

Magnitude-based pulse width estimation via efficient edge detection

Kenneth Ranney^{✉,*}, Kwok Tom, Dannielle Tadas, Neal Tesny, Andre Magill, and William Diehl

DEVCOM U.S. Army Research Laboratory, Adelphi, Maryland, United States

Abstract. In recent years, researchers have addressed the problem of using noncoherent approaches to estimate pulse width and pulse repetition interval. Since the measured transmitter is noncooperative, and noncoherent integration gain can be realized, the input signal-to-noise ratio (SNR) for these estimators becomes critical. We examine multiple edge detectors that exploit moving sums calculated as part of a Haar filtering of the received signal magnitudes. Two different ratio tests are considered in addition to the Haar filtering (or “difference of boxes”) approach, and a binary hypothesis test is designed based on a “smallest of” constant false alarm rate formulation. Probability arguments are then invoked to derive readily evaluated expressions for the detection thresholds. Tests are conducted, indicating that performance of the ratio-based approaches is comparable in terms of processed peak-to-background ratio. However, comparisons of root mean-squared (RMS) error indicate that the difference-based (Haar) approach produces lower error than both ratio-based approaches. The Haar filter approach is further demonstrated to remain effective (100% detection, 0% false alarm, RMS estimation errors of <3%) at low SNRs of ~0 dB. © The Authors. Published by SPIE under a Creative Commons Attribution 4.0 International License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JRS.16.016509](https://doi.org/10.1117/1.JRS.16.016509)]

Keywords: pulse detection; pulse width estimation; radar pulse; edge detection; constant false alarm rate.

Paper 210633 received Oct. 5, 2021; accepted for publication Jan. 5, 2022; published online Feb. 16, 2022.

1 Introduction

Recent research has produced a variety of techniques to characterize measured radar pulses by separating them into subsets—each potentially transmitted by a different radar system. The first steps in this process include a pulse width (PW) estimator, and many effective component systems have been proposed.^{1–8} Unfortunately, these approaches often maximize performance at the expense of operational efficiency, where we refer to an implementation as “efficient” if (1) it attempts to minimize the number of required arithmetic operations, and (2) it prioritizes addition over multiplication and multiplication over division. This designation arises due to the relative ease of implementing addition operations as part of a hardware design. One approach of particular interest is based on a variant of Haar wavelet filtering and can be efficiently implemented using moving average windows.^{1,2} To improve performance, however, the algorithm first processes the signal in the frequency domain before calculating the Haar filter output in the time domain. Hence, this approach accepts additional latency in the output data sequence to achieve a higher fidelity signal. Other PW estimation strategies that leverage averaging (integration) and/or thresholding techniques of some sort have also been reported.^{3–6} These approaches frequently detect candidate pulse edges and then determine whether these detections represent leading edges, trailing edges, or samples that fall within a pulse. Some require two thresholds—a higher threshold to indicate a transition from noise to pulse, and a lower threshold to indicate a transition from pulse to noise. If two thresholds are *not* employed, then a clustering algorithm can group (cluster) closely spaced detections to form a final PW estimate from individual samples that

*Address all correspondence to Kenneth Ranney, kenneth.i.ranney.civ@army.mil

exceed the detection threshold. The output of the PW estimator then serves as input to a pulse repetition interval (PRI) estimator and further downstream processing.

One particularly appealing implementation leverages the Haar filter (or “difference of boxes”) to increase the accuracy of edge detection estimates without the need for additional frequency domain preprocessing.^{7,8} Such an approach lends itself to an extremely efficient implementation in which the output of each sum is updated with one addition and one subtraction. It also represents a natural extension of edge detection concepts commonly found in image processing.^{9,10} Another appealing implementation can be derived by considering the statistical techniques used to detect edges in synthetic aperture radar imagery.¹¹ These approaches mirror, to some extent, the classic constant false alarm rate (CFAR) approaches that act as statistical anomaly detectors,^{12,13} operating on ratio statistics rather than differences. Ultimately, however, the problem can be framed in terms of a binary hypothesis test,¹⁴ where H_0 represents the hypothesis that no edge is present.

While many authors present effective implementations of the proposed algorithms, some do not take full advantage of the special form of convolution with a rectangular window (i.e., the moving sum).^{1,2} The normalization at each step (to determine the average in some implementations) also introduces an expensive division, and the calculation of the square root introduces additional computational overhead. Finally, the mean and standard deviations of both the Rayleigh and exponential distributions include a single, common parameter; hence, a threshold defined by $T = \mu + k_1\sigma$ (for mean μ , standard deviation σ , and scale factor k_1) can also be expressed as $T = k_2\mu$ (for a different scale factor k_2). This property is overlooked in currently documented approaches; even when a Rayleigh distribution assumption is incorporated.⁷

In what follows, we consider three different test statistics based on variants of the Haar filter paradigm, all designed to minimize latency by reducing computational complexity. The proposed moving sum filter designs reduce the number of operations required by (1) omitting divisions (normalization) that involve constants whose reciprocals can be precalculated and (2) efficiently updating the leading and lagging sums. Here, we view the two halves of the Haar filter as separate moving sums. In addition, the filter inputs, themselves, comprise *squared* magnitudes, eliminating square root calculations. Since the inputs to the pulse estimators are in-phase and quadrature (I/Q) outputs from the (coherent) receiver, calculating the complex magnitude requires *both* squaring the I/Q samples and taking the square root of their sum.

The use of squared magnitudes also engenders convenient relationships between the input and output probability density functions (pdfs) under certain (reasonable) assumptions (i.e., exponentially distributed between-pulse samples). These pdf relationships are particularly appealing when the ratio of the leading and lagging sums (rather than their difference) constitutes the test statistic. Although the ratio calculation replaces the difference operation with a division operation, we still consider this formulation for the sake of completeness. We also document certain properties regarding threshold determination that are unique to the ratio approaches. The major contributions of this paper are (1) development of ratio test statistics and probability-based threshold calculations for both the Haar filter and ratio methods and (2) quantification of performance for the Haar filter and ratio methods.

This paper is organized as follows: Section 2 begins the discussion with definitions of the test statistics for all methods—the difference- and ratio-based approaches. Plots of Haar filter and ratio-based test statistics qualitatively illustrate the performance of the two methods. Section 2 then proceeds with a description of the underlying pdf assumptions, noting how these assumptions influence the calculation of detection thresholds for each method. In particular, various techniques are considered for calculating the detection thresholds based on assumptions about the probability distribution of between-pulse data samples. Section 2 concludes with the introduction of a modified version of the ratio statistic. Both a detailed examination of the threshold calculation procedures and a qualitative comparison with the difference-based approach are presented. A block diagram of the entire system (with Haar filtering) is shown in Fig. 1.

