

# A Fast 2D-AR(1) Filtering for Bitemporal Change Detection on UWB SAR Images

Marcello Costa<sup>a</sup>, Ingo Sander<sup>a</sup>, Ingemar Söderquist<sup>a,b</sup>, Patrick Dammert<sup>b</sup>, Anders Åhlander<sup>b</sup>, and Christer Fuglesang<sup>a</sup>

<sup>a</sup>KTH Royal Institute of Technology, Stockholm, Sweden

<sup>b</sup>Saab AB, Sweden

## ABSTRACT

This article presents an elementary change detection algorithm designed using a synchronous model of computation (MoC) aiming at efficient implementations on parallel architectures. The change detection method is based on a 2D-first-order autoregressive ([2D-AR(1)]) recursion that predicts one-lag changes over bitemporal signals, followed by a high-parallelized spatial filtering for neighborhood training, and an estimated quantile function to detect anomalies. The proposed method uses a model-based on the functional language paradigm and a well-defined MoC, potentially enabling energy and runtime optimizations with deterministic data parallelism over multicore, GPU, or FPGA architectures. Experimental results over the bitemporal CARABAS-II SAR UWB dataset are evaluated using the synchronous MoC implementation, achieving gains in detection and hardware performance compared to a closed-form and well-known complexity model over the generalized likelihood ratio test (GLRT). In addition, since the one-lag AR(1) is a Markov process, its extension for a Markov chain in multitemporal ( $n$ -lags) analysis is applicable, potentially improving the detection performance still subject to high-parallelized structures.

**Keywords:** SAR UWB, change detection algorithms, time series prediction, synchronous MoC, deterministic parallelism.

## 1. INTRODUCTION

In remote sensing, change detection allows rapid mapping analysis between a new acquisition and the available data in the same analytical area.<sup>1</sup> This application aims to identify anomalies over different images considering a background model, being traditionally solved with statistical inference on well-fitted models.<sup>2–5</sup> In those approaches, the local interference is reduced with a constant false alarm rate (CFAR) filtering, which increases with quadratic complexity the processing demands to maintain the optimal decision under maximum detection probability ( $P_D$ ).<sup>6</sup>

In particular, imagery systems based on ultra-wideband (UWB) radars produce stable scatters in time due the low frequency, being less influenced by environmental effects.<sup>7</sup> This characteristic is useful for detecting concealed targets such as foliage penetration applications,<sup>8</sup> or for use in time series methods for change detection. Among those methods, the autoregressive (AR), autoregressive, and moving average (ARMA), and their variations have been applied in,<sup>9–12</sup> whose detection performance and complexity depend on the statistical pixel model, the order of time series, and the appropriate spatial size for training data. The highest complexity is due to the numerical methods involved in the inference, usually with non-trivial parallelized structure for hardware acceleration.

Assuming the random motions of targets in the clutter, a simplest first-order autoregressive [AR(1)] process can be used, similarly as considered in Ref. 13 for the motions in the atmosphere. In such case, the AR(1) process describes a stochastic time series for stationary noise with zero mean, constant variance, and positive autocorrelation defined between two successive time samples. Hence, the memory between time-spaced bitemporal signals, in terms of autocorrelation, determines the degree of change in the updated signal. Since that the recursion involves storing states to construct the outputs conditioned to the current inputs, an elementary

---

Further author information: (Send correspondence to M.G.C.)  
M.G.C.: E-mail: marcello.costa.br@ieee.org

sequential structure of a synchronous model of computation (MoC)<sup>14</sup> can be used for implementation. Therefore, elements of the computational system can be directly associated with the change detection model, enabling a prediction of the hardware performance.

In this article, the proposed synchronous model of computation uses the tools of ForSyDe<sup>15</sup> to develop an elementary change detection (bitemporal) algorithm. This model performs the AR(1) filtering efficiently to produce the change map subject to a refined anomaly detection. The processes involving events over the signal are adjusted for suitable matrices manipulation and parallelism scheduling, reflecting in runtime and power consumption performance. The detection performance is compared with the well-known baseline algorithm presented in Ref. 16.

The rest of the article is structured as follows. Section 2 presents the dataset details as a baseline statistical method, used as a reference. Section 3 describes the proposed 2D-AR(1) model for change detection and its derivations. The implementation using a synchronous MoC is presented in Section 4. Experimental results are shown in Section 5, and finally, the conclusions are in Section 6.

## 2. BITEMPORAL DATASET

The bitemporal dataset used in this article was obtained by the Swedish Defence Research Agency (FOI) in 2002, using the UWB VHF SAR CARABAS II system.<sup>7</sup> The collection consists of 24 incoherent wavelength-resolution SAR images of a ground area of  $2 \times 3$  km. The images content 25 target vehicles hidden under two different areas of homogeneous Boreal forest in Vidsel, north of Sweden. The ability of detection varies through 24 bitemporal pairs, defined on the combinations between six groups of flight direction (passes: 1,2,3,4,5,6) and four different target positioning (missions: S1, K1, F1, and AF1).

Particularly, for our tests we selected the area of  $2 \times 0.5$  km, covering the region around the targets to construct and test the elementary bitemporal detector on synchronous model of computation, which can be extended for the full dataset. Fig. 1 shows a sample of the dataset used in this article model, where 4 targets positioning are tested subjected to their previous scenario.

