Adaptive Spatial-Temporal Filtering Methods for Clutter Removal and Target Tracking

Alexander G. Tartakovsky, *Senior Member, IEEE* and James Brown

***Abstract*— In space-based infrared ballistic missile defense sensor systems, cluttered backgrounds are typically much more intense than the equivalent sensor noise or the targets being detected. Therefore, the development of efficient clutter removal and target preservation/enhancement algorithms is of crucial importance. To meet customer requirements, the advanced clutter rejection algorithms should provide more than 20 dB improvement in detection sensitivity. We propose an adaptive parametric spatial-temporal filtering technique together with the jitter compensation (scene stabilization). The results of simulations and processing of real data show that the developed adaptive spatial-temporal clutter suppression algorithms allow for efficient clutter rejection in all tested situations. Proposed algorithms completely remove heavy clutter in the presence of substantial jitter and do not require expensive sub-pixel jitter stabilizers. In contrast, spatial-only filters and temporal differencing methods can be used only for weak and relatively correlated clutter. A stand-alone simulator was developed to demonstrate capabilities and performance of various algorithmic approaches. Simulations model various geometries, resolutions, illuminations and meteorological conditions for space-based IR staring sensor systems.**

***Index Terms*— Infrared sensors, solar cloud clutter, spatial-temporal image processing, clutter removal, scene stabilization, jitter compensation, detection, tracking.**

I. INTRODUCTION

THE problem of efficient clutter rejection is a challenge for Space-Based Infrared (IR) and Space Tracking and Surveillance System (STSS) sensors with chaotically vibrating lines-of-sight (LOS) that have to provide early detection and mid-course tracking of missile launches in the presence of highly-structured cloud backgrounds. In such systems, the intensities of cluttered backgrounds due to solar scattering by clouds, aerosols and earth surface (ground, sea, etc.) or by IR airglow emissions are typically dozens and even hundreds of times greater than either sensor noise or the intensities of the targets that are to be detected and then tracked. As a result, reliable target detection and subsequent tracking is impossible without clutter rejection to, or even below, the level of sensor noise.

Most existing clutter rejection technologies for unstabilized or mechanically stabilized platforms rely on spatial-only or simple image differencing methods. There are many spatial filtering methods [2]-[5],[12], [13], [26] such as matrix filters of various orders, fast Fourier filters, wavelet filters, parametric and nonparametric (kernel) regression filters, rank-order filters, etc. Median filters and, more generally, rank-order filters are robust and particularly useful for extracting smooth objects with sharp edges [26]. However, the results of our study presented below show that even the best spatial-only filters are not efficient enough, especially in unfavorable conditions that are typical for applications of interest. Moreover, these filters cannot be combined with temporal processing in cases where clutter is non-stationary in time due to platform vibrations (jitter), which is always the case. A similar conclusion holds for an industry standard differencing method, as can be seen from Section IV.

To be more specific, MTI (Moving Target Indicator) algorithms have attempted to exploit the relative motion between the target and the background. The classic differencing paradigm has been to register the backgrounds in two successive frames and then subtract one image from the other. In theory, the background in the difference image will be reduced to twice the noise level of the sensor while the level of the target above the background will remain unchanged. The result is an increase in the S(C+N)R (signal to clutter +noise ratio) of the target. The success of this approach is highly dependent upon, and extremely sensitive to, the accurate registration of the two images. The ability to achieve the requisite registration accuracy is frustrated by the sensor platform motion.

Theoretically, in order to interpolate or reconstruct the intensity of an image at points not directly on the pixel or sample grid, the image must be sampled at 2-3 times above the highest frequency of the image being sampled. This requirement places a significant constraint upon the design and performance of the system and impacts the timeline and reduces the tracker update rate. Therefore, differencing techniques require autonomous, very accurate frame registration methods. In principle sufficiently accurate frame registration and resampling are possible, as was shown by Lucke & Stocker [16] and Schaum [18], [19]. However, experience shows that even with all these sacrifices and an excellent registration algorithm, there is an unacceptable amount of clutter leakage past the differencing operation. In space-based IR applications this will be exacerbated by the movement of clouds and other natural phenomena.

In this paper, we argue that the level of clutter suppression required for reliable detection and tracking (20 dB and higher) can be achieved only by implementing *spatial-temporal* and/or *space-temporal-spectral* (i.e., multispectral) filtering rather than spatial filtering alone. Note that in order to make temporal

A.G. Tartakovsky is with the Department of Mathematics, University of Southern California, 3620 S. Vermont Avenue, KAP-108, Los Angeles, CA 90089-2532, USA and with ADSANTEC, Torrance, CA. E-mail: tartakov@usc.edu; WWW: http://www.usc.edu/dept/LAS/CAMS/usr/facmemb/tartakov (corresponding author).

James Brown is with Air Force Research Laboratory/VSBYB, 29 Randolph Road, Hanscom AFB, MA 01731, USA.

Manuscript received October 31, 2006; revised March 28, 2007

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
SEP 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Adaptive Spatial-Temporal Filtering Methods for Clutter Removal and Target Tracking	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Southern California,Department of Mathematics,Los Angeles,CA,90089**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**14**	

processing and multispectral fusion efficient, clutter rejection algorithms must be supplemented by very accurate *jitter estimation and scene stabilization techniques* that compensate for platform vibrations and eliminate residual frame misalignment. Our image registration/stabilization techniques are entirely different from those previously used [1], [16], [17], [18], [19]. Stabilization is performed iteratively in the course of clutter rejection, and the corresponding stabilization algorithm is a part of the clutter rejection architecture. We show that this novel approach is extremely efficient in applications of interest. Our spatial-temporal iterative method allows not only for very accurate interpolation but also for very accurate image reconstruction and, therefore, clutter suppression in a wide spectrum of conditions, including highly structured scenes characteristic for IR solar cloud clutter.

To address super-stabilization and super-rejection challenges, we further develop the idea that has been proposed by Tartakovsky and Blažek [23], which allows us to avoid the use of expensive equipment for super-stabilization. The idea is to use clutter itself for stabilization. Indeed, since clutter is typically highly correlated in time and changes slowly during certain periods of time, i.e., quasi-stationary (depending on weather conditions, season, etc.), substantial temporal changes in these intervals occur only due to jitter. Since clutter is normally much more intense than sensor noise, it can be effectively used for instability estimation and compensation. When the registration problem is solved, i.e., the scene is stabilized, we can build an effective spatial-temporal scheme for clutter rejection, in which case, clutter can be suppressed to the level of sensor noise or even lower. We develop novel statistical parametric and nonparametric spatial-temporal-spectral techniques for clutter suppression (CLS) and scene stabilization. The corresponding stabilization and CLS algorithms are robust since they are invariant to prior uncertainty with respect to clutter statistical properties and adaptive with respect to clutter variability. An important feature of the developed CLS algorithms is that they allow for reducing the cost of expensive mechanical and electronic stabilizers.