Section 3 includes a description of two additional procedures required as part of the PW estimation process along with some issues related to the determination of their key parameters. These procedures comprise a local maximum (local max) detector and an edge associator, and together they represent important components of the PW estimation system. The local max detector retains only the largest sample within a specified radius of the sample under test

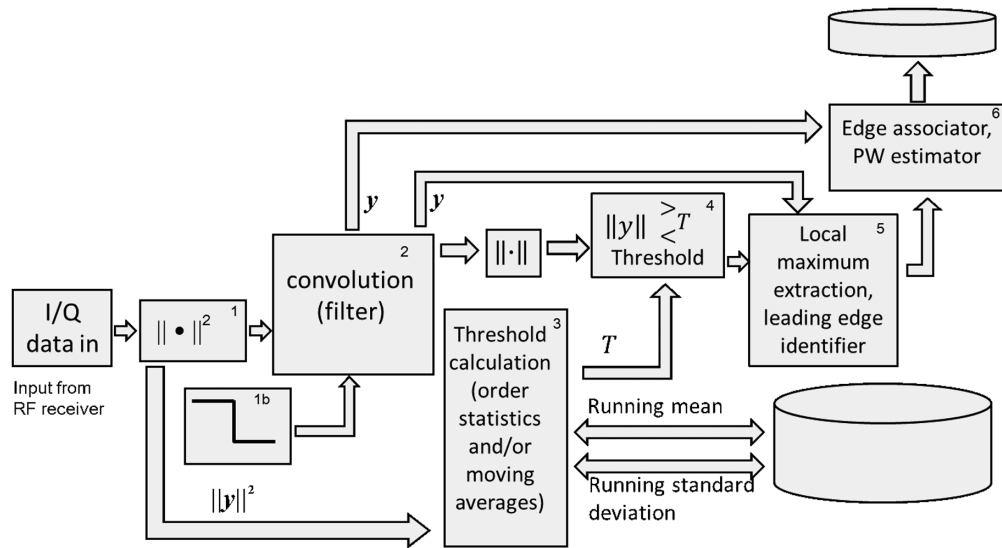


Fig. 1 Block diagram of the PW and PRI estimation system with the Haar filter PW estimator.

(SUT), and the edge associator assigns a detected leading (rising) edge to each detected trailing (falling) edge. Section 4 documents experiments conducted using signal-generation hardware at the DEVCOM Army Research Laboratory, quantifying the effectiveness of all three methods as well as the similarities between their performances. Finally, Sec. 5 summarizes all results, highlighting the most significant issues.

We emphasize here that we did not consider coherent additional techniques based on time-frequency analysis (e.g., Refs. 15–17), since they were beyond the scope of this research.

2 Pulse Detection and PW Estimation Using the Haar Filter or Moving Sum Ratio

The Haar filter pulse detection and PW estimation procedure begin with the convolution of the signal magnitude and the Haar wavelet, as shown in blocks 1, 1b, and 2 of Fig. 1. This operation is also defined by

$$\begin{aligned}
 \hat{y}_{\text{out,Haar}}(n) &= \sum_{i=n+1}^{n+N/2} \|x(i)\|^2 - \sum_{i=n-N/2+1}^n \|x(i)\|^2 \\
 &= \hat{y}_{\text{out,Haar}}(n-1) + \|x(n+N/2)\|^2 - 2\|x(n)\|^2 + \|x(n-N/2)\|^2 \\
 y_{\text{out,Haar}} &= 2/N \hat{y}_{\text{out,Haar}}(n),
 \end{aligned} \tag{1}$$

where $x(n)$ denotes the input (complex), baseband data sequence, N represents the Haar filter length, and $\|x(n)\|$ denotes the complex magnitude of $x(n)$. We note that the output sample lags the input sample by $N/2$ samples, and the new output at time n can be determined from the current output at time $(n-1)$ in three steps. First, the input sample at time $n+N/2$ is added to the current output. Next, the input sample from time n is subtracted twice from the current output. This operation converts the sample farthest from the leading edge of the positive portion of the Haar summation into the sample closest to the leading edge of the negative portion of the Haar summation. That is, the sample moves from the positive box of the Haar filter to the negative box of the Haar filter. Finally, the input sample at time $n-N/2$ is added to the current output. This removes the sample farthest from leading of the negative box of the Haar filter, yielding the new contribution for that portion of the Haar filter. To simplify subsequent analyses, we have included the scale factor that converts the sum to an average. In a general implementation, however, this scale factor would not be required.

The second, less efficient PW estimator is based on a ratio test and is defined by

$$\begin{aligned}
 y_{\text{out,ratio}}(n) &= \sum_{i=n+1}^{n+N/2} \|x(i)\|^2 / \sum_{i=n-N/2+1}^n \|x(i)\|^2 = s_{\text{num}}(n)/s_{\text{den}}(n) \\
 &= [s_{\text{num}}(n-1) + \|x(n+N/2)\|^2 - \|x(n)\|^2] / [s_{\text{den}}(n-1) + \|x(n)\|^2 - \|x(n-N/2)\|^2].
 \end{aligned}
 \tag{2}$$

We immediately recognize the similarity between Eqs. (1) and (2) described in the introduction—the moving averages constituting the difference in Eq. (1) are the moving averages constituting the ratio in Eq. (2). In this case, block 2 in Fig. 1 is replaced by a ratio of the moving averages. Note that the ratio test as described in Eq. (2) represents the leading sum divided by the trailing sum; hence, for a rectangular pulse $y_{\text{out,ratio}}(n)$ will be largest when the center of the filter is located at the rising edge. Similarly, it will be smallest when the center of the filter is located at the falling edge. Let $y_{\text{out,ratio}}(m)$ represents an isolated noise sample, and assume that all I/Q noise samples contributing to the sums in Eq. (2) are independent, identically distributed (i.i.d.) Gaussian random variables. Invoking independence and symmetry arguments for the case where only noise samples are present, we recognize that the pdfs for $y_{\text{out,ratio}}(n)$ and $1/y_{\text{out,ratio}}(n)$ are the same. This implies that

$$\begin{aligned}
 \Pr\{\text{leading edge}\} &= \Pr\{y_{\text{out,ratio}}(n) > T\} \\
 &= \Pr\{1/y_{\text{out,ratio}}(n) < 1/T\} \\
 &= \Pr\{y_{\text{out,ratio}}(n) < 1/T\} \\
 &= \Pr\{\text{trailing edge}\}
 \end{aligned}
 \tag{3}$$

and the trailing edge threshold can be determined from the leading edge threshold for a CFAR-based approach.

Both approaches introduce a latency of $N/2$ samples in the output, and the ratio requires one division followed by a second threshold calculation. Figure 2 shows the magnitude of the input I/Q data (i.e., the output of block 1 in Fig. 1), whereas Fig. 3 shows the difference between the output of the Haar filter and output of the ratio for $r = (\text{rising average})/(\text{falling average})$. These output plots illustrate why a second threshold is required if a single ratio calculation is performed. We determine these thresholds based on the moving averages represented by block 3 in Fig. 1. The signal-to-noise ratio (SNR) for this dataset has been set to 3 dB as part of the dataset creation process.

2.1 Determination of Thresholds for the Haar Filter

In addition to its computational simplicity, the Haar filter approach also lends itself to an effective threshold calculation based on slightly modified, classic CFAR concepts.^{12,14} In particular,

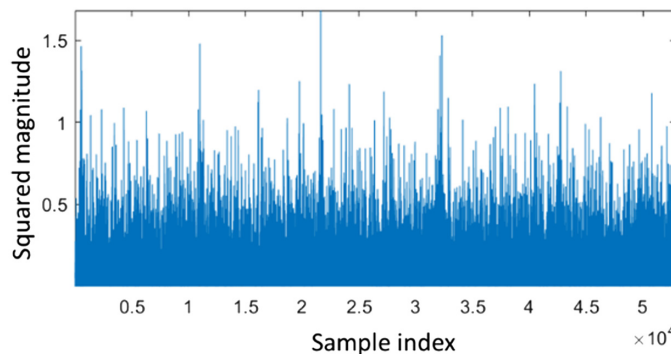


Fig. 2 Squared magnitude of input, baseband data, SNR = 3 dB.