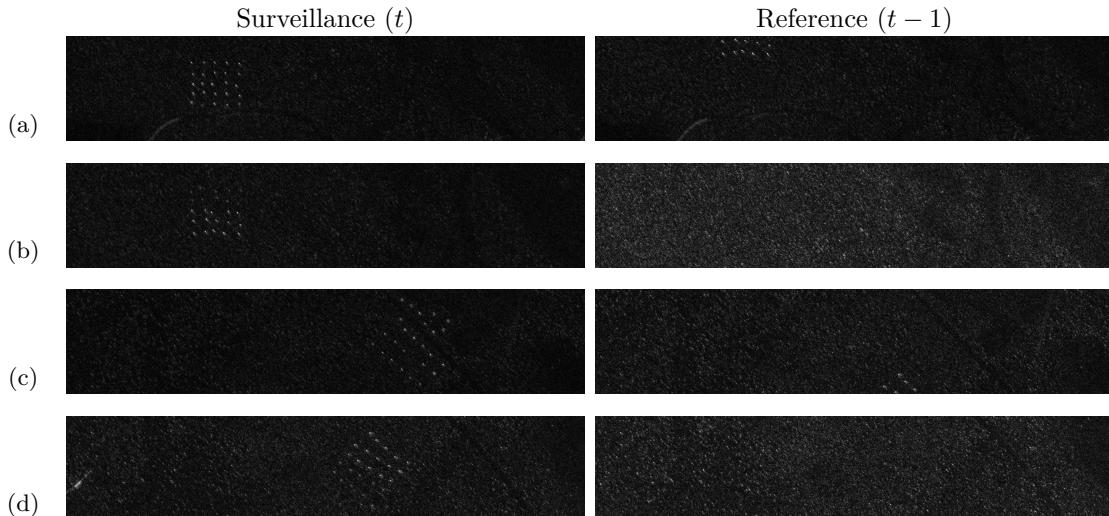


Figure 1: Sample of  $2 \times 0.5$  km of four missions under the specific pass 1 from the original bitemporal CARABAS-II SAR UWB dataset: (a) Mission S1, (b) Mission K1, (c) Mission F1, and (d) Mission AF1.

Such a problem is solved in classical detection theory using binary hypothesis testing, where a log-likelihood ratio (LRT) determines the alternative hypothesis  $\mathcal{H}_1$  against the null hypothesis  $\mathcal{H}_0$ , defined in terms of their probability density functions (pdf). The optimal solution in the Neyman-Pearson (NP) sense, establishes the region  $\mathcal{R}_1$  where the power or probability of detection ( $P_d$ ) is maximum, given a fixed false alarm probability ( $P_{FA}$ ), being adjusted in terms of a scalar threshold  $\gamma$ .<sup>17</sup> The composition model in bitemporal approach considers

the previous image, as the background (clutter), and the current image under testing, as a target at anomaly presence.

Usually, homogeneous magnitude clutter is characterized by Gaussian, exponential, log-normal, K, gamma, and  $\mathcal{G}_A^0$  distributions.<sup>18</sup> A simplest bivariate mixture model for complex SAR images derived in Ref. 19, describes as a circular Gaussian process  $\mathbf{z} = |a| \exp^{j\phi} \mathbf{s}$ , with amplitude  $a$  and random phase  $\phi$  uniformly distributed between 0 and  $2\pi$ , resulting in the close-form likelihood-ratio test (LRT)

$$\ln \Lambda(z) = \frac{|\mathbf{s}^T C^{-1} \mathbf{z}|}{\mathbf{s}^T C^{-1} \mathbf{s}} \stackrel{H_1}{\leqslant} \stackrel{H_0}{\geqslant} \gamma. \quad (1a)$$

where  $\mathbf{s}$  is vector component of the signal,  $\mathbf{z}$  is the vector of samples with mixtures of components  $x$  and  $y$ , and the covariance matrix  $C$  is defined over  $i = 1, \dots, NM$  positions where the random vector  $\mathbf{z}$  is evaluated, so that

$$C = \begin{bmatrix} \sigma_x^2 & -\rho_{xy}\sigma_x\sigma_y \\ -\rho_{yx}\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}. \quad (1b)$$

Given that the elements of  $C$  are estimated, the test is extended to a generalized likelihood-ratio test (GLRT).

The baseline structure for the change detection algorithm presented in Ref. 16 selects the element of  $\mathbf{s}$  and its applicable row in  $C$  to define the change direction. Substituting this convection in Eq. (1a), is obtained

$$\ln \Lambda(z) = \frac{\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} \sigma_y^2 & -\rho_{xy}\sigma_x\sigma_y \end{bmatrix} \begin{bmatrix} x_{n,m} \\ y_{n,m} \end{bmatrix}}{\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} \sigma_y^2 & -\rho_{xy}\sigma_x\sigma_y \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} \stackrel{H_1}{\leqslant} \stackrel{H_0}{\geqslant} \gamma \quad (2)$$

where the the estimation of  $C$  elements determines the degree of change between the bi-dimensional signals  $x_{n,m}$  and  $y_{n,m}$ , subject to  $\gamma$ .

