,39].
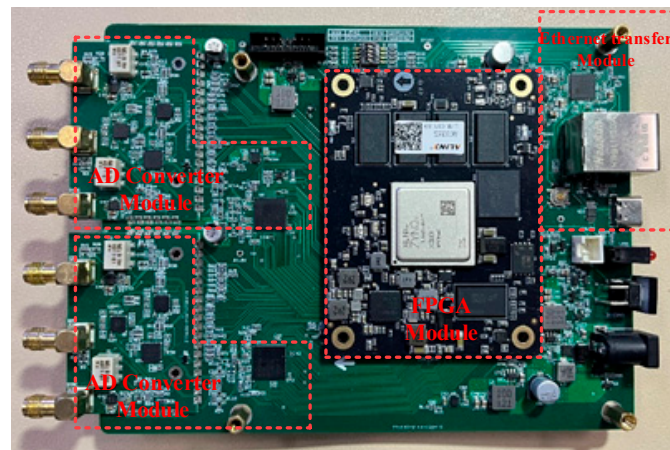
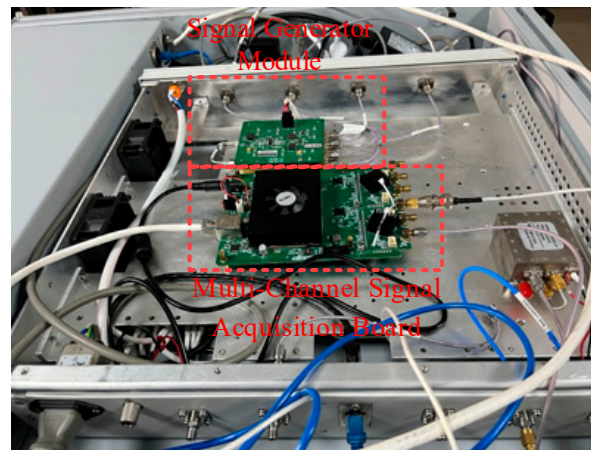**Figure 18.** Multi-channel signal acquisition board.

Figure 19. Test hardware platform.

Figure 20 shows the convergence of the DSLMS filter deployed in Matlab and the FPGA for data signal processing, from which it can be seen that the DSLMS algorithm filter implemented on the FPGA proposed in this paper is close to the simulation results of Matlab, which demonstrates the effectiveness of the FPGA hardware filter in signal denoising.
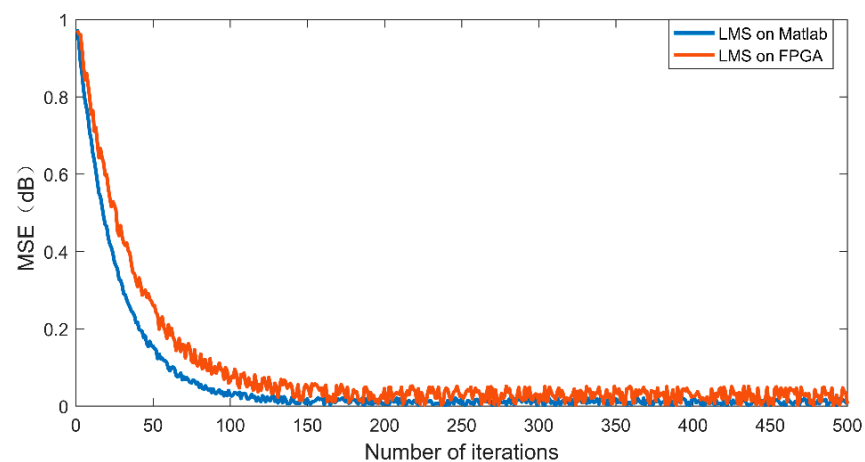

Figure 20. LMS algorithm performance comparison chart.

6. Conclusions

This study proposes an adaptive filter design scheme based on the DSLMS algorithm to improve the performance of weak signal processing in photoelectric detection technology. A new adaptive filtering algorithm with variable step size is proposed based on the minutiae function, combined with the pair elimination system to effectively reduce the noise component in error and improve the adaptive filter's anti-noise interference ability by utilizing the noise signal's low correlation. The adaptive filtering algorithm realizes an improvement in convergence speed and a significant reduction in steady-state error by adjusting the step factor of weight coefficients by segments. The experimental validation with the FPGA and Matlab shows that this scheme obtains better and more optimal results in noise suppression, accelerated algorithmic convergence, and steady-state error reduction compared to the traditional LMS filter, the variable-step-size LMS filter, and the traditional noise canceller. This scheme's effectiveness and application potential in the field of weak signal processing for photoelectric detection are confirmed.