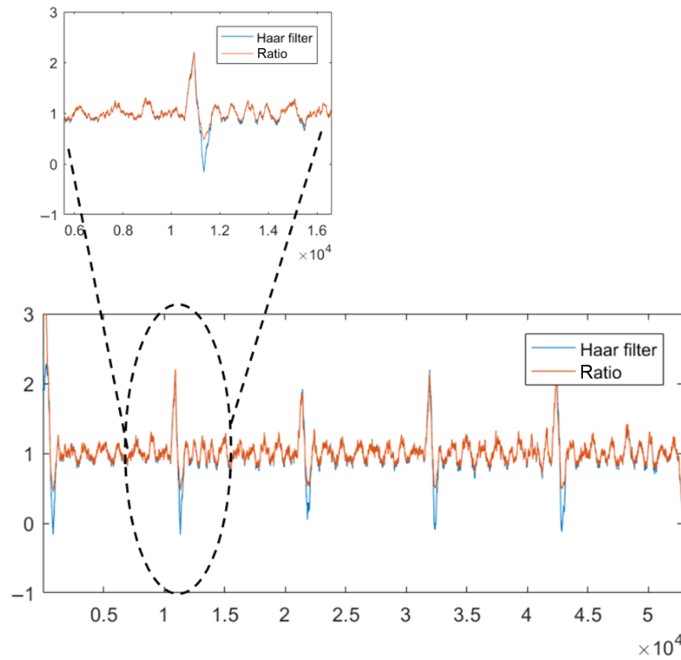


Fig. 3 Scaled and translated output of Haar filter (blue) and unscaled output of ratio (red). Zoom of the first pulse illustrates the nature of the trailing edge ratio estimate.

if we assume the I/Q samples *between pulses* are i.i.d. and normally distributed random variables with mean zero and variance σ^2 , then $X = (\frac{I}{\sigma})^2 + (\frac{Q}{\sigma})^2$ follows a chi-squared distribution with two degrees of freedom¹⁸ with pdf

$$f(x) = \frac{1}{2} e^{-x/2}.$$

From this, we have that $Y = \sigma^2 X = I^2 + Q^2$ follows the exponential pdf

$$f(y) = \frac{1}{2\sigma^2} e^{-y/2\sigma^2}. \quad (4a)$$

The sum of $N/2$ i.i.d. random variables with pdf $f(y)$ follows a $\text{gamma}(2\sigma^2, N/2) = \text{gamma}(\beta, \alpha)$ distribution with pdf¹⁹

$$f(x) = \frac{(x/\beta)^{\alpha-1} e^{-x/\beta}}{\beta\Gamma(\alpha)}, \quad (4b)$$

where β is the *scale parameter*, α is the *shape parameter*, and $\Gamma(\cdot)$ is the gamma function. Note that the order of the scale and shape parameters within the expression $\text{gamma}(\alpha, \beta)$ has been observed to vary from one author to the next.^{19,20}

Hence, each Haar filter output sample can be viewed as the difference of two i.i.d. gamma random variables with a moment generating function of the form²¹

$$M_{\text{difference}}(t) = (1 - \beta t)^{-\alpha} (1 + \beta t)^{-\alpha}, \quad (5)$$

where $\alpha = N/2$ and $\beta = 2\sigma^2$. While the pdf corresponding to $M_{\text{difference}}(t)$ is symmetric about zero, it is difficult to work with; so, we proceed by deriving a threshold in terms of the mean and standard deviation of the Haar filter output.

Based on our assumptions about the I/Q input samples, the difference of moving averages that constitutes the Haar filter output is just the difference of two i.i.d. gamma random variables. Assume that both Y_1 and Y_2 are i.i.d. $\text{gamma}(2\sigma^2, N/2)$ random variables, and denote the expected value (mean) of Y_1 as $E\{Y_1\}$; then $E\{Y_1\} = E\{Y_2\} = \beta\alpha = 2\sigma^2 N/2$.²² Similarly, the

variance of Y_1 , $\text{var}\{Y_1\} = \text{var}\{Y_2\} = \beta^2\alpha = (2\sigma^2)^2(N/2)$, and it follows that for $Z = Y_1 - Y_2$, $E\{Z\} = 0$ and $\text{var}\{Z\} = \text{var}\{Y_1\} + \text{var}\{Y_2\} = (2\sigma^2)^2N$.²² Let $\hat{\mu}$ be the estimate of the mean of the original (input) data sequence ($\hat{\mu} = 2\sigma^2$), then the estimator, $\hat{\sigma}_{\text{difference}}$, becomes

$$\hat{\sigma}_{\text{difference}} = \sqrt{(2\sigma^2)^2N/N^2} = \hat{\mu}/\sqrt{N}, \tag{6}$$

where N is the length of the Haar filter, and the estimate $\hat{\mu}$ has been obtained in an operation separate from the Haar filtering operation. In particular, $\hat{\mu}$ is determined by the moving average calculator shown in box 3 of Fig. 1, and it constitutes the background estimate for a CFAR implementation.

We now have the quantities necessary to formulate a classic CFAR threshold for the Haar filter output in terms of the background mean and standard deviation, expressed as $T = \mu \pm k\sigma$. Since the mean of the output is zero, the threshold becomes

$$T_{\text{Haar}} = \pm k\hat{\sigma}_{\text{difference}}, \tag{7}$$

for some scale factor k . This approach has the added advantage that it requires only moving sums of the input data, and these can be determined using a computationally efficient algorithm. No additional calculation involving the Haar filter outputs, such as means and standard deviations, is required.

To determine an acceptable estimate of the noise background, we leverage a variant of the cell-averaging (CA) CFAR referred to as the smallest of (SO) CA CFAR, or simply SO-CFAR.^{12,23} The locations of samples within the data stream used to calculate the CFAR statistics are shown in Fig. 4, where the input sequence, $\|x(i)\|^2$, is indicated at the left of the figure. In this CFAR configuration, two background estimates are calculated—one on either side of the SUT—and the smallest value is retained as the final estimate. The operation in Fig. 4 can be expressed mathematically as

$$\hat{\mu} = \min\left(\frac{1}{M} \sum_{n \in \text{region 1}} \|x(n)\|^2, \frac{1}{M} \sum_{n \in \text{region 2}} \|x(n)\|^2\right), \tag{8}$$

where $\min(a, b)$ is the minimum of a and b . We have selected this method in an effort to reduce masking effects that occur when smaller-magnitude pulses appear near larger-magnitude ones. Other authors also deal with such effects,^{13,24} but their approaches are too computationally expensive for our envisioned applications. The background averages from the regions shown in Fig. 4 can be calculated efficiently using the technique described by Eq. (1), with the M samples from each background estimator replacing the $N/2$ samples from each half of the Haar filter. This formulation allows for additional flexibility in determining the background regions; however, samples constituting each half of the Haar filter represent a logical choice for the two background regions (i.e., setting $G = 0$).

Examples of the scaled background estimate (i.e., threshold) obtained using the SO-CFAR and the similarly scaled background estimate using the CA-CFAR are superimposed on the

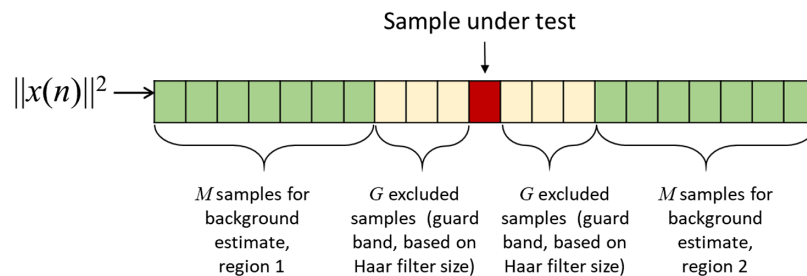


Fig. 4 Illustration of the CFAR architecture. The background estimate for the SUT becomes $\min(A_1, A_2)$, where $A_i = \frac{1}{M} \sum_{n \in \text{region } i} \|x(n)\|^2$ is the average from region i , $i = 1, 2$ and $\min(a, b)$ is the minimum of a and b .