In addition, the spatial analysis in such a model applies convolution windowing  $N \times N$  for  $C$  estimation and testing evaluation, followed by pixel-level CFAR filtering to adapt the local threshold and maintain the optimal detection. Hence, the overall solution implies a quadratic complexity in both stages, which makes the hardware acceleration and memory requirements essential for effective computation. Other more advanced distributions were explored in such model, based on bivariate Gamma,<sup>5</sup> Rayleight, and K-distributions.<sup>20</sup> However, the costs to improve the detection performance usually increases the computation demands, given that more complete models have more parameters to be estimated, not always with close-form solution.

### 3. 2D-AR(1) MODEL

A bitemporal time series on a first-order autoregressive model, AR(1), predicts one-lag changes between two input signals delayed in time  $x_t$  and  $y_{t-1}$ . That is the equivalent of a Markovian Process, where the actual stage depends only on the previous to predict the next state. The AR(1) model in Ref. 13 describes a stationary series for independent zero mean with unitary variance Gaussian variables, where the retained memory between the delayed signals is specified in terms of the auto-correlation function (ACF), obeying the Yule-Walker Eq. for one-lag, i.e.,  $\rho = \phi_1$ .<sup>21</sup> This model outputs i.i.d\* samples belonging to a certain positive exponential probability density functions, which point to a unique change direction.

For 2D signals, an extended spatial-temporal series predicts changes between a prior 2-D signal  $y_{t-1}[n, m]$ , and its updated version  $x_t[n, m]$ , where  $t$  and  $t-1$  represent one-time step (one-lag) used in the prediction under a two-dimensional space  $\{n = 1, \dots, N; m = 1, \dots, M\}$ . The AR(1) model can therefore be represented as

$$\hat{y}_t[n, m] = \rho y_{t-1}[n, m] + \sqrt{1 - \rho^2} x_t[n, m], \quad (3a)$$

---

\*i.i.d. stands for independent identically distributed.

where  $\rho$  is the cross-correlation between  $y_{t-1}[n, m]$  and  $x_t[n, m]$ , normalized in the interval  $(0, 1)$ . Assuming the  $T$  observations, the sample cross-correlation is obtained as

$$\rho = \frac{\sum_{i=1}^T (x_i - \bar{x}) \sum_{i=1}^T (y_i - \bar{y})}{\sigma_x \sigma_y}, \quad (3b)$$

where  $\bar{x}$  and  $\bar{y}$  represent the sample means and  $\sigma_x^2$  and  $\sigma_y^2$  are the sample variances.

The filter structure for a dynamic model is shown in Fig. 2, where the AR(1) process produces the change detection map  $\hat{y}_t[n, m]$  according to the current input and an initial state obtained by delay  $D$ . The speed of variation in  $\sqrt{1 - \rho^2}$ , emphasizes the presence of large changes. Such characteristic is useful for anomaly detection on the filter outputs. Even under more complex signal models, these outputs are i.i.d samples positive valued, where one-sided confidence intervals can be effectively implemented to limit the hypothesis testing space, i.e., in the presence of anomaly's movement.

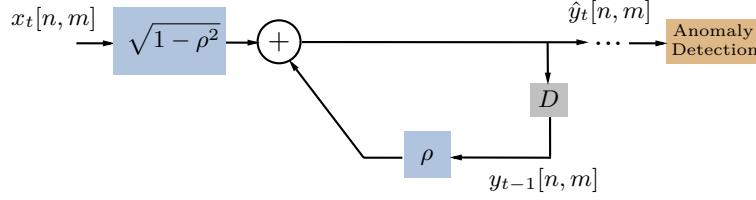


Figure 2: The stationary first-order AR structure

### 3.1 Spatial Filtering and Anomaly Detection

The AR(1) recursion outputs all changes into one-lag, however, since in the typical target-clutter models, the clutter is predominant and target is treated as anomaly, the neighborhood effect can be included to improve the detection performance. In 2-D case, a spatial filtering based on moving average (MA) with finite impulse filters (FIR) produces output that depends on a linear combination of a finite number of samples, defined as

$$\begin{aligned} y[n] &= b_0 x[n] + b_1 x[n-1] + \dots + b_q x[n-q] \\ &= \sum_{q=0}^{N-1} b_q x[n-q] \end{aligned} \quad (4)$$

where  $b_q$  are the linear combination coefficients and  $x$  are the samples, representative for neighborhood. The averaging filtering uses convolution with a stencil kernel to compute an operation on the signal. If a spatial data function  $f[n, m]$  is applied to the kernel convolution, we obtain a smoothed function  $g[n, m]$  under the neighborhood influence, determined by the kernel elements  $h$ , given as

$$g[n, m] = (f * h)[n, m] = \sum_{k, \ell} f[k, \ell] h[n-k, m-\ell] \quad (5a)$$

$$h = \frac{1}{k} \begin{bmatrix} 1 & 1 & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & k \end{bmatrix}_{k \times k}. \quad (5b)$$

The change map produced in AR(1) after averaging generates rows strongly correlated to a potential anomaly of interest. Using an empirical cumulative distribution function (ECDF), defined by  $\hat{F}(\cdot)$ , we can identify such anomalies of interest in the  $p$ -quantiles. This process is obtained from  $y_1, \dots, y_n$  i.i.d samples of  $y$ , making

$$\hat{F}(y) = \frac{1}{n} \sum_{i=1}^n I(x_i \leq x) \quad (3a)$$

$$y(p) = \hat{F}^{-1}(p), \quad (3b)$$

where  $I(\cdot)$  is the indicator function. The value  $y(p)$  establishes the threshold using in AR(1) recursion when  $x_t[n, m]$  has an anomaly of interest against the lower clutter anomalies due  $y_{t-1}[n, m]$ .