Author Contributions: Conceptualization, Y.W. and M.W.; Methodology, Y.W., M.W. and W.B.; Software, Y.W. and M.W.; Validation, Y.W., M.W. and Z.S.; Formal analysis, Y.W. and W.B.; Investigation, Y.W., Z.S. and W.B.; Resources, Y.W. and W.B.; Data curation, Y.W. and Z.S.; Writing—original draft, Y.W. and W.B.; Writing—review & editing, Y.W., M.W., Z.S. and W.B.; Visualization, Y.W., M.W., Z.S. and W.B.; Supervision, Y.W. and M.W.; Project administration, Y.W.; Funding acquisition, M.W. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author. The data are not available to the public due to a confidentiality agreement with the company.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, B.; Zhang, X.; Jia, S.; Zou, S.; Tian, D. Adaptive time delay estimation based on signal preprocessing and fourth-order cumulant. *Circuits Syst. Signal Process.* **2023**, *42*, 6160–6181. [CrossRef]
2. Claser, R.; Nascimento, V.H. On the tracking performance of adaptive filters and their combinations. *IEEE Trans. Signal Process.* **2021**, *69*, 3104–3116. [CrossRef]
3. Guicking, D. On the invention of active noise control by paul lueg. *J. Acoust. Soc. Am.* **1990**, *87*, 2251–2254. [CrossRef]
4. Zhao, H.; Yu, Y.; Gao, S.; Zeng, X.; He, Z. A new normalized lmat algorithm and its performance analysis. *Signal Process.* **2014**, *105*, 399–409. [CrossRef]
5. Zhang, S.; Zhang, J. Fast stable normalised least-mean fourth algorithm. *Electron. Lett.* **2015**, *51*, 1276–1277. [CrossRef]
6. Dwivedi, S.; Aggarwal, P.; Jagannatham, A.K. Fast block lms and rlsbased parameter estimation and two-dimensional imaging in monostatic mimo radar systems with multiple mobile targets. *IEEE Trans. Signal Process.* **2018**, *66*, 1775–1790. [CrossRef]
7. Rao, T.K.; Srinivasan, M.; Lakshmaiah, D. Implementation of a low power and high speed adaptive noise canceller using lms algorithm. *Mater. Today Proc.* **2023**, *80*, 2055–2059.
8. Yu, C.; Gu, R.; Wang, Y. The application of improved variable step-size lms algorithm in sonar signal processing. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; Volume 1, pp. 1856–1860.
9. Jalal, B.; Yang, X.; Wu, X.; Long, T.; Sarkar, T.K. Efficient directionof-arrival estimation method based on variable-step-size lms algorithm. *IEEE Antennas Wirel. Propag. Lett.* **2019**, *18*, 1576–1580. [CrossRef]
10. Senapati, A.; Sharma, R.; Roy, J.S.; Singh, P. Beamforming in smart antenna with multiple interferers using leaky lms and variable step size leaky lms. In Proceedings of the 2018 International Conference on Applied Electromagnetics, Signal Processing and Communication (AESPC), Bhubaneswar, India, 22–24 October 2018; Volume 1, pp. 1–4.
11. Chergui, L.; Bouguezel, S. Variable step size pre-whitening transform domain lms adaptive noise canceller. In Proceedings of the 2019 International Conference on Advanced Systems and Emergent Technologies (IC ASET), Hammamet, Tunisia, 19–22 March 2019; pp. 327–331.

12. Wu, L.; Nie, Y.; Zhang, Y.; He, S.; Zhao, Y. Adaptive filtering denoising method based on variational mode decomposition. *J. Electron. Sci.* **2021**, *49*, 1457–1465.

13. Tobar, F.A.; Kuh, A.; Mandic, D.P. A novel augmented complex valued kernel lms. In Proceedings of the 2012 IEEE 7th Sensor Array and Multichannel Signal Processing Workshop (SAM), Hoboken, NJ, USA, 17–20 June 2012; pp. 473–476.

14. Gao, Y.; Zhao, H.; Zhu, Y.; Lou, J. The q-gradient lms spline adaptive filtering algorithm and its variable step-size variant. *Inf. Sci.* **2024**, *658*, 119983. [CrossRef]

15. Zhang, Y.; Yang, M.; Liu, X.; Zhang, Z.-z. Improved least mean square algorithm based in satellite multi-beamforming. *J. Commun.* **2017**, *38*, 171–178.

16. Lu, B.; Feng, C.; Long, G. A new variable step-size lms adaptive algorithm based on marr function. In Proceedings of the 2013 International Conference on Information Technology and Applications, Chengdu, China, 16–17 November 2013; pp. 214–217. [CrossRef]

17. He, D.; Wang, M.; Han, Y.; Hui, S. Variable step size lms adaptive algorithm based on exponential function. In Proceedings of the 2019 IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP), Weihai, China, 28–30 September 2019; pp. 473–477. [CrossRef]

18. Ding, F. Complexity, convergence and computational efficiency for system identification algorithms. *Control Decis.* **2016**, *31*, 1729–1741.

19. Ji, Z. Comparison and simulation of system identification algorithms based on correlation analysis. *Comput. Knowl. Technol.* **2016**, *12*, 253–254.