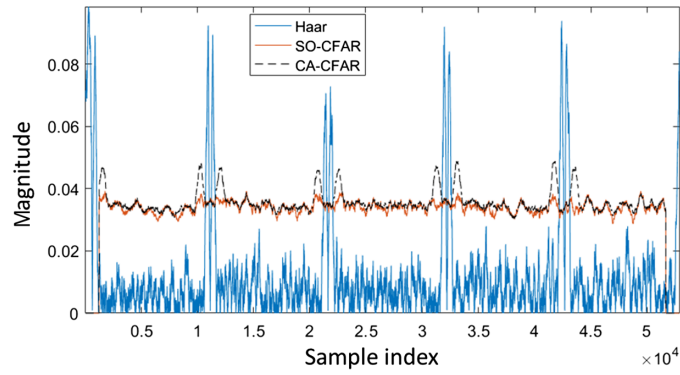


Fig. 5 Plots of similar threshold settings obtained using the SO-CFAR (red) and the CA-CFAR (black dashed). The absolute value (magnitude) of the Haar output is shown in blue. Plots have been scaled to be comparable, and CFAR parameters (guard band and background regions) are the same in both cases. A potential for masking targets is apparent in the vicinity of the rising and falling edge estimates. Edge effects of the calculation are visible at either end of the plot.

magnitude of the Haar filter output in Fig. 5. The problem of potential masking is apparent in the CA-CFAR plot before each rising edge and after each falling edge.

Figure 6 shows a similar plot for the case where $M = N/2$ and $G = 0$ in the SO-CFAR formulation, and we immediately notice a large “spike” in threshold values for test samples within the pulse. Such local maxima could be on the order lower-magnitude signals of interest, as suggested by a comparison with between-pulse samples at the edge of the plots. This could be particularly problematic if a flatter, global threshold is incorporated, as alluded to in our earlier

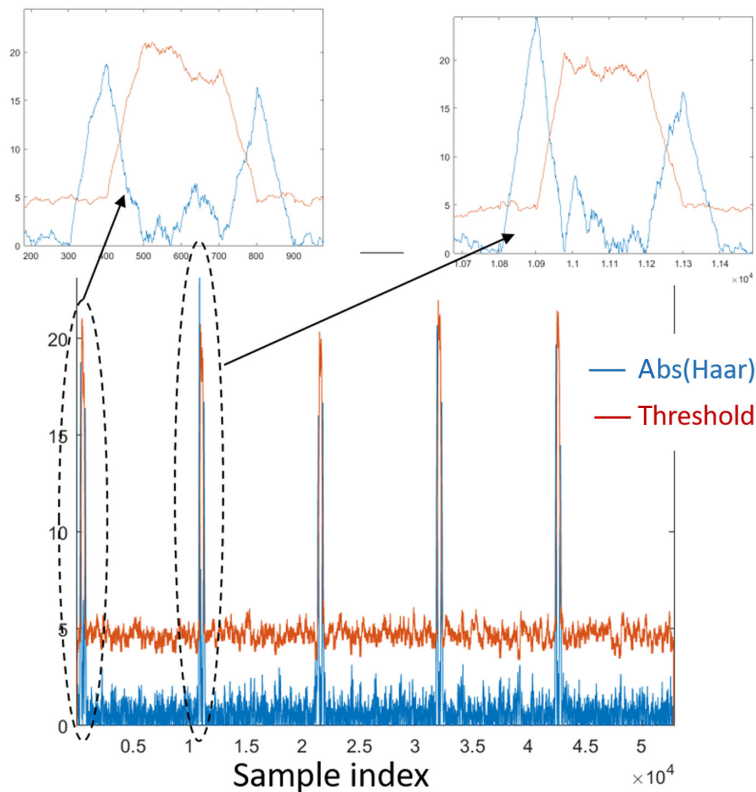


Fig. 6 Plots of threshold settings obtained using the SO-CFAR (red) and the absolute value (magnitude) of the Haar output (blue) for a relatively high SNR of 4.7 dB. Potential intrapulse false alarms are evident in the zoomed plots of the first two pulses if threshold settings remain the same within the pulse as they are between the pulses. Setting $G = 0$ alleviates this problem.

proceedings paper.⁸ These observations support the exclusion of guard regions from the SO-CFAR calculations.

2.2 Determination of Thresholds for the Ratio and Comparison with the Haar Filter

A CFAR test for the ratio can be obtained in a manner similar to that for the Haar filter by beginning with the assumption that I/Q samples constituting the numerator and denominator summations in the ratio, denoted as S_{num} and S_{den} , are i.i.d., $N(0, \sigma^2)$ random variables that are processed according to Eq. (2). In this case, we can view the $y_{\text{out, ratio}}(n)$ as the ratio of two sample variances, and it follows a Fisher–Snedecor F -distribution with parameters (N, N) .²⁵ As a result, we can set a threshold based on the filter size and the Fisher–Snedecor pdf

$$f(x; N, N) = \frac{\Gamma(N)x^{(N-2)/2}}{\Gamma\left(\frac{N}{2}\right)\Gamma\left(\frac{N}{2}\right)(1+x)^N}, \tag{9}$$

where $\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx$ is the gamma function and $\Gamma(N) = (N-1)!$ for integer N . Examples of the F -distribution pdf are included in Fig. 7 for different filter lengths (values of N).

We have developed a second method for performing the ratio test that also does not require an estimate of the parameter σ^2 . We define a new random variable

$$\mathbf{R} = S_{\text{num}}/(S_{\text{num}} + S_{\text{den}}) \tag{10}$$

and observe that it follows a beta($N/2, N/2$) pdf,²⁶ defined as

$$b(x; \alpha, \beta) = x^{\alpha-1}(1-x)^{\beta-1}/B(\alpha, \beta), \tag{11}$$

where $0 \leq x \leq 1$, $B(\alpha, \beta) = (\Gamma(\alpha)\Gamma(\beta))/\Gamma(\alpha + \beta)$, $\alpha = \beta = N/2$, and $\Gamma(\cdot)$ is the gamma function.

Since this distribution has the added advantage of being symmetric about $1/2$, we can set upper and lower thresholds, $0.5 \pm \delta$, where δ is selected using $b(x; N/2, N/2)$ based on the desired false alarm probability. Values of $\mathbf{R} < 0.5 - \delta$ indicate falling edges, whereas values of $\mathbf{R} > 0.5 + \delta$ indicate rising edges. As a result, calculation of a “reciprocal” threshold (to detect

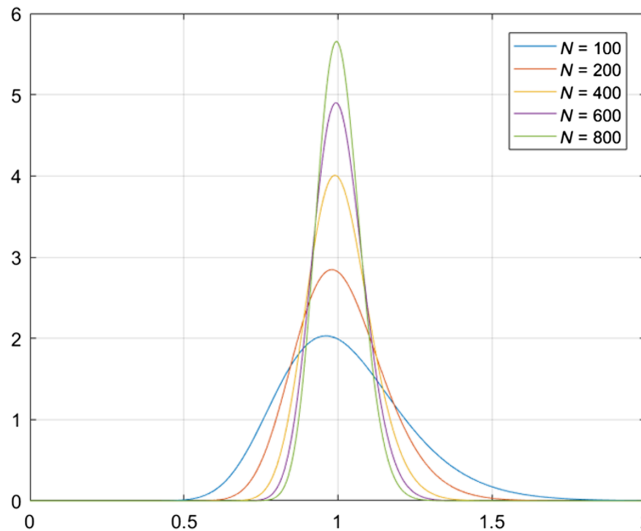


Fig. 7 Plots of Fisher–Snedecor pdf for different values of N . Generated using built-in Matlab F -distribution function.