#### 4. IMPLEMENTATION MODEL USING A SYNCHRONOUS MoC

Synchronous systems are composed of synchronized parallel components, running in successive computation steps. In each step, all components perform some quantum of computation.<sup>22</sup> Synchronous computation models are particularly adequate for hardware, real-time systems and streaming systems. The Formal System Design (ForSyDe) methodology (for more information visit the home page <https://forsyde.github.io/>) is based on a functional language and the theory of models of computation and provides libraries focusing on functionality modelling and designing of heterogeneous embedded systems.

Particularly, the ForSyDe-Shallow library supports the synchronous MoC, using signals and process constructors for combinational and sequential processes to develop a well-defined structured formal model. ForSyDe also supports data-parallel skeletons, enabling the efficient modelling of parallelism, which can later be exploited when targeting a multicore or parallel computer architecture.

Fig. 3 shows the equivalent change detection algorithm using the synchronous MoC in ForSyDe. The proposed model implements the AR(1) recursion as a module function, where the input signals are defined for the bitemporal dataset, partitioned into feasible sub-blocks<sup>†</sup>. Each process is created using a process constructor (like `mapSY` or `zipWithSY`) and a function argument, and performs a specific processing step consuming the input values during an event cycle.

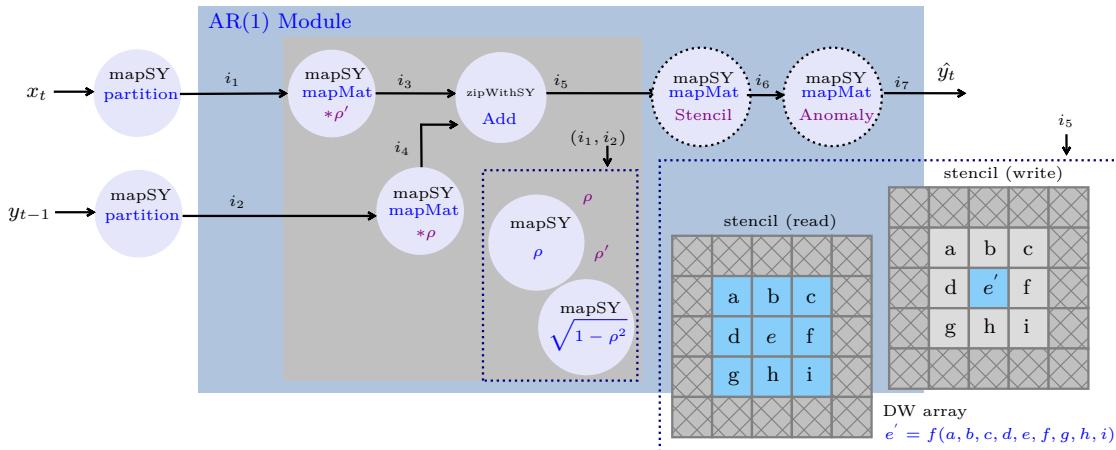


Figure 3: 2D-AR(1) filter designed in ForSyDe using the synchronous MoC.

In AR(1) module, the spatial filtering using a stencil function demands a quadratic complexity with the kernel size used in the convolution with the input signal  $i_5$ . The anomaly detection function performs the processing in a vectored shape, where the ECDF is achieved in linear complexity with the  $i_6$  size.

Due to its computational complexity, the spatial filter is separated in an optimized memory and distributed parallel processing using a Delayed Windowed DW array structure from Massiv library (It is available on line at <https://hackage.haskell.org/package/massiv>). DW array do not exist in memory and its contents describes the average function performed by kernel convolution, using an offset index and a multiplier to reproduce this operation. Instead of replacing elements in a mutable array, the DW array just map over the incoming elements and keep some elements while replacing others. Furthermore, the constructor `Par` describes a parallel computation distributed over the available cores.

The full process composes an elementary change detection model implemented in a synchronous MoC applied for sequential signals obtained by chunks from the full size images. The size of partitioning  $N$  and the size of

<sup>†</sup>Considering memory and CPU clock requirements.

spatial filtering kernel  $k$  are constrained by budget time, hardware architecture capacity, and sample sizes to provide sufficient statistics.

## 5. EXPERIMENTAL RESULTS

In the experiments, we applied the change detection algorithm over samples of CARABAS-II dataset, characterized in Sec. 2. For such a dataset, the GLRT approach based on the Gaussian clutter model<sup>16</sup> is treated as the lower bound regarding additional complexity involving CFAR filtering and its low ROC performance, being used as a reference for more recent results over the same dataset.