Experiments and simulations presented below show that the proposed algorithm gives a substantial gain compared to the best existing spatial techniques as well as to the industry standard temporal differencing method in certain difficult scenarios that are typical for applications of interest. The results of simulations also show that any particular clutter rejection filter is not uniformly optimal for all possible conditions. Since environmental conditions may change quite dramatically, it is important to develop a bank of CLS filters along with a procedure of automatic selection of the best filter and its parameters for specifically encountered conditions. To this end, a reconfigurable CLS system that includes a bank of filters was developed. This system utilizes auto-tuning and auto-selection procedures for optimal configuration, reducing susceptibility to sensor vibrations and to changes in environmental conditions. These procedures use an overall system quality metric that is a function of the current sub-system performance indices, including expected SNR, clutter severity metric, false alarm rate, and tracking quality.

This paper is organized as follows. In Section II, we outline the signal-data processing system being developed. In Section III-A, we formulate the problem of clutter rejection and describe basic assumptions and constraints. In Section III-B, we develop the CLS algorithms that include the jitter estimation and compensation algorithm. In Section IV, the results of simulations are presented. These results allow us to evaluate the performance of the developed algorithms and to make important practical conclusions that are outlined in Section V.

II. The Developed System

In this research, we focus on developing algorithms and software for adaptive clutter suppression, target detection and multiple target tracking for a variety of observation conditions; tuning and optimization of these algorithms for particular scenarios; and testing and validation through synthetic simulations and processing of real data. The primary goal is to develop a viable prototype of the multiple target tracking system that includes a reconfigurable, adaptive clutter suppression system that can be tested using a built-in simulator which mimics real environments. The developed system and corresponding software tools have the following functionalities and capabilities:

1) Possibility of working with external data sets of interest for users (e.g., real data) which are presented in a specific standard format.
2) Built-in generator of image sequences with moving point illumination sources (targets), background clutter due to cloud cover, jitter due to platform vibrations, and sensor noise.
3) A bank of clutter suppression filters with a reconfigurable architecture and ability to compensate for strong signals from bright targets and outliers.
4) Auto-tuning and auto-selection algorithms that allow for automatic selection of the optimal filter from the bank for current conditions.
5) In-frame detection algorithms with constant false alarm rate (stabilization of false alarms).
6) Multitarget tracking algorithms, in particular:
    a) Change-point detection-based track initiation algorithms
    b) Identification of detections as belonging to existing tracks
    c) Forming new tracks and deletion of false tracks
    d) Change-point detection-based track termination
7) Preliminary target track classification in 2D space.
8) Performance evaluation (probability of detection, false alarm rate, tracking accuracy) for synthetic data (physics-based models) and real data.
9) Graphical user interface (GUI) for visualization of the results of processing and for input data and parameters.

A general block-diagram of the system with the corresponding data/signal processing flow is shown in Fig. 1.

The strong signal estimation block shown in Fig. 1 detects bright targets in the raw image $\mathbf{Z}_n$ and estimates its parameters (e.g., intensities and positions). The signal estimates $\hat{\mathbf{s}}_n$ are subtracted from the raw data to form the statistic $\widehat{\mathbf{Z}}_n =$

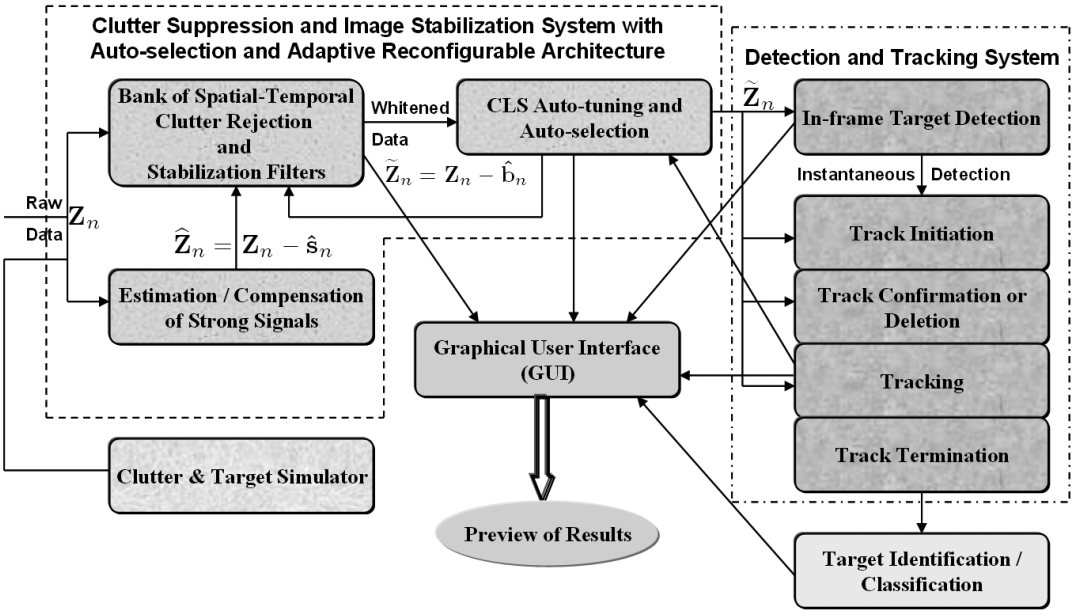


Fig. 1. Block-diagram of the CLS-detection-tracking system

$\mathbf{Z}_n - \hat{\mathbf{s}}_n$. Spatial-temporal CLS filters process the sequence of statistics $\widehat{\mathbf{Z}}_n, \ldots, \widehat{\mathbf{Z}}_{n-m+1}$ in the time window m to estimate the background. The resulting estimate $\hat{\mathbf{b}}_n$ is subtracted from $\mathbf{Z}_n$ to form the residuals $\widetilde{\mathbf{Z}}_n = \mathbf{Z}_n - \hat{\mathbf{b}}_n$ which are the output of the CLS filter. The data at the outputs of all filters in the bank are analyzed by the auto-selection block to choose the best configuration for the current conditions. The feedback from the target detection and tracking blocks is used to provide more comprehensive filter selection and tuning based on the evaluated detection and tracking performance (e.g., based on the $Q-$factor; see Section III-A).

The processed data from the chosen CLS filter come to the Target Detection Block. The output of this block are the instantaneous detections and their attributes, such as the estimates of the target positions and intensities. The in-frame detection algorithm realizes either a generalized likelihood ratio hypothesis test or its modification under the conditions of prior uncertainty. See Ref [22] for further details.