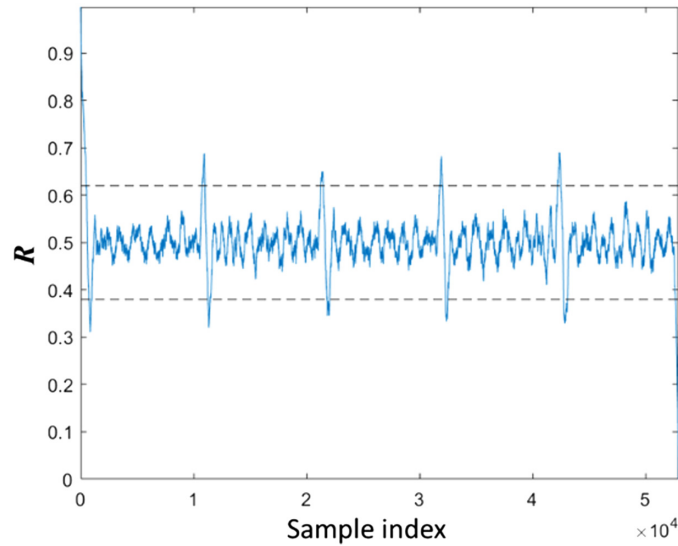


Fig. 8 Plot of modified ratio, R , for sample data of Figure 1. $\Pr\{R < 0.38\} + \Pr\{R > 0.62\} \approx 1.1254 (10^{-6})$ for $N/2 = 200$ (smaller than the window used). Initial and final samples are discarded due to edge effects.

the falling edge shown in Fig. 3) is no longer necessary. Figure 8 shows a plot of the ratio R from Eq. (9) for $N/2 = 400$, whereas Fig. 8 shows plots of beta pdfs for several values of $N/2$. Notional threshold values are included in Fig. 8 and superimposed on Fig. 9, as well. For $N/2 = 200$ and $\delta = 0.12$, we calculate a predicted false alarm probability for R of $\Pr\{R < 0.38\} + \Pr\{R > 0.62\} \approx 1.13 \times 10^{-6}$. Hence, we would expect to see no samples < 0.38 or > 0.62 within the plot of Fig. 8, and that is, indeed, the case.

From Eq. (11), we also observe that the size of the moving average, $N/2$, is the only parameter required to calculate the detection threshold. By contrast, the Haar filter requires the definition of additional parameters. If we leverage the formulation of Eq. (7), for example, then we must specify the scale factor, k . Such a scale factor must be determined experimentally if no assumptions regarding the underlying between-pulse statistics are made. So, a threshold calculation for the Haar outputs similar to that for the ratio in Eq. (9) would require either the

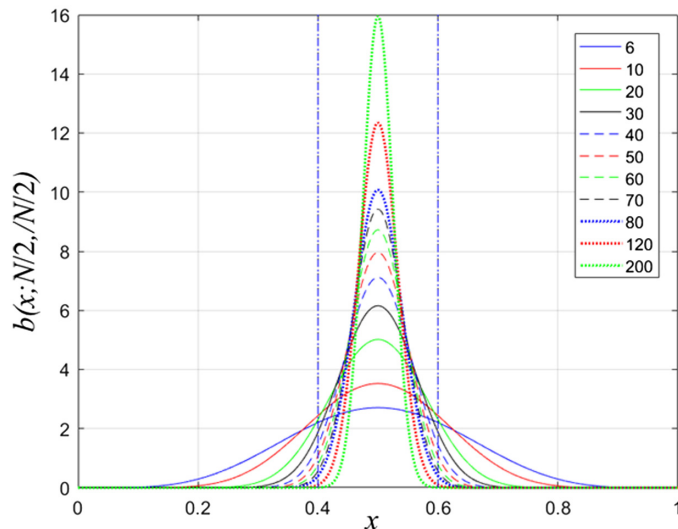


Fig. 9 Plots of $b(x; N/2, N/2)$ for different values of $N/2$. Generated using built-in MATLAB beta distribution function.

evaluation of Whittaker’s W function or a careful numerical integration.²¹ Calculation of $\hat{\sigma}_{\text{difference}}$ would also be required for both approaches [i.e., the approach based on Eq. (7) and the approach exploiting the W function]; this estimate could leverage the existing averages used to calculate Haar filter outputs in Eq. (1). Since $\hat{\sigma}_{\text{difference}}$ will vary with time, multiple evaluations of Eq. (7) or the W function may become necessary. The calculation of the threshold T_{ratio} based on Eq. (9) requires similar calculations to those of T_{Haar} , relying on the same moving averages of the input data stream. However, threshold calculations for \mathbf{R} are based exclusively on $N/2$, which remains constant and is defined *a priori*. We will refer to \mathbf{R} as “beta” to distinguish it from the ratio of Eq. (2).

3 Peak Location and Edge Association

Because both of the PW estimators considered here incorporate moving averages, the transitions between inter- and intrapulse regions are gradual. (The plots in Figs. 3 and 8 show this behavior.) Hence, several samples near the peak will likely pass the threshold test, creating the need for a local maximum estimator (local max). The output of this local max, shown by block 5 in Fig. 1, is described by

$$f_{\text{max}}(n) = \begin{cases} (n) & \text{if } |f(n)| = \max_{n \in [n-w, n+w]} |f(n)| \\ 0 & \text{otherwise} \end{cases} . \quad (12)$$

Here, the value of w is selected to eliminate false alarms at locations that are close to (or within) the pulse, but not on one of its edges. This typically occurs when the leading or lagging convolution window includes both pulse and nonpulse samples. Note that w will typically be a fraction of the Haar filter length (e.g., $N/2$). Since this local max operation is serial, it introduces an additional delay of w Haar filter output or ratio output samples. If a delay of $N/2$ samples is required to accumulate the input samples for the local max—the case for both the Haar filter and the ratio—the total latency through the local max becomes $N/2 + w$ samples.

The second part of the operation of box 5 in Fig. 1 comprises an algorithm developed to associate rising (leading) and falling (trailing) pulse edges to produce PW estimates. This approach represents a generalization of an approach proposed by Smith,²⁷ and it is intended for eventual realization via a hardware design or a hardware/software co-design. The outline below summarizes the process, which attempts to minimize the computations required to perform the association. The procedure consists of the steps in Algorithm 1.

Algorithm 1 Edge association algorithm

-
-
1. LAE = []; /* list of all pairs of associated edges: 2xD array if there are D detections */
 2. LURE = []; /* list of unassociated rising edges */
 3. startIndex = locateFirstRisingEdge(data); /* location of first rising edge. “data” are the input data stream */
 4. LURE[0] = startIndex;
 5. ureCounter = 0; /* largest index of LURE that contains valid data */
 6. stopCount = 1e10; /* number of data samples to be processed */
 7. outputDataCounter = 0; /* index of output data samples */
 8. lastEdgeFalling = 0; /* most recently encountered edge is not a falling edge */
 9. for i = startIndex+1:stopCount
 10. currentEdge = data(i); /* data are the input data stream */
-
-

Algorithm 1 (*Continued*).

```

11. currentIndex = i;
12. if currentEdge > 0      /* current edge is a rising (leading) edge */
13.   if lastEdgeFalling == 1
14.     ureCounter = ureCounter-1;      /* no consecutive falling edges: overwrite the saved rising edge */
15.   end
16.   ureCounter = ureCounter+1;      /* add current index to the list of unassociated rising edges*/
17.   LURE[ureCounter] = currIndex;    /* increment count, there is always at least one element of LURE*/
18.   lastEdgeFalling = 0;      /* reset flag to indicate last edge not a falling edge */
19. else if currentEdge < 0      /* current edge is a falling (trailing) edge
20.   LAE[0] [outCounter:outCounter+ureCounter-1] = LURE;      /* add all elements of LURE to LRE
21.   /*associate all elements of LURE with currentEdge */
22.   LAE[1] [outCounter:outCounter+ureCounter-1] = currentEdge;
23.   tmp = LURE[ureCounter-1];      /* save the most recent rising edge */
24.   LURE = [];      /* clear the LURE */
25.   LURE[0] = tmp;      /* load most recent rising edge back into LURE */
26.   ureCounter = 0;      /* reset the unassociated rising edge counter to initial array element */
27.   outCounter = outCounter+ureCounter;      /* increment the global output counter */
28.   lastEdgeFalling = 1;      /* set flag to indicate last edge was a falling edge */
29. else      /* Do nothing. Samples = 0 did not pass detection and local max. */
30. end      /* if */
end      /* for */