20. Arenas-Garcia, J.; Azpicueta-Ruiz, L.A.; Silva, M.T.; Nascimento, V.H.; Sayed, A.H. Combinations of adaptive filters: Performance and convergence properties. *IEEE Signal Process. Mag.* **2015**, *33*, 120–140. [CrossRef]

21. Bhotto, M.Z.A.; Antoniou, A. Affine-projection-like adaptive-filtering algorithms using gradient-based step size. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2014**, *61*, 2048–2056. [CrossRef]

22. Kirubarajan, T.; Bar-Shalom, Y. Low observable target motion analysis using amplitude information. *IEEE Trans. Aerosp. Electron. Syst.* **1996**, *32*, 1367–1384. [CrossRef]

23. Yu, T.; Li, W.; Yu, Y.; de Lamare, R.C. Robust spline adaptive filtering based on accelerated gradient learning: Design and performance analysis. *Signal Process.* **2021**, *183*, 107965. [CrossRef]

24. Gelfand, S.B.; Wei, Y.; Krogmeier, J.V. The stability of variable step-size lms algorithms. *IEEE Trans. Signal Process.* **1999**, *47*, 3277–3288. [CrossRef] [PubMed]

25. Lim, J.S. *Two-Dimensional Signal and Image Processing*; Prentice-Hall, Inc.: Hoboken, NJ, USA, 1990.

26. Zhang, X.; Yang, S.; Liu, Y.; Zhao, W. Improved variable step size least mean square algorithm for pipeline noise. *Sci. Program.* **2022**, *2022*, 3294674. [CrossRef]

27. Ahmad, N.A. A Globally Convergent Stochastic Pairwise Conjugate Gradient-Based Algorithm for Adaptive Filtering. *IEEE Signal Process. Lett.* **2008**, *15*, 914–917. [CrossRef]

28. Vettori, S.; Di Lorenzo, E.; Peeters, B.; Luczak, M.; Chatzi, E. An adaptive-noise Augmented Kalman Filter approach for input-state estimation in structural dynamics. *Mech. Syst. Signal Process.* **2023**, *184*, 109654. [CrossRef]

29. Gorriz, J.M.; Ramirez, J.; Cruces-Alvarez, S.; Puntonet, C.G.; Lang, E.W.; Erdogmus, D. A novel lms algorithm applied to adaptive noise cancellation. *IEEE Signal Process. Lett.* **2008**, *16*, 34–37. [CrossRef]

30. Ploder, O.; Lang, O.; Paireder, T.; Motz, C.; Huemer, M. A new class of self-normalising lms algorithms. *Electron. Lett.* **2022**, *58*, 492–494. [CrossRef]

31. Suma, G. Fpga based high speed data acquisition system with ethernet interface. *Int. J. Adv. Sci. Eng. Technol.* **2014**, *2*, 2321–9009.

32. Wang, X.; Zeng, T.; Yin, S.; Wang, Y.; Hu, Z. Design and implementation of automatic acquisition and processing system for balance based on arm and fpga. In Proceedings of the 2018 5th International Conference on Systems and Informatics (ICSAI), Nanjing, China, 10–12 November 2018; pp. 19–23.

33. Rai, A.; Roy, A.; Qamar, S.; Saif, A.G.F.; Hamoda, M.M.; Azeem, A.; Mohammed, S.A. Modeling and simulation of fir filter using distributed arithmetic algorithm on fpga. *Multimed. Tools Appl.* **2024**, 1–14. [CrossRef]

34. Zhang, Y.; Zhang, L.; Wu, Z.; Su, Y.; Yan, F. Design and fpga implementation of an adaptive narrowband interference suppression filter. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 8002215. [CrossRef]

35. Lakshmaiah, G.S.; Narayanappa, C.K.; Shrinivasan, L.; Narasimhaiah, D.M. Efficient very large-scale integration architecture design of proportionate-type least mean square adaptive filters. *Int. J. Reconfigurable Embed. Syst.* **2024**, *13*, 69–75. [CrossRef]

36. Hamzah, A.E.; Zan, M.S.D.; Hamzah, M.E.; Fadhel, M.M.; Sapiee, N.M.; Bakar, A.A.A. Fast and accurate measurement in botda fibre sensor through the application of filtering techniques in frequency and time domains. *IEEE Sens. J.* **2024**, *24*, 4531–4541. [CrossRef]

37. Mao, J.; Wang, Z.; Liu, J.; Song, D. A forward- backward splitting equivalent source method based on s- difference. *Appl. Sci.* **2024**, *14*, 1086. [CrossRef]

38. Salzenstein, P.; Pavlyuchenko, E. Uncertainty Evaluation on a 10.52 GHz (5 dBm) Optoelectronic Oscillator Phase Noise Performance. *Micromachines* **2021**, *12*, 474. [CrossRef]
39. Meng, X.; Yuan, H.; Wang, Y. Research on the construction method of photoelectric detection preamplifier circuit combined with single chip microcomputer technology. In Proceedings of the 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Dalian, China, 27–28 August 2021; pp. 516–520. [CrossRef]