Previously, extracting a partition of  $x_t[n, n]$  and  $y_{t-1}[n, n]$  given by  $\{n = 1, \dots, N; n = 1, \dots, N\}$ , with  $N = 115$ , two scenarios are evaluated with and without the average filter: (1) presence of anomaly due the targets in  $x_t[n, n]$  and  $y_{t-1}[n, n]$  contents only clutter (target-clutter); and (2) both  $x_t[n, n]$  and  $y_{t-1}[n, n]$  content only clutter (clutter-clutter). These scenarios produce the outputs of AR(1) shown in Fig. 4a. As noted, the spatial filtering using the averaging improves the signal separation between change due the targets and the correspond clutter changes.

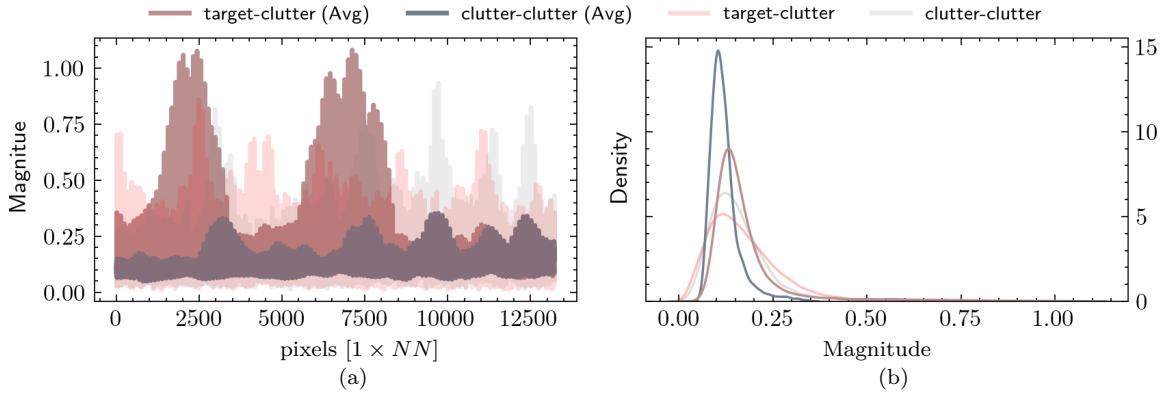


Figure 4: AR(1) outputs under spatial filtering effect for  $N = 115$  on selected areas: (a) vectored pixel-level response, and (b) estimated pdf.

Fig. 4b shows this effect in the correspondence positive pdf estimated from AR(1) output, where a proper quantile function can easily defines the threshold for outliers in terms of targets presence, as shown in Fig. 5.

For experiments, the setup with  $N = 40$ , and  $k = 9$ , produces a balanced compromises between hardware performance and testing response when performed in a general-purpose hardware architecture<sup>†</sup>. This same setup was replied in GLRT algorithm for the equivalent convolution windows, as used in Ref. 16.

### 5.1 Detection Performance

The detection performance was evaluated in terms of ROC, where the produced change maps on both tests were subjected to morphological operations<sup>‡</sup> for target classification, and defining the target size as  $12 \times 12$  m. Fig. 6 shows the ROC performance over the dataset under selected thresholds. As results, the maximum power ( $P_D$ ) is achieved in AR(1) between 0.95 and 0.99 with  $\log_{10}(\text{FAR})$  of  $-2.4$  and  $-2$ , respectively, which represents an approximate gain of 2.4 dB for the maximum  $P_D$  achieved in the lower bound (GLRT).

The signal separation is observed in the group of selected mission scenarios in Fig. 8, where the AR(1) test achieves a considerable decrease of false alarm in the  $P_D$  convergence. On the other hand, the GLRT achieves higher  $P_D$  under explosive false alarm costs. Since the AR(1) model is an elementary time series recursion, other higher-order approaches or ARMA structures potentially increase the complexity of computing to obtain detection gain, requiring parallelized distribution processing to obey runtime restrictions and power consumption.

<sup>†</sup>The CPU processor Intel Core i9-12900KF, 16-Core, 24-threads, clock of 3.2 GHz and 32 GB DDR5 of memory.

<sup>‡</sup>A kernel for structure element of  $3 \times 3$  applied in 1 erosion followed by 1 dilatation, then a kernel of  $5 \times 5$  applied to a dilatation.

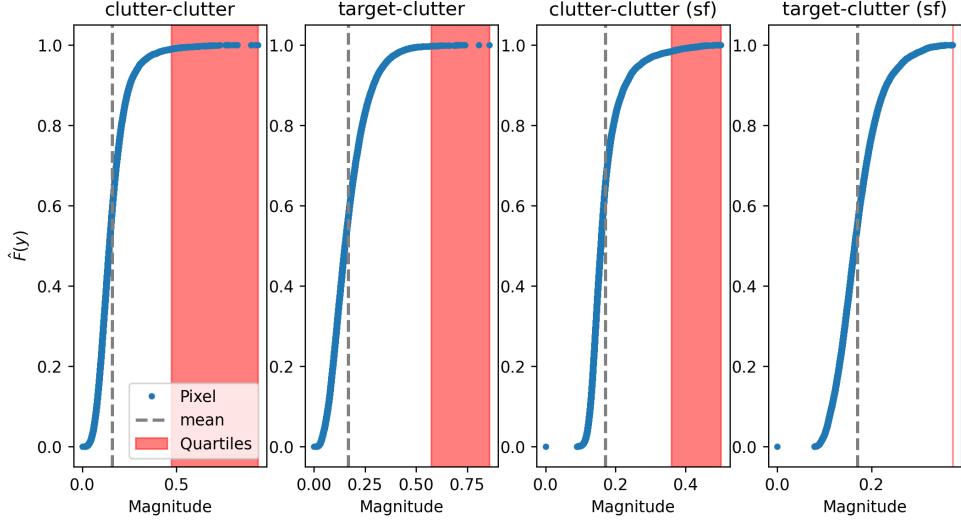


Figure 5: The threshold function of ECDF produced by AR(1) outputs under spatial filtering (sf) effect.