```

The algorithm maintains a list of unassociated leading (rising) edges and a list of associated edge pairs. When a new falling edge is encountered, it is associated with all unassociated rising edges, and the newly associated pairs of rising and falling edges are added to the list of associated rising and falling edges. Following a successful edge association, the most recent rising edge is retained in the list of unassociated edges (in case the next encountered edge is a falling edge). If the next encountered is a falling edge then the process repeats, and the most recent rising edge remains as the only entry in the list of unassociated rising edges. However, if the edge following a successful association is a rising edge, this rising edge replaces the already-used rising edge in the list of unassociated rising edges. If multiple rising edges are encountered, then they are added to the list of unassociated edges and remain in the list until a falling edge is encountered.

This edge association is performed as outputs of the local max become available, so it only introduces latencies associated with the data transfer between lists. It does not require accumulation of additional data samples.

4 Experimental Evaluation of Haar and Ratio-Based Methods

The DEVCOM Army Research Laboratory has recently obtained highly flexible signal generation and recording hardware. This hardware enables researchers to generate and record pulse trains comprising various parameters. That is, it allows the user to record pristine transmitted waveforms with various PWs, PRIs, and waveform modulations. For our experiments, we modified the SNR both by attenuating the signal within the hardware and by injecting noise into

pristine data. A sample data set is shown in Fig. 2, for an SNR of ~ 0 dB. Here, we have adjusted the SNR by adding appropriately scaled, complex, i.i.d. Gaussian noise to nearly pristine, measured I/Q data samples. We have also scaled and translated the Haar filter output so that the Haar filter and ratio plots are aligned in the plots of Fig. 2. Following alignment, the locations of both rising and falling edges are readily apparent.

All calculations were performed using Mathworks' MATLAB computing platform. In particular, the Haar filter and the rectangular window filters (for the ratio tests) were constructed and outputs were generated using built-in convolution functions and vector operations. This approach realized the desired system operation while adhering to MATLAB's recommended practices. It also eliminated explicit programming of loops from the software.

To compare the performance of the techniques for rising edge detection, we compared the aligned peak values from the rising edges with the average of the between-pulse samples. Since the ratio test produced an estimate that was asymmetric about 1, we included only interpulse values greater than one when calculating the background average. We repeated the process for a second data set generated by the signal generation hardware and shown in Fig. 10, where the lower-SNR samples (at ~ -2.9 dB) extracted for evaluation are indicated by the dashed box. This data set comprised the same signal at two different attenuations. The transition between attenuation levels is evidenced by the sudden increase in SNR, making the pulses visible even without any additional integration. The results of this analysis are summarized in Table 1, where we have denoted the data in Fig. 2 "data set 1" and the data in Fig. 10 "data set 2." Here, the peak-to-background ratio (PBR) was calculated by first converting all test statistics to zero mean and then scaling them relative to a reference peak value. Following this step, the reference peak for the Haar and ratio test statistics had the same value. Next, leading-edge peaks were selected and averaged, and between-pulse samples were also selected and their magnitudes averaged. The ratio between these quantities constitutes the PBR shown in Table 1, and it provides a measure of how easily a pulse edge could be detected. A larger PBR implies that, on average, a larger threshold may be set, thereby eliminating potential false alarms. From Table 1, we see that all of the approaches are similarly effective pulse detectors. The performance difference between the Haar and ratio tests is not significant (between 0.3% and 1% difference in PBR). Similarly, the difference between the Haar and beta tests is between 1.5% and 3.7%. As noted previously,^{1,2,7,8} the PW estimator improves as the size of the moving averages (used to calculate the ratio and Haar filter outputs) increases, and the best performance is achieved when the Haar filter length is equal to twice the PW.

A second analysis was conducted to further characterize the algorithms' PW-estimation performance as a function of SNR. In this case, both the root mean-squared (RMS) and the standard deviation of the PW estimation errors were considered. To create the requisite variations in SNR, we leveraged another hardware-generated data set that featured multiple PRIs and the pristine waveform samples shown in Fig. 11(d). Gaussian noise was added to the data to obtain specified pulse-to-noise ratios (PNRs) as described by the procedure in the appendix of Ref. 11. That is,

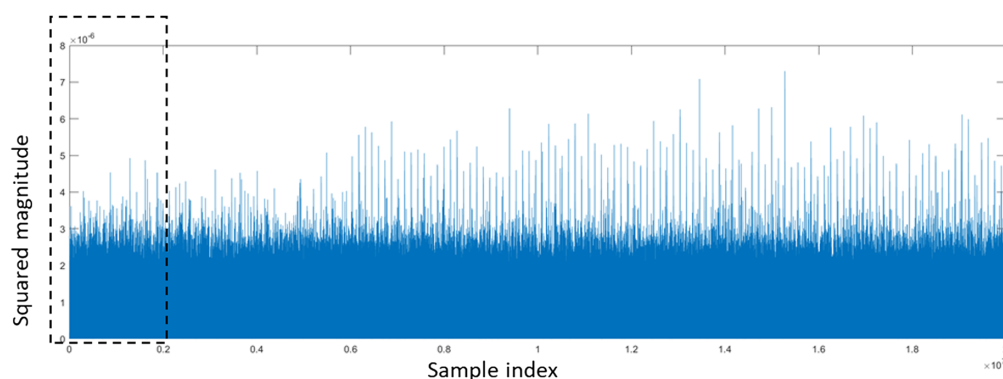


Fig. 10 Input data set 2 from the signal generation hardware. In this scenario, the target changes range abruptly (moves closer) in the vicinity of sample 6,000,000. We focus attention on the lower-SNR pulses indicated by the dashed box. SNR ~ -2.9 dB.

Table 1 Comparison of ratio, Haar, and beta approaches for rising edge detection. All of the peak-to-background statistics are similar, indicating similar performance for edge detection. PW estimator accuracy is considered separately. Haar filter and beta outputs have been scaled and translated to match the ratio outputs as in Fig. 2. SNR for data set 1 ≈ 0 dB, SNR for data set 2 ≈ -2.9 dB.

Comparison of edge detection performance						
Data used	Test	Normalized background mean	Normalized peak mean	PBR	Delta, ratio-to-Haar (approx.)	Delta, Haar-to-beta (approx.)
Data set 1	Ratio	1.0859	2.1100	1.9431	0.023/1.966 (1.1%)	0.0719/1.966 (3.7%)
	Haar	1.0868	2.1368	1.9661		
	Beta	1.1292	2.1390	1.8942		
Data set 2	Ratio	1.0518	1.5591	1.4839	0.0044/1.4795 (0.3%)	0.0225/1.4795 (1.5%)
	Haar	1.0507	1.5561	1.4795		
	Beta	1.0724	1.5624	1.4570		

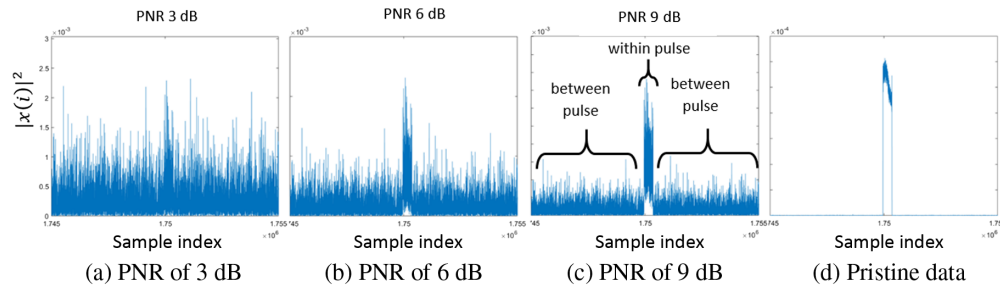


Fig. 11 Pulse from data set 1 showing effects of noise injection. Noise level adjusted for (a) PNR = 3 dB, (b) PNR = 6 dB, and (c) PNR = 9 dB. (d) Pristine data (prior to noise injection).