The processed data from the output of the Target Detection Block come to the tracking subsystem (Track Initiation Block) along with the data $\widetilde{\mathbf{Z}}_n$ from the output of the Auto-selection Block. The multitarget tracking scheme (including track management) is more or less standard in terms of the sequence of operations (see, e.g., [6], [8], [9]). However, the following important innovations are used: (1) global data association (optimal for association of all detections but not locally optimal for a particular detection); (2) adaptive selection of the polynomial power for current conditions through introduction of virtual tracks in the polynomial filter; and (3) optimal and/or quasi-optimal procedures for track initiation and termination based on change-point detection methods.

The track initiation algorithm is based on sequential change-point detection methods [7], [20], [21], [25]. This algorithm detects and estimates the moment of target appearance with a minimal detection delay for a given false alarm rate [15], [22], [24].

The initialized tracks are transferred to the Track Confirmation/Deletion Block. If the track is not confirmed, it is deleted. The confirmed tracks are further processed in the tracking block.

The tracking process is interrupted (tracks are terminated) in the Track Termination Block according to sequential algorithms that detect target disappearance with a minimal detection delay for a given false termination rate. Further details may be found in [15], [22], [24].

Finally, the parameters of the confirmed and terminated tracks go to the Target Classification Block where the targets are classified as belonging to the class of interest versus "everything else" in 2D space. We stress that classification is done in 2D space, not in 3D space.

The GUI is being used as a part of a testbed to test the system performance for particular scenarios and to visualize the results of data processing. See Section IV and Fig. 3 for further details.

Comprehensive target detection, tracking, and classification with their full set of problems are out of the scope of this paper. The major problem that is of interest in the present paper is the development of efficient spatial-temporal algorithms for clutter rejection. This problem is addressed in the subsequent sections in detail.

III. Parametric Clutter Suppression Algorithms

The developed baseline CLS technique is based on a multi-parametric approximation of clutter (regression-type modeling) that leads to an adaptive spatial-temporal filter. The "coefficients" of the filter are calculated adaptively according

to the minimum distance algorithm. The adaptive spatial-temporal filter allows one to suppress any background, regardless of its spatial variation. It not only whitens the data but also corrects all translational distortions.[1] The results of an experimental study presented in Section IV show that the proposed algorithm gives a substantial gain compared to the best existing spatial techniques as well as to the industry standard temporal differencing method.

A. Problem Formulation, Assumptions, and Operating Characteristics

We now turn to a formal problem setting. It is assumed that we observe a sequence of 2D $N_x \times N_y$ images (frames) that are registered by an IR sensor:

$$\begin{aligned} Z_n(\mathbf{r}_{ij}) = \sum_{k=1}^{K_n} I_n(k) S(\mathbf{r}_{ij} - \mathbf{r}_n(k) - \boldsymbol{\delta}_n) \\ + b_n(\mathbf{r}_{ij} - \boldsymbol{\delta}_n) + \xi_n(\mathbf{r}_{ij}), \quad n = 1, 2, \dots, \end{aligned} \tag{1}$$

where $\xi_n(\mathbf{r}_{ij})$ is sensor noise; $b(\mathbf{r}_{ij})$ is clutter (background); $I_n(k)S(\mathbf{r}_{ij}-\mathbf{r}_n(k))$ is a signal from the k-th target with spatial coordinates $\mathbf{r}_n(k) = (X_n(k), Y_n(k))$ and maximal intensity $I_n(k)$; $S(\mathbf{r}_{ij})$ is the target signature related to the sensor's point spread function (PSF); K_n is an unknown number of targets in the n-th frame; $\boldsymbol{\delta}_n = (\delta_x(n), \delta_y(n))$ is an unknown 2D shift due to the jitter (LOS vibrations); $\mathbf{r}_{ij} = (x_i, y_j)$ is the pixel in the plain image with coordinates (x_i, y_j), $i = 1, \dots, N_x$, $j = 1, \dots, N_y$.

In the following we denote as (Δ_x, Δ_y) effective pixel sizes in x and y directions. For the convenience's sake and without loss of generality we will assume that $\Delta_x = \Delta_y = 1$, which can always be done by normalization.

We will also use boldface for a vector/matrix notation, e.g.,

$$\begin{aligned} \mathbf{Z}_n &= \{Z_n(\mathbf{r}_{ij}), i = 1, \dots, N_x, j = 1, \dots, N_y\}, \\ \boldsymbol{\xi}_n &= \{\xi_n(\mathbf{r}_{ij}), i = 1, \dots, N_x, j = 1, \dots, N_y\}, \\ \mathbf{b}_n &= \{b_n(\mathbf{r}_{ij} - \boldsymbol{\delta}_n), i = 1, \dots, N_x, j = 1, \dots, N_y\}. \end{aligned}$$

The goal is to build a spatial-temporal filter that rejects clutter (suppresses it to the level of noise or below) and simultaneously compensates for the jitter (stabilizes the scene), while preserving signals from targets as much as possible.

We will restrict ourselves to the class of *residual-type* filtering algorithms of the form

$$\widetilde{Z}_n(\mathbf{r}_{ij}) = Z_n(\mathbf{r}_{ij}) - \hat{b}_n(\mathbf{r}_{ij}), \tag{2}$$

where $\widetilde{Z}_n(\mathbf{r}_{ij})$ is the output of the filter and $\hat{b}_n(\mathbf{r}_{ij})$ is an estimate of clutter $b_n(\mathbf{r}_{ij})$ in the pixel (i, j) of the current n-th frame based on the previous frames $Z_{n-m+1}(\mathbf{r}_{ij}), \dots, Z_n(\mathbf{r}_{ij})$ in a sliding window of the size m ($n \geq m$). The filtering procedure will be considered *ideal* if the clutter component in (1) is completely suppressed and the signal components $I_n(k)S(\mathbf{r}_{ij} - \mathbf{r}_n(k) - \boldsymbol{\delta}_n)$ are completely preserved (no degradation). For brevity the residual frame $\widetilde{Z}_n(\mathbf{r}_{ij})$ will be sometimes called "whitened."

Suppose that in the time interval m (frames) the function $b_n(\mathbf{r})$ is a slowly varying function, so that one can neglect changes of this function due to physical causes such as wind, illumination conditions, temperature, convection, etc. In practice this condition requires a proper choice of the frame rate. Under this assumption, the clutter function $b_n(\mathbf{r}_{ij})$ changes in time in each pixel only because of the sensor (LOS) vibrations. Thus, the condition of local stationarity is invoked. It should be noted, however, that changes that occur in time intervals bigger than m are treated by the algorithms and so global non-stationarity is allowed.

In real conditions statistical properties of neither clutter $b_n(\mathbf{r})$ nor vibrations are known. The development of the CLS algorithms will be performed under almost complete prior uncertainty with respect to the properties of clutter and jitter. This guarantees robustness of the algorithms with respect to variations of the statistical properties of clutter and jitter over wide ranges. More specifically, we will suppose that translations $\boldsymbol{\delta}_n$ are arbitrary unknown variables bounded by a maximum possible amplitude of the sensor vibrations and $b_n(\mathbf{r})$ is an arbitrary unknown non-negative function with a bounded spatial band.