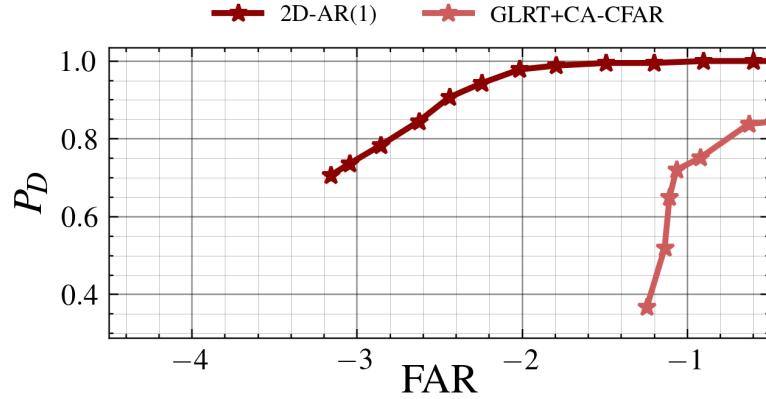


Figure 6: Performance ROC of the proposed 2D-AR(1) in comparison to the GLRT for Gaussian clutter over the full bitemporal dataset of CARABAS-II SAR UWB system.

## 5.2 Complexity and Energy Analysis

The hardware measurements use the Linux-based monitor tools `htop` and `powerstat`, which collect 1 second samples of CPU usage, memory access, and the power consumption over cores. In experiments, the MoC for AR(1) performs a parallel processing of the inputs and the spacial filtering uses, in average, 4 cores. In GLRT, the CA-CFAR Filtering uses 8 parallel processing over 8 CPU threads.

The windowing for spatial analysis in both tests uses the same dimensions, although with different strategies. However, the GLRT has the additional spatial filtering using a  $K = 30$  to provide a dynamic threshold. Tab. 1 summarizes the complexity involving the tests under analysis, where the main matrix operations and their runtime are computed.

The collection of hardware measurements is presented in Fig. 7, where the data points are exclusively due to the stages of the test referenced in Tab. 1. The arithmetic throughput, in terms of floating-point operations per second (FLOPS), is computed through each bitemporal pair execution. Fig. 7a shows that the runtime costs to execute one pair in the GLRT algorithm is about the time to process all the 24 pairs in AR(1). This is directly reflected in the power consumption, shown in Fig. 7b, where the GLRT demands  $\approx 65\%$  more energy to compute only 4% of the dataset (1 entire pair from 24).

Table 1: Complexity of change detection tests on dataset.

<b>Instance</b>	<b>GLRT</b>		<b>2D-AR(1)</b>	
	GLRT	CFAR	sf	AR(1)+ECDF
Complexity	$\mathcal{O}(N^2)$	$\mathcal{O}(N^2K^2)$	$\mathcal{O}(N^2k^2)$	$\mathcal{O}(N^2 \log N^2)$
Runtime (sec)	8.47	242.60	10.65	7.00

NOTE: sf = spatial filtering stage with averaging filter.

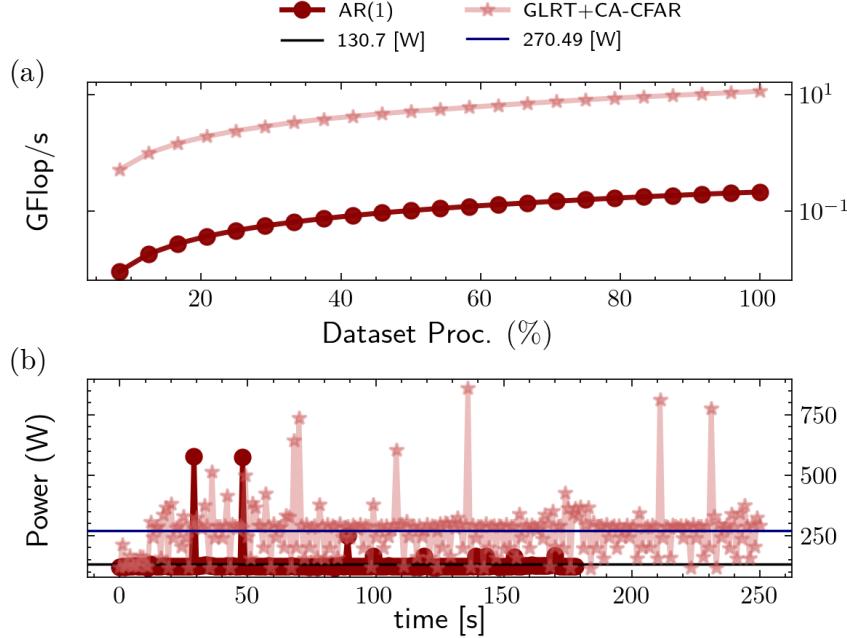


Figure 7: Comparative HW measurements with the GLRT and 2D-AR(1): (a) computational complexity on the entire dataset, and (b) power consumption in a slot time to compute 1 entire pair in GLRT.