I/Q noise samples were scaled so that, following addition, the ratio of the mean-squared within-pulse values to the mean-squared between-pulse values achieved the desired ratio, defined to be the PNR. These values could then be translated to SNR if desired. Examples of preprocessed (spoiled) data are included in Fig. 11 for different PNR levels. Plot (c) further identifies which

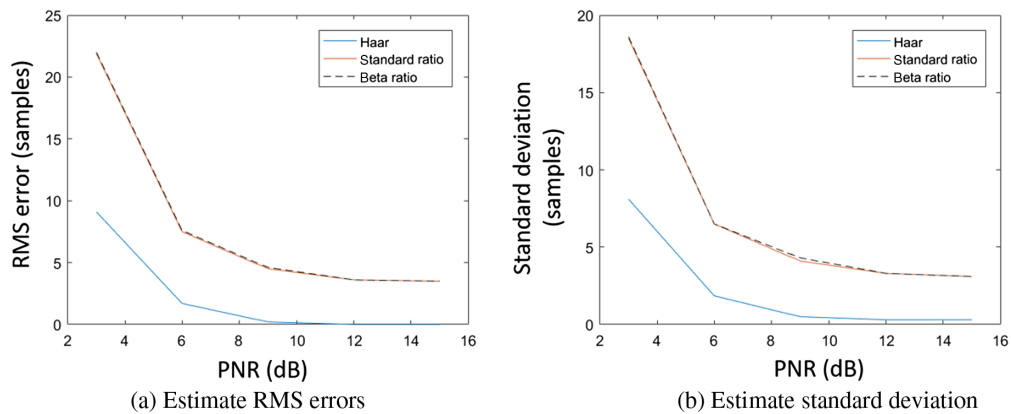


Fig. 12 RMS error and standard deviation as a function of PNR. Performance of ratio-based approaches similar (curves coincident), while the performance of the Haar approach is superior (lower RMS error and lower estimation variance at all values of PNR). (a) RMS errors. (b) standard deviations.

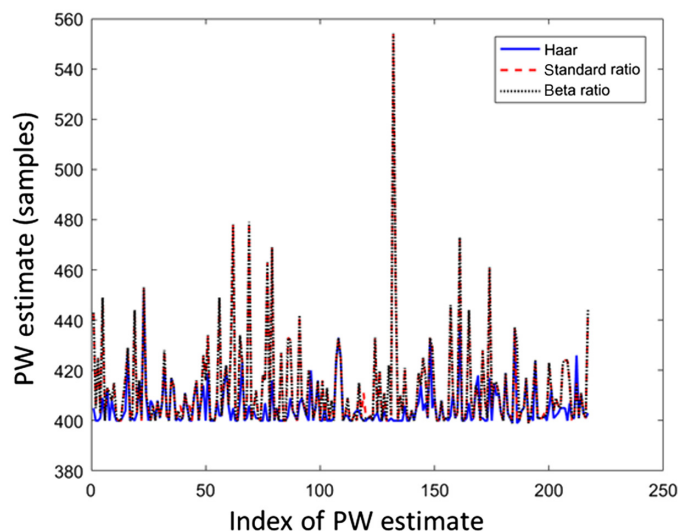


Fig. 13 PW estimates for a PNR of 3 dB; the true PW is 400 samples. Several samples exhibiting high standard ratio or beta ratio errors do not exhibit high Haar errors. Note that the radius of the local max was set at 400, so no PW estimates were <400.

samples (within-pulse) were used to calculate the numerator of the PNR and which samples (between-pulse) were used to calculate the denominator. Here, the location of pulse edges was determined from the pristine data.

The performance of each method is summarized in the plots of Fig. 12 for multiple values of SNR. Since the true PW was 400 samples, Fig. 12(a) shows that the RMS error for the Haar filter method was <1% for PNR > 6 dB (SNR > 4.7 dB). However, the ratio methods achieved RMS errors approaching 1% only for PNRs of 12 or greater. The standard deviation shows a similar result, with the Haar filter outperforming the ratio-based methods. The plots in Fig. 13 show the PW estimates used to calculate the statistics in Fig. 12. Here, the PNR has been fixed at 3 dB, and large errors in the ratio-based estimates are readily apparent. Note that the local max radius [w in Eq. (10)] was set to half of the Haar filter length [$N/2$ in Eq. (1)]; hence, there are no PW estimates <400. For these investigations, the filter length was set to $2 \times$ PW, the optimal value.

5 Summary and Conclusions

We have described and analyzed three PW estimators, two of which exploit the ratios of moving sums and one of which exploits the differences of moving sums. The analyses indicated that all three approaches were similarly effective at *detecting* rising pulse edges. However, the Haar filtering, difference-based approach (often referred to as a “difference of boxes”) produced the most accurate *PW* estimates, performing effectively down to SNRs approaching -3 dB. Here, the *pulse detector* performance was quantified via the PBR, whereas the *PW estimator* performance was quantified in terms of the RMS error and the standard deviation of the *PW* estimates.

Relative performance of the ratio and Haar filtering approaches was first evaluated in terms of the percentage difference between the PBRs of the various filter outputs. A higher PBR indicated that a higher threshold value could be used to obtain the same probability of detection; hence, if the PBR of one approach were significantly higher than that of another approach, the approach with the higher PBR was deemed to be more effective. Here, the Haar approach served as the reference for scaling and translating outputs to a common scale, and two different data sets were leveraged to perform the evaluation. These metrics indicated the amount of contrast between the average of output peaks (at the pulse leading edge locations) and the average of between-pulse samples differed by <4% in the worst case and differed by 1.5% or less in typical cases. The largest differences occurred at the lowest SNR levels, and the performance of the standard ratio was closest to that of the Haar filter. Both the Haar filter and standard ratio approaches exhibited higher PBRs than the beta ratio approach. The similarity of the PBRs confirmed qualitative

conclusions based on the similarities between appropriately scaled plots of outputs from the various approaches. These statistics were quantitatively similar, indicating that all approaches are, on average, similarly effective at *detecting* pulse edges.

A second experiment was performed to evaluate PW estimator performance using the RMS error and the estimator standard deviation. Plots of PW estimates obtained at a PNR or 3 dB (SNR of 0 dB) were also included to further illustrate the behavior of the two approaches. The comparison demonstrated that

1. all approaches produced PW estimate errors of approximately eight samples (2% of the true PW) or less for PNRs ≥ 6 dB (SNR > 4.7 dB),
2. the Haar filter approach continued to perform well at SNRs of ~ 0 dB (RMS error $< 3\%$ of the true PW),
3. the Haar filter approach consistently outperformed the ratio-based approaches, and
4. the performance of all approaches improved monotonically, i.e., the RMS error decreasing as the PNR increased.

As part of the algorithm evaluations, we also described the statistical behavior of various estimators under certain reasonable assumptions about the probability distributions governing between-pulse data samples. In particular, we demonstrated an effective SO-CFAR formulation and threshold-calculation procedure for the Haar filter approach that did not require use of the filter outputs; only the moving sums constituting the difference were required. For the ratio-based approaches, the threshold was determined based on the filter size alone, and the moving sums were not needed.