Specifically, the following assumptions on noise, clutter, and jitter are used throughout the paper:

1) Noise $\xi_n(\mathbf{r})$ is uncorrelated in time and space with mean zero and variance σ_N^2.
2) Clutter $b_n(\mathbf{r})$ is an arbitrary, unknown function of $\mathbf{r}$ and is a slowly-changing function of n in the following sense: there exists an interval m such that
$$|b_{n+m}(\mathbf{r}) - b_n(\mathbf{r})| \leq \sigma_N \text{ for all } n \text{ and } \mathbf{r}. \tag{3}$$
3) Vibrations of the platform can be very fast such that the jitter $\boldsymbol{\delta}_n$ can change abruptly from frame-to-frame, but these changes do not exceed some known value $\boldsymbol{\delta}_{\max}$ that depends on the mechanical stabilizers and other conditions.

It is worth noting that in a variety of airborne and low-earth orbit satellite scenarios the shift $\boldsymbol{\delta}_n(\mathbf{r}_{ij})$ depends on the pixel $\mathbf{r}_{ij}$ due to rotations and parallax. However, in the geostationary staring IR sensor scenario that is of major interest in this paper, rotations can be neglected, and the value of $\boldsymbol{\delta}_n(\mathbf{r}) = \boldsymbol{\delta}_n$ is just the parallel shift that does not depend on $\mathbf{r}$.

The value of m defines the length of the interval in which clutter does not change substantially (slowly varying). This value is estimated experimentally and corresponds to the length of the temporal window where temporal filtering may be performed. In other words, spatial-temporal filtering for clutter rejection is based on the data $(\mathbf{Z}_{n-m+1}, \dots, \mathbf{Z}_n)$ in the sliding window m.

Assuming, without loss of generality, that there is only one target in the scene with maximal intensity $I_n = 1$, the output of the CLS filter can be written as follows

$$\widetilde{Z}_n(i, j) = \tilde{S}(i, j) + V_n(i, j), \tag{4}$$

where $\tilde{S}(i, j)$ is the signal from a target and $V_n(i, j)$ is residual clutter (plus sensor noise) after CLS filtering. Hereafter we

[1] In this paper, we are interested in geostationary staring IR sensors for which rotational distortions can be neglected. The algorithm can be modified to compensate for rotational and zooming distortions which is important in certain conditions, e.g., for low-earth orbit and airborne platforms.

write $Y(i,j)$ in place of $Y(\mathbf{r}_{ij}) = Y(x_i, y_j)$ when it is possible. Indeed, there is no loss of generality in assuming that there is only a single target, since the evaluation of the capabilities of the CLS algorithms can be performed independently for each point target based on the background suppression and signal preservation indices defined below in (5)–(7). For point and slightly extended targets, inter-occlusions can be neglected, in which case it suffices to evaluate the effective signal-to-clutter-plus-noise ratio for a single target. (This ratio is given by the Q-factor defined in (7), which is the most comprehensive index.) The situation, however, changes when detecting substantially extended targets when often inter-occlusions may occur. This latter case is not of interest for our applications.

Good filtering algorithms have to preserve the signal as much as possible (i.e., minimize the discrepancy between $S(i,j)$, the signal at the CLS filter input, and $\tilde{S}(i,j)$, the signal at its output) and, at the same time, suppress clutter as much as possible (the minimal level is of course limited by sensor noise).

The quality of the CLS filtering algorithms will be characterized by the following indices:

• ***G*-factor** that defines the relative value of the residual clutter standard deviation:

$$G = \sigma_{\text{out}}/\sigma_N, \tag{5}$$

where σ_N^2 is the variance of sensor noise and $\sigma_{\text{out}}^2 = \mathbf{E}[V_n(i,j)]^2$ is the variance of the output (filtered) frame. Obviously, $G \geq 1$ and the ideal value of this factor is equal to 1, which means complete clutter rejection.

• ***D*-factor** that defines the coefficient of signal degradation:

$$D = 1 - \frac{\sum_{i,j} S(i,j)\tilde{S}(i,j)}{\sqrt{\sum_{i,j} \tilde{S}^2(i,j) \sum_{i,j} S^2(i,j)}}. \tag{6}$$

In the ideal case, $D = 0$, which means precise signal recovery (no signal degradation).

• ***Q*-factor** that defines the relative value of the effective signal-to-clutter-plus-noise ratio (S(C+N)R):

$$Q = \frac{\left(\sum_{i,j} S(i,j)\tilde{S}(i,j)\right)^{1/2}/\sigma_{\text{out}}}{\left(\sum_{i,j} S^2(i,j)\right)^{1/2}/\sigma_N}. \tag{7}$$

Because of the signal degradation, $Q \leq \sigma_N/\sigma_{\text{out}} = 1/G$. The ideal value of the Q-factor is equal to 1, which means that clutter is suppressed completely without signal degradation (i.e., $\tilde{S}(i,j) = S(i,j)$ and $\sigma_{\text{out}} = \sigma_N$).

In experiments with synthetic targets presented in Section IV, the signal at the input $S(i,j)$ is known, while the values of the output signal $\tilde{S}(i,j)$ are estimated after detection along with the input and output background standard deviations. This allows us to evaluate both the D-factor and Q-factor in a straightforward manner.

B. Adaptive Spatial-Temporal CLS Filter

1) The Idea of the Parametric CLS Filter: We start with a description of the basic idea and a generic CLS algorithm for the class of parametric problems that involve parametric approximations of the function $b_n(\mathbf{r})$. This approach was first proposed by Tartakovsky and Blažek [23]. Recall that we do not use any assumptions on the statistical properties of clutter. All we assume is that clutter is an arbitrary function of spatial coordinates (may be a quite sharp function) and a slowly varying function of time in a certain time interval m.

Assume that the function $b_n(\mathbf{r})$ can be approximated by a parametric model

$$\tilde{b}_n^{\boldsymbol{\theta}}(\mathbf{r}) = \mathbf{f}^T(\mathbf{r})\boldsymbol{\theta}(n), \tag{8}$$

where $\boldsymbol{\theta}(n) = (\theta_1, \ldots, \theta_M)$ is a vector of unknown, slowly changing in the interval m parameters, $\mathbf{f}(\mathbf{r}) = (f_1(\mathbf{r}), \ldots, f_M(\mathbf{r}))$ is a known (chosen) vector-function, and T denotes a transpose. The model $\tilde{b}_n^{\boldsymbol{\theta}}(\mathbf{r})$ determines clutter in a "non-inertial" coordinate system that corresponds to a nominal translation value $\boldsymbol{\delta}$, which without loss of generality may be assumed 0. In what follows, for the sake of simplicity, the parameters $\boldsymbol{\theta}(n)$ are assumed constant, i.e., $\boldsymbol{\theta}(n) = \boldsymbol{\theta}$.