## 6. CONCLUSIONS

The proposed change detection algorithm is designed in an elementary synchronous model of computation, where the matrices operations and parallelism scheduling are properly computed in the signal events. Even with one-lag for prediction, this model improves the detection performance compared with a particular statistical magnitude model with adaptive threshold filter. In addition, the hardware efforts, considering a specific multiprocessor, presented a significant reduction of power consumption and the computational throughput demand, leading to the feasible time for large dataset analysis. Finally, its extension for a Markov chain in multitemporal ( $n$ -lags) analysis is applicable to improve the detection performance considering relevant statistical dependences and keeping the potential deterministic parallelism over the synchronous MoC structure.

## ACKNOWLEDGMENTS

This study was financed by Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq, Centro de Pesquisa e Inovação Sueco-Brasileiro – CISB, and Saab AB. Grant 200159/2023-2. The authors also would like to thank the FOI for the support related to the baseline algorithm.

## APPENDIX A. SELECTED DATASET PERFORMANCE

The thresholds in GLRT and 2D-AR(1) algorithms are adjusted to maximize the target detection.

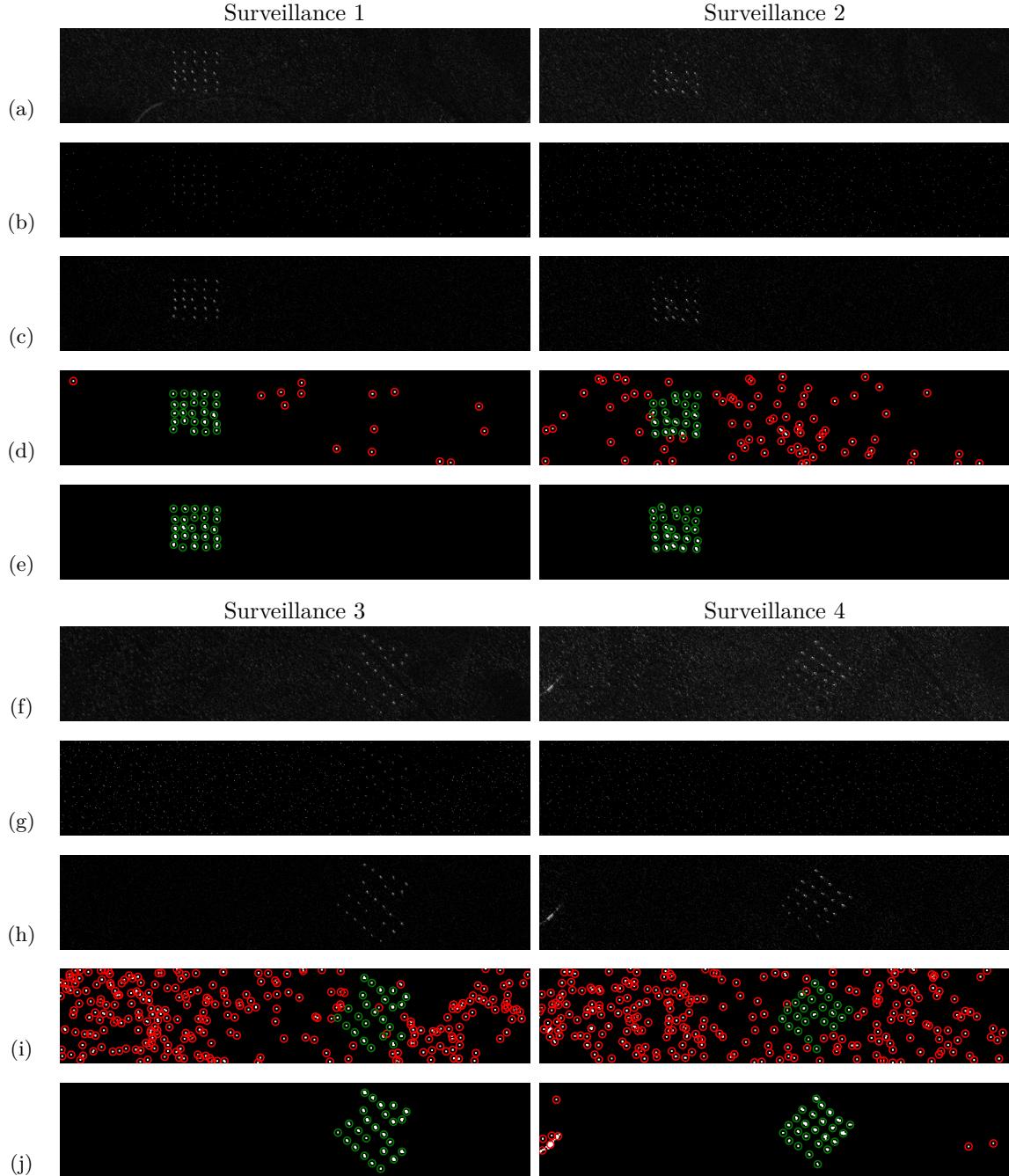


Figure 8: Test results over a sample of CARABAS-II dataset: (a) and (f) are 4 selected Surveillance images; (b) and (g) GLRT+CA-CFAR filtering output; (c) and (h) the output of 2D-AR(1) filtering; (d) and (i) classification over GLRT+CA-CFAR output; and (e) and (j), classification over 2D-AR(1) output.