Of all methods considered, the Haar filter approach was particularly appealing due to its lack of division computations. In addition to this, the algorithm continually updated the two moving sums that were leveraged for both the filter output and the threshold calculations; hence, the adaptive threshold incurred no significant additional cost. For these reasons, the Haar filter appears particularly well suited for hardware design and implementation. It shows promise when signals of interest are powerful enough to allow for noncoherent processing.

References

1. X. Liu, X. Fan, and S. Su, "Adaptive pulse edge detection algorithm based on short-time Fourier transforms and difference of box filter," *J. Appl. Remote Sens.* **13**(2), 024502 (2019).
2. X. Fan et al., "A new method of pulse edge detection in low SNR," *Proc. SPIE* **9812**, 981213 (2015).
3. A. A. Adam, B. A. Adegboye, and I. A. Ademoh, "Inter-pulse analysis of airborne radar signals using smoothed instantaneous energy," *Int. J. Signal Process. Syst.* **4**(2), 139–143 (2016).
4. B. M. Albaker and N. A. Rahim, "Signal acquisition and parameter estimation of radio frequency pulse radar using novel method," *IETE J. Res.* **55**(3), 128–134 (2009).
5. T. Pare, C. C. Tung, and K. Uln, "Configurable radar detection and avoidance system for wireless OFDM transceivers," US 2009/0160696A1 (2009).
6. A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," *IEEE Trans. Commun.* **46**(4), 431–436 (1998).
7. C. Li et al., "Pulse train detection algorithm based on edge enhancement," in *Proc. Int. Conf. Wireless Commun. and Sens. Networks*, Association for Computing Machinery, New York, pp. 30–34 (2020).
8. K. Ranney et al., "An efficient pulse detector and pulse width estimator for waveform characterization," *Proc. SPIE* **11742**, 117421E (2021).
9. S. R. Joshi and R. Koju, "Study and comparison of edge detection algorithms," in *Proc. IEEE Third Asian Himalayas Int. Conf. Internet*, pp. 1–5 (2012).
10. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Chapter 7, pp. 413–432, Addison-Wesley Publishing, New York (1993).
11. R. Touzi, A. Lopes, and P. Bousquet, "A statistical and geometrical edge detector for SAR images," *IEEE Trans. Geosci. Remote Sens.* **26**(6), 764–773 (1988).

12. M. Weiss, "Analysis of some modified cell-averaging CFAR processors in multiple-target situations," *IEEE Trans. Aerosp. Electron. Syst.* **AES-18**(1), 102–114 (1982).
13. A. Zaimbashi, "An adaptive cell averaging-based CFAR detector for interfering targets and clutter-edge situations," *Digital Signal Process.* **31**, 59–68 (2014).
14. H.V. Poor, "Signal detection in discrete time," Chapter 2 in *An Introduction to Signal Detection and Estimation*, Springer Texts in Electrical Engineering, pp. 45–104, Springer, New York (1994).
15. A. A. Ahmad et al., "Extraction of the pulse width and pulse repetition period of linear FM radar signal using time-frequency analysis," *J. Adv. Sci. Eng.* **3**, 1–8 (2020).
16. L. M. Hoang, M. Kim, and S. Kong, "Automatic recognition of general LPI radar waveform using SSD and supplementary classifier," *IEEE Trans. Signal Process.* **67**(13), 3516–3530 (2019).
17. J. Wang et al., "Detection algorithm of big bandwidth chirp signals based on STFT," *Proc. SPIE* **9252**, 92520M (2014).
18. M. Evans, N. Hastings, and B. Peacock, "Normal (Gaussian) Distribution," Chapter 30 in *Statistical Distributions*, Wiley Series in Probability and Statistics, pp. 145–151, Wiley & Sons, Inc., New York (2000).
19. M. Evans, N. Hastings, and B. Peacock, "Gamma Distribution," Chapter 19 in *Statistical Distributions*, Wiley Series in Probability and Statistics, pp. 98–106, Wiley & Sons, Inc., New York (2000).
20. "Wolfram Mathworld," <https://mathworld.wolfram.com/GammaDistribution.html>.
21. B. Klar, "A note on gamma difference distributions," *J. Stat. Comput. Simul.* **85**, 3708–3715 (2015).
22. S. Ross, *A First Course in Probability*, 8th ed., Chapter 7, pp. 322–331, Pearson Prentice Hall, Saddle River, NJ (2010).
23. E. K. Al-Hussaini and B. M. Ibrahim, "Comparison of adaptive cell-averaging detectors for multiple-target situations," *IEE Proc. Part F* **133**(3), 217–223 (1986).
24. H. M. Finn and R. S. Johnson, "Adaptive detection mode with threshold control as a function of spatially sampled clutter estimation," *RCA Rev.* **29**(3), 414–464 (1968).
25. M. Evans, N. Hastings, and B. Peacock, "F (Variance Ratio) or Fisher-Snedecor Distribution," Chapter 17 in *Statistical Distributions*, Wiley Series in Probability and Statistics, pp. 90–94, Wiley & Sons, Inc., New York (2000).
26. M. Evans, N. Hastings, and B. Peacock, "Beta Distribution," Chapter 5 in *Statistical Distributions*, Wiley Series in Probability and Statistics, pp. 34–42, Wiley & Sons, Inc., New York (2000).
27. D. A. Smith, "A quantitative method for the detection of edges in noisy time-series," *Philos. Trans. Biol. Sci.* **353**(1378), 1969–1981 (1998).

Kenneth Ranney is a research engineer at the DEVCOM Army Research Laboratory (ARL), where he focuses primarily on the development of detection and signal processing techniques for radar and other electronic sensor systems. He received his undergraduate degree from Johns Hopkins University in Baltimore and received his graduate degree from the University of Maryland, College Park. He is currently a chair of the SPIE Radar Sensor Technology Conference within the annual Defense and Commercial Sensing Symposium.

Kwok Tom retired in 2020 as researcher for the DEVCOM Army Research Laboratory (ARL). During his tenure, his work focused on target detection, ultrawideband radar signal processing for the detection and identification of underground objects, Ka-band monopulse radar system development, prognostic and diagnostic technique development for condition-based maintenance, and applications of spectrum sensing and modulation recognition for electromagnetic situational awareness.

Dannielle Tadas is an electronics engineer at the DEVCOM Army Research Laboratory (ARL), where she supports research in electronic warfare. She received her MS degree in electrical engineering and has worked in the telecommunications field in the past.

Neal Tesny is a research scientist at the DEVCOM Army Research Laboratory (ARL), where he conducts research involving electronic warfare (EW) and electromagnetic (EM) spectrum utilization. He received his MS degree in electrical engineering from Johns Hopkins University. His experience with EM research extends back to 1987 and has concentrated on EW related areas since 2012. His current interests include researching EM spectrum capabilities via computer simulation and hardware and software development and testing.

Andre Magill is an electronics engineer at the DEVCOM Army Research Laboratory (ARL) in Adelphi, Maryland. His research focuses on the implementation of radar and target models and the development of advanced electronic warfare techniques in hardware-in-the-loop environments. He graduated from Notre Dame and is currently pursuing a graduate degree at Johns Hopkins University.

William Diehl is a researcher at the DEVCOM Army Research Laboratory (ARL), where he conducts foundational research to improve U.S. Army and joint service electronic warfare and electromagnetic spectrum capabilities. He is a retired U.S. Navy Cryptologic Officer, and an alumnus of the Naval Postgraduate School, where he received his MS degree in electrical engineering. He received his PhD from George Mason University in 2018 and is an author of more than 30 peer-reviewed publications.