The choice of the "basis" function $\mathbf{f}(\mathbf{r})$ and its dimensionality M is determined by the allowed approximation error of real clutter that may have very high spatial variation.

Therefore, according to the model (8), it is proposed to use the following parametric approximation of clutter:

$$b_n(\mathbf{r}) \approx \sum_{k=1}^{M} \theta_k f_k(\mathbf{r}), \tag{9}$$

where θ_k are unknown parameters and $f_k(\mathbf{r})$ are given functions chosen from the best fitting criterion.

Let $\hat{\theta}_k(n)$ denote an estimate of θ_k based on the data $\mathbf{Z}_{n-m+1}, \ldots, \mathbf{Z}_n$ in the time window $[n-m+1, n]$ of the length m, where $\mathbf{Z}_k = \{Z_k(i,j)\}$, $i = 1, \ldots, N_x$, $j = 1, \ldots, N_y$. According to (9), for any shift $\boldsymbol{\delta}$, the prediction estimate of the background at time n has the form:

$$\hat{b}_n(\mathbf{r} - \boldsymbol{\delta}) = \sum_{k=1}^{M} \hat{\theta}_k(n) f_k(\mathbf{r} - \boldsymbol{\delta}). \tag{10}$$

Assume that the shifts $\boldsymbol{\delta}_s$ were somehow estimated for all frames $s = n-m+1, \ldots, n$ and let $\hat{\boldsymbol{\delta}}_s$ denote these estimates. Write

$$\mathcal{E}_n(\boldsymbol{\theta}, \{\hat{\boldsymbol{\delta}}_s\}) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{s=n-m+1}^{n} \Big(Z_s(\mathbf{r}_{ij}) - \sum_{k=1}^{M} \theta_k f_k(\mathbf{r}_{ij} - \hat{\boldsymbol{\delta}}_s) \Big)^2.$$

The estimate $\hat{\boldsymbol{\theta}}_n$ is found from the following optimality criterion:

$$\hat{\boldsymbol{\theta}}_n = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, \mathcal{E}_n(\boldsymbol{\theta}, \{\hat{\boldsymbol{\delta}}_s\}), \tag{11}$$

which is nothing but the least squares method.

The parameter estimation algorithm requires a reasonably accurate estimation of the shift $\boldsymbol{\delta}$, i.e., jitter compensation. This latter estimation/compensation can be done either by an independent jitter estimation algorithm or iteratively in the course of estimating parameters $\boldsymbol{\theta}$.

We now turn to the explanation of the iterative algorithm, which is usually the most accurate. To this end, suppose that the estimate $\hat{b}_{n-1}(\mathbf{r})$ is already obtained.

For jitter estimation in the n-th frame, we will use the minimum distance (MD) estimate $\hat{\boldsymbol{\delta}}_n = (\hat{\delta}_x(n), \hat{\delta}_y(n))$, which is the solution of the following *nonlinear* minimization problem:

$$\hat{\boldsymbol{\delta}}_n = \underset{|\boldsymbol{\delta}| \le \boldsymbol{\delta}_{\max}}{\operatorname{argmin}} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \Big(Z_n(\mathbf{r}_{ij}) - \hat{b}_{n-1}(\mathbf{r}_{ij} - \boldsymbol{\delta}) \Big)^2. \quad (12)$$

Note that we did not use any assumptions on noise distribution so far. In the case where the distribution of $\xi_n(\mathbf{r}_{ij})$ is Gaussian, this estimate corresponds to the maximum likelihood (ML) estimator. The alignment of frames according to the method (12) will be called the MD/ML-alignment.

2) A Generic Iterative CLS / Jitter Compensation Algorithm: We now describe the generic CLS filtering algorithm that uses an iterative procedure for jitter compensation (frame alignment) and the background estimation within the class of parametric models. This algorithm was first proposed in Tartakovsky and Blažek [23]. The abbreviations RCS and CCS are used for the reference coordinate system and corrected coordinate system, respectively.

The basic iterative CLS algorithm that includes jitter compensation has the following form:

1. Initialization. This step can be performed in various ways. Typically this step requires about m observations and the result of the initialization stage are the pilot estimates $\hat{\theta}_k(m)$, $(\hat{\boldsymbol{\delta}}_1, \ldots, \hat{\boldsymbol{\delta}}_m)$, and $\hat{b}_m(\mathbf{r}_{ij}) = \sum_{k=1}^{M} \hat{\theta}_k(m) f_k(\mathbf{r}_{ij} - \hat{\boldsymbol{\delta}}_m)$. Initialization schemes include autonomous algorithms of estimation of shifts between two frames based on simple spline-approximations. One simple but particularly efficient scheme is described below.

2. Typical Step $n, n > m$.

(a) *Jitter Estimation.* The estimate $\hat{b}_{n-1}(\mathbf{r}_{ij})$ obtained from the previous step is compared with the n-th frame (starting with $n = m$), and the ML/MD estimate of jitter $\hat{\boldsymbol{\delta}}_n$ is computed as the solution of the nonlinear optimization problem (12) with

$$\hat{b}_{n-1}(\mathbf{r}_{ij} - \boldsymbol{\delta}) = \sum_{k=1}^{M} \hat{\theta}_k(n-1) f_k(\mathbf{r}_{ij} - \boldsymbol{\delta}).$$

(b) *Estimation of Parameters in CCS*. Having the estimates $\hat{\boldsymbol{\delta}}_{n-m+1}, \ldots, \hat{\boldsymbol{\delta}}_n$, the estimates $\hat{\theta}_k(n)$ are computed for the n-th frame from the minimization problem (11). This recomputing in the corrected coordinate system is equivalent to frame alignment.

(c) *Clutter estimation in CCS*. Using the estimates obtained from (11) and (12), compute the estimate of $b_n(\mathbf{r}_{ij})$ for all i and j in the corrected coordinate system,

$$\hat{b}_n(\mathbf{r}_{ij}) = \sum_{k=1}^{M} \hat{\theta}_k(n) f_k(\mathbf{r}_{ij} - \hat{\boldsymbol{\delta}}_n). \quad (13)$$

(d) *Clutter Rejection.* Using the estimate (13), compute the residuals (filtered background)

$$\widetilde{Z}_n(\mathbf{r}_{ij}, \hat{\boldsymbol{\delta}}_n) = Z_n(\mathbf{r}_{ij}) - \sum_{k=1}^{M} \hat{\theta}_k(n) f_k(\mathbf{r}_{ij} - \hat{\boldsymbol{\delta}}_n).$$

The following autonomous algorithm of jitter compensation is simple but fairly efficient and can be recommended for initialization.

Consider the following simplest linear model for background prediction:

$$\tilde{b}(x_i - \delta_x, y_j - \delta_y) = a_{i,j} + g_{i,j}\delta_y + h_{i,j}\delta_x. \quad (14)$$

This model is used for any frame $\|Z_n(i,j)\|$ that is shifted with respect to a reference frame $\|Z_n^R(i,j)\|$ by (δ_x, δ_y).

Assuming a grid with a unit step (i.e., $\Delta_x = \Delta_y = 1$), for each pixel (i,j), the "optimal" estimates $\hat{a}_{i,j}$, $\hat{g}_{i,j}$, and $\hat{h}_{i,j}$ are found from the following least squares criterion

$$\min_{a,g,h} \sum_{k,m} \Big(Z_n^R(i+k, j+m) - [a_{i,j} + g_{i,j}m + h_{i,j}k] \Big)^2, \quad (15)$$

where the summation is performed over the values

$$\{(k,m) = (0,0), (0,-1), (0,+1), (-1,0), (+1,0)\}.$$

The jitter estimate $\hat{\boldsymbol{\delta}}_n = (\hat{\delta}_x(n), \hat{\delta}_y(n))$ is found as

$$(\hat{\delta}_x(n), \hat{\delta}_y(n)) = \underset{(\delta_x, \delta_y)}{\operatorname{argmin}} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \Big(Z_n(i,j) - [\hat{a}_{i,j} + \hat{g}_{i,j}\delta_y + \hat{h}_{i,j}\delta_x] \Big)^2. \quad (16)$$

The minimization problem (16) admits an explicit solution, i.e., the shift estimates can be written in an explicit form.

This algorithm can be improved by introducing higher order approximations. For example, the second order approximation gives slightly better estimation accuracy in certain situations at the expense of higher computational complexity.

As has been discussed above, the proposed autonomous jitter estimation algorithm is needed at the initialization stage of the parametric CLS filters. In stationary modes, the shift can be determined based on the comparison of the parametric model with the current data frame as follows

$$(\hat{\delta}_x(n), \hat{\delta}_y(n)) = \underset{\delta_x, \delta_y}{\operatorname{argmin}} \sum_{i,j} \Big(Z_n(x_i, y_j) - \hat{b}_{n-1}(x_i - \delta_x, y_j - \delta_y; \hat{\boldsymbol{\theta}}(n-1)) \Big)^2 \quad (17)$$

(see (12)).

Assuming that the shifts $\boldsymbol{\delta}_n$ are estimated by an independent algorithm (e.g., by the one described above), computing the estimate $\hat{\boldsymbol{\theta}}(n)$ is reduced to computing the matrix $\mathbf{R}_n$ and the vector $\mathbf{h}_n$ according to the following formulas

$$\mathbf{R}_n = \frac{1}{mN_xN_y} \sum_{i,j} \sum_{k=n-m+1}^{n} \mathbf{f}_{i,j}(k) \mathbf{f}_{i,j}^T(k),$$
$$\mathbf{h}_n = \frac{1}{mN_xN_y} \sum_{i,j} \sum_{k=n-m+1}^{n} \mathbf{f}_{i,j}(k) Z_k(i,j), \quad (18)$$

where $\mathbf{f}_{i,j}(k) = \mathbf{f}(x_i - \hat{\delta}_x(k), y_j - \hat{\delta}_y(k))$, with the subsequent solving the system of linear equations

$$\mathbf{R}_n \hat{\boldsymbol{\theta}}(n) = \mathbf{h}_n. \quad (19)$$

It is worth mentioning that in general an implementation of the identification algorithm (18)–(19) may be complicated by the curse of dimensionality. For example, usually the method of parametric approximation (the spatial functions $f_k(\mathbf{r})$ and the number M) has to satisfy the following requirements [23]:

- The error of approximation should be small.
- The system $\{f_k(\mathbf{r})\}$ should be quasi-orthogonal: $\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} f_\ell(\mathbf{r}_{ij})f_k(\mathbf{r}_{ij}) \approx 0$ for all $\ell \neq k$.

The first requirement may necessitate the use of a very large number M in the clutter approximation (9) in cases where clutter has fast spatial variations. In fact, M may be comparable or may even exceed N^2, where $N = N_x \times N_y$ is the number of pixels in the frame. This number is typically in the order of 10^6. In the latter case, the size of the matrix $\mathbf{R}_n$ is equal to 10^{12} and straightforward realization of the CLS filtering algorithm becomes impossible.

Therefore, it is important to find approximation models that allow for a decomposition of the vector $\boldsymbol{\theta}$ into independent blocks and of the equation (19) into a set of independent equations for these blocks. In other words, parallelization is crucial for the real-time implementation of this very powerful but computationally intense parametric method.

It is worth mentioning that the number of parameters to be estimated does not determine the *absolute computational complexity* of the algorithms. Parallelization of computations is the most important factor.

C. The Bank of CLS Filters

Our study of various algorithms showed that the following parametric models and corresponding spatial-temporal filters are feasible for implementation in the bank of CLS filters.

1. Two-dimensional Fourier Series with Double Nyquist Rate —"Fourier." From the point of view of the approximation error, a good choice is the 2D Fourier decomposition with the double Nyquist rate. In this case, the function $b(x, y)$ is represented by the Fourier series with spatial frequencies $\frac{2\pi\ell}{N_x\Delta_x}$ and $\frac{2\pi k}{N_y\Delta_y}$, $\ell = 1, \ldots, N_x$, $k = 1, \ldots, N_y$. The spectrum of the function $b(x, y)$ can be recovered in a larger frequency band than the Nyquist band namely because of platform vibrations, since vibrations allow us to observe the values of the function $b(x, y)$ in intermediate points and recover its spectrum in any frequency band.

The *Fourier* method has at least two merits. First, the parameters θ_k are the coefficients of the orthogonal decomposition, which automatically solves the problem of parallelization. Second, the form of the function $b_n(x - \delta_x, y - \delta_y)$ is very simple (the values of δ_x, δ_y determine phases of the terms of the Fourier series), which makes it easier to compute the shift estimates. The drawback is that it does not allow for an accurate estimation of sharp edges.

2. Two-dimensional Wavelet Series —"Wavelet." This method is similar to the previous method. Instead of the Fourier basis, it uses wavelet bases. In our model, the main benefit of using a wavelet basis is that, as opposed to the Fourier basis, wavelets have compact support. Wavelets are very efficient when describing functions with many jumps, edges, and similar local and "sharp" features with "bad frequency properties." This is often the situation in the case of clutter. In addition, they require a smaller number of parameters to be estimated (2 to 3 times smaller). In our experiments we used Daubechies wavelets [11].

3. Local Polynomial Approximation —"Pol." This model exploits an approximation of the function $b(x, y)$ by a k-th order polynomial surface independently for each pixel. Specifically, for each pixel (x_i, y_j) the frame shifted with respect to the reference frame by (δ_x, δ_y) can be approximated as in (8), i.e., has the form

$$b(x_i - \delta_x, y_j - \delta_y) = \mathbf{f}^T(\delta_x, \delta_y)\boldsymbol{\theta}_{i,j}, \tag{20}$$

where

$$\begin{aligned}&\mathbf{f}^T(\delta_x, \delta_y) = \\ &\left\|1, \delta_y, \ldots, \delta_y^k, \delta_x, \ldots, \delta_x^k, \delta_x\delta_y, \ldots, \delta_x^k\delta_y^k\right\|,\end{aligned} \tag{21}$$

and $\boldsymbol{\theta}_{i,j} = (\theta_{i,j}^0, \ldots, \theta_{i,j}^{p-1})$ is the $p-$dimensional vector of coefficient of the k-th order polynomial, $p = (k+1)(k+2)/2$. The reference frame is the first component $\theta_{i,j}^0$, $i = 1, \ldots, N_x$, $j = 1, \ldots, N_y$.

This model leads to a decomposition of the identification problem into N_xN_y independent problems of estimation of the $p-$dimensional vectors $\boldsymbol{\theta}_{i,j}$.

Under certain "regularization" (adding pseudo-noise and initial conditions) the least squares method (11) and (19) may be reduced to the recursive Kalman filter (see, e.g., [10], [14]). As a result, the optimal estimates $\hat{\theta}_{i,j}(n)$ of the parameters θ_{ij} are calculated independently for each pixel according to the following recursive Kalman equations [10], [14]

$$\begin{aligned}\hat{\theta}_{i,j}(n) =& \hat{\theta}_{i,j}(n-1) + K_n\hat{Z}_n(i,j);\\ K_n =& \Gamma_{n-1}\mathbf{f}_n^T/(\sigma^2 + \mathbf{f}_n^T\Gamma_{n-1}\mathbf{f}_n);\\ \hat{Z}_n(i,j) =& Z_n(i,j) - \mathbf{f}_n^T\hat{\theta}_{i,j}(n-1);\\ \Gamma_n =& \Gamma_{n-1} - K_n\mathbf{f}_n^T\Gamma_{n-1}\end{aligned} \tag{22}$$

with the initial conditions $\hat{\theta}_{i,j}(0) = 0$ and $\Gamma_0 = \gamma I$, where I is a unit matrix, γ is a large constant, and $\mathbf{f}_n = \mathbf{f}(\hat{\delta}_x(n), \hat{\delta}_y(n))$ is given by (21). The estimates of the shifts $(\hat{\delta}_x(n), \hat{\delta}_y(n))$ are found from (17).

Thus, the identification problem is reduced to the parallel implementation of linear Kalman filters. To limit the filter memory the update of the covariance matrix Γ_n is stopped for $n \geq m$. Note that $\mathbf{f}_n$, Γ_n, and K_n are the same for all pixels, which substantially simplifies the algorithm.

We implemented the approximation with a two-dimensional polynomial of the third order in discrete points with the step of a pixel size.

4. Spline-based Interpolation Methods with Double Resolution. Let $f(x, y) = f(x)f(y)$ be a symmetric spline and let $b_{k,\ell}^* = b(x_k, y_\ell)$ denote the values of the background in the discrete points $x_k = k/2$, $y_\ell = \ell/2$, $k = 1, \ldots, 2N_x$, $\ell = 1, \ldots, 2N_y$ (i.e., double resolution grid). The approximations (8) and (9) are used with $\boldsymbol{\theta} = \{b_{k,\ell}^*\}$ and $f(\mathbf{r}) = f(x)f(y)$. We implemented two particular splines – bilinear (triangular in both coordinates) and cubic that are shown in Fig. 2.

4.A. Regression with Bilinear Double-resolution Interpolation — "DRbil." In this method, for prediction of the function

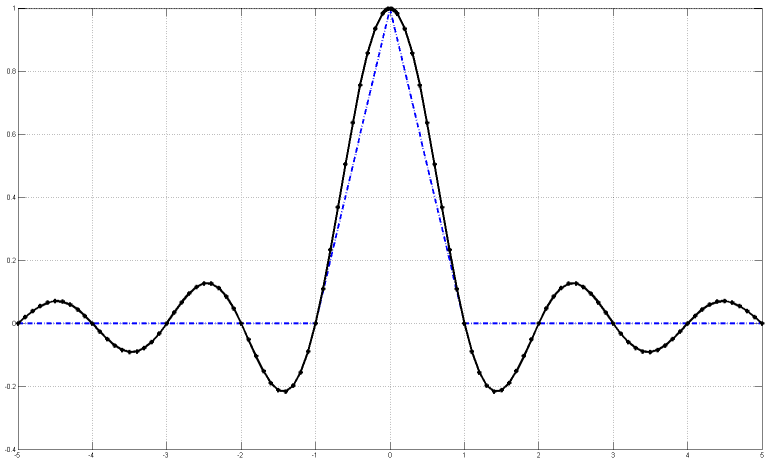


Fig. 2. The form of $f(t)$ for natural cubic (solid) and bilinear (dashed) splines

$b(x,y)$ between grid points a bilinear transformation (see Fig. 2) and the values $b^*_{k,\ell}$ in the discrete points $x_k = k/2$, $y_\ell = \ell/2$, $k = 1,\dots,2N_x$, $\ell = 1,\dots,2N_y$ are used. More specifically, the vector $\mathbf{f}(x,y)$ (see (8) and (9)) is determined by a bilinear interpolation of the function $b(x,y)$ based on its discrete values $b^*_{k,\ell}$ at points (x_k, y_ℓ), i.e., the prediction background estimate at the pixel (i,j) shifted by (δ_x, δ_y) (with respect to the reference frame) is calculated as

$$b^{(p)}(x_i - \delta_x, y_j - \delta_y) = \mathbf{f}^T(\delta_x, \delta_y)\mathbf{b}^*_{i,j},$$

where $\mathbf{b}^*_{i,j} = ||b^*_{k,\ell}, b^*_{k+s_x,\ell}, b^*_{k,\ell+s_y}, b^*_{k+s_x,\ell+s_y}||$, $k = 2i+1, l = 2j+1$ and

$$\mathbf{f}(\delta_x,\delta_y) = ((1-\tilde{\delta}_x)(1-\tilde{\delta}_y), \tilde{\delta}_x(1-\tilde{\delta}_y), (1-\tilde{\delta}_x)\tilde{\delta}_y, \tilde{\delta}_x\tilde{\delta}_y).$$

Here $\tilde{\delta}_x = 2\delta_x$, $\tilde{\delta}_y = 2\delta_y$, $s_x = \mathrm{sign}(\delta_x)$, and $s_y = \mathrm{sign}(\delta_y)$. The parameter $\boldsymbol{\theta} = \mathbf{b}^*_{i,j}$ as well as the values of shifts are estimated, i.e., the estimate $(\hat{\delta}_x, \hat{\delta}_y)$ is used in the algorithm.

Obviously, under the condition that the amplitude of vibrations (after jitter compensation) does not exceed one-half of a pixel size (i.e., $\delta_x \le 1/2$ and $\delta_y \le 1/2$), the interpolated value $b^{(p)}(x_i - \delta_x, y_j - \delta_y)$ depends only on the four values of $b^*_{k,\ell}$ that correspond to the four (x_k, y_ℓ) closest to this pixel.

Furthermore, a Kalman procedure similar to (22) is used for updating the estimate $\hat{\mathbf{b}}_{i,j}(n)$ obtained based on the previous n observations. The estimate $(\hat{\delta}_x, \hat{\delta}_y)$ is used in place of (δ_x, δ_y), which is computed as discussed above. As a result, the output of the CLS filter at the pixel (i,j) has the form

$$\widetilde{Z}_n(i,j) = Z_n(i,j) - \mathbf{f}^T(\hat{\delta}_x(n), \hat{\delta}_y(n))\hat{\mathbf{b}}_{i,j}(n).$$

Ideal independent decomposition (parallelization) is automatically guaranteed whenever $\delta_x \le 1/4$ and $\delta_y \le 1/4$. For larger shifts it is approximate but the algorithm has high performance unless the shifts exceed half of a pixel size.

4.B. Double-resolution Cubic Spline Interpolation — "DRspl." This method is absolutely similar to *DRbil*. The difference is in the form of interpolation: we used cubic spline-interpolation with a so-called natural cubic spline (see Fig. 2). A natural cubic spline is produced by piecewise third-order polynomials with the second derivative of each polynomial being set to zero at the endpoints. This provides a boundary condition that leads to a simple tridiagonal system which can be solved easily to calculate the coefficients of the polynomials.

Since double resolution is implemented, the *DRspl* filter is based on 5 discrete points independently in each coordinate, in which case the output of the CLS filter has the form

$$\widetilde{Z}_n(i,j) = Z_n(i,j) - \sum_{\ell,k=-2}^{2} f(x_i - \hat{\delta}_x(n)) f(y_j - \hat{\delta}_y(n)) \hat{b}_{2i+1-\ell,2j+1-k}(n),$$

where $f(t)$ is the cubic spline shown in Fig. 2.

Again, a Kalman update similar to (22) is used for updating the estimate $\hat{\mathbf{b}}_{i,j}(n)$.

In addition to the above spatial-temporal CLS filters, we tested and included in the bank a number of the most efficient spatial-only filters as well as purely temporal (without spatial processing) differencing algorithms. In particular, spatial algorithms include: "$M \times M$ matrix of order k" with $M = 3,4,5$ and $k = 0,2,4$, kernel-type nonparametric filters, and a "spatial in-frame adaptive" filter (based on the 2-D Fourier transform). It turns out that the following window-limited weighted algorithm with specially designed adaptive weights (kernel) is typically the most efficient:

$$\hat{b}_n(i,j) = \sum_{\ell,k\in\Omega} Z_n(i+\ell, j+k) W_{\ell,k}, \qquad (23)$$

where Ω is a spatial window of a certain size and form. The weights $W_{\ell,k}$ are chosen based on the trade-off between the efficiency of the clutter suppression and computational complexity. The results of experimental study given below correspond to the case of simple but one of the most efficient

spatial filters with the window size of 3×3 pixels and specially designed weights.

The temporal CLS algorithm is based on a sliding window of the size m, i.e., the estimate of the background is a moving average with a finite window:

$$\hat{b}_n(i,j) = \frac{1}{m}\sum_{t=n-m+1}^{n} Z_t(i,j), \quad n \geq m. \tag{24}$$

Note that for $m = 1$ this algorithm is nothing but a standard differencing method. This algorithm works well when there are no sensor vibrations and when the target velocity is greater than $1.5 - 2$ pixels/frame. In this case, it has the following performance indices:

$$G = \sqrt{(m+1)/m},\ Q \approx \sqrt{m/(m+1)},\ D \approx 0.$$

Already for the memory $m = 5$, it gives $Q \approx 0.9$, i.e., about 10% loss as compared to the ideal CLS algorithm. However, when the target velocity decreases the signal degradation factor D increases and the Q-factor rapidly decreases. For example, for $m = 5$ and velocity 0.3 pixels/frame, the value of Q drops to $0.2 - 0.3$. Also, the algorithm performance degrades dramatically when the jitter amplitude exceeds one-half of a pixel. Therefore, this algorithm requires a fairly accurate, separate algorithm for jitter compensation. See Section IV-B for further details.

IV. The Results of Experiments

The quality of clutter suppression algorithms should be evaluated based on the global operating characteristics that include the probability of detection, the track accuracy, and the false alarm rate. The global operating characteristics are monotone functions of intermediate indices that were discussed in detail in Section III-A. The following indices are particularly important: (a) the relative clutter suppression index G (G-factor, see (5)) that measures the degree of clutter rejection; and (b) the relative effective S(C+N)R Q (Q-factor, see (7)) that measures the degree of clutter rejection and signal preservation simultaneously. Recall that $G \geq 1$, $Q \leq 1$ and the ideal values are $G = Q = 1$, which means that clutter is totally suppressed, while the signal is completely preserved.

Besides, the mean-squared error (MSE) of jitter estimation is important. This error has a substantial impact on the performance of CLS algorithms and estimation of the targets' coordinates.

Despite the fact that the problems of target detection and tracking is out of the scope of this paper, it is worth mentioning that when comparing CLS algorithms, e.g., spatial-temporal and spatial-only methods, a quasi-optimal (in-frame) adaptive detection algorithm with constant false alarm rate (CFAR) in each frame has been applied. When the sensor PSF is known, the detection algorithm represents a generalized likelihood ratio test (assuming Gaussian distribution of residuals) with an adaptive threshold, which ensures an automatic adjustment of the detection threshold with the use of the background estimates. As a result, a density of false detections is maintained approximately constant in different fragments of a frame. In order to control the frequency of false detections (CFAR property), the maximum number of instantaneous detections is restricted by a certain value which is computed based on the maximum expected number of targets and other similar considerations. This strategy is being used for all CLS filters, which guarantees an optimal threshold selection for each and every CLS filtering algorithm so that the detection threshold for any particular CLS filter could not have been selected better.

A. Comparison of Developed CLS Algorithms in Various Conditions

Extensive simulations and experiments in a variety of conditions have been performed to estimate the capabilities and performance of the jitter estimation, clutter suppression, and target tracking algorithms. Sample results are shown in Figures 3-5 and Tables I-V.

As it was discussed in Section II, the developed architecture includes the built-in simulator that allows for simulating backgrounds in various spectral bands, daytime, and whether conditions as well as the Graphical User Interface (GUI), which allows one to input/output synthetic and external real data, specify and change parameters, and display all the results (see Fig. 1). The GUI is