## REFERENCES

- [1] T. G. Bryant, G. B. Morse, L. Novak, and J. Henry, "Tactical radars for ground surveillance," *Digital Signal Processing* **12**(2), pp. 341–354, 2000.
- [2] G. Moser, J. Zerubia, and S. Serpico, "SAR amplitude probability density function estimation based on a generalized Gaussian model," *IEEE Transactions on Image Processing* **15**(6), pp. 1429–1442, 2006.
- [3] J. Ingla and G. Mercier, "A new statistical similarity measure for change detection in multitemporal SAR images and its extension to multiscale change analysis," *IEEE Transactions on Geoscience and Remote Sensing* **45**(5), pp. 1432–1445, 2007.
- [4] M. Sagrillo, R. R. Guerra, F. M. Bayer, and R. Machado, "The  $\mathcal{L}_A$  distribution: An approximation of the  $\mathcal{G}_A^0$  distribution for amplitude SAR image modeling," *IEEE Transactions on Geoscience and Remote Sensing* **61**, pp. 1–8, 2023.
- [5] V. T. Vu, N. R. Gomes, M. I. Pettersson, P. Dammert, and H. Hellsten, "Bivariate gamma distribution for wavelength-resolution SAR change detection," *IEEE Transactions on Geoscience and Remote Sensing* **57**(1), pp. 473–481, 2019.
- [6] M. A. Richards, *Fundamentals of Radar Signal Processing*, McGraw-Hill Professional, Reading, MA, 2005.
- [7] L. M. Ulander, M. Lundberg, W. Pierson, and A. Gustavsson, "Change detection for low-frequency sar ground surveillance," *IEE Proceedings-Radar, Sonar and Navigation* **152**(6), pp. 413–420, 2005.
- [8] M. E. Davis, *Foliage Penetration Radar: Detection and characterisation of objects under trees*, Institution of Engineering and Technology, Reading, MA, 2011.
- [9] D. E. Wahl, D. A. Yocky, C. V. Jakowitz, and K. M. Simonson, "A new maximum-likelihood change estimator for two-pass SAR coherent change detection," *IEEE Transactions on Geoscience and Remote Sensing* **54**(4), pp. 2460–2469, 2016.
- [10] G. Quin, B. Pinel-Puysségur, J.-M. Nicolas, and P. Loreaux, "MIMOSA: An automatic change detection method for SAR time series," *IEEE Transactions on Geoscience and Remote Sensing* **52**(9), pp. 5349–5363, 2014.
- [11] P. Johannesson, K. Podgórski, I. Rychlik, and N. Shariati, "AR(1) time series with autoregressive gamma variance for road topography modeling," *Probabilistic Engineering Mechanics* **43**, pp. 106–116, 2016.
- [12] B. G. Palm, D. I. Alves, M. I. Pettersson, V. T. Vu, R. Machado, R. J. Cintra, F. M. Bayer, P. Dammert, and H. Hellsten, "Wavelength-resolution SAR ground scene prediction based on image stack," *Sensors* **20**(7), 2020.
- [13] M. Davis and R. Vinter, *Stochastic Modelling and Control*, Monographs on Statistics and Applied Probability, Springer Netherlands, 1985.
- [14] E. Lee and A. Sangiovanni-Vincentelli, "A framework for comparing models of computation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **17**(12), pp. 1217–1229, 1998.
- [15] I. Sander, A. Jantsch, and S.-H. Attarzadeh-Niaki, *ForSyDe: System Design Using a Functional Language and Models of Computation*, pp. 99–140. Springer Netherlands, Dordrecht, 2017.
- [16] M. Lundberg, L. Ulander, W. Pierson, and A. Gustavsson, "A challenge problem for detection of targets in foliage," in *Algorithms for Synthetic Aperture Radar Imagery XIII*, p. 62370K, (Orlando, United States), May 2006.
- [17] S. M. Kay, *Statistical signal processing: Detection theory*, Prentice Hall, Reading, MA, 1998.
- [18] A. Frery, H.-J. Muller, C. Yanasse, and S. Sant'Anna, "A model for extremely heterogeneous clutter," *IEEE Transactions on Geoscience and Remote Sensing* **35**(3), pp. 648–659, 1997.
- [19] W. S. Chen and I. S. Reed, "A new cfar detection test for radar," *Digital Signal Processing* **1**(4), pp. 198–214, 1991.
- [20] N. R. Gomes, P. Dammert, M. I. Pettersson, V. T. Vu, and H. Hellsten, "Comparison of the Rayleigh and k-distributions for application in incoherent change detection," *IEEE Geoscience and Remote Sensing Letters* **16**(5), pp. 756–760, 2019.
- [21] G. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, Holden-Day, 1976.
- [22] A. Jantsch, "chapter four - the synchronous model of computation," in *Modeling Embedded Systems and SoC's*, A. Jantsch, ed., *Systems on Silicon*, pp. 181–221, Morgan Kaufmann, San Francisco, 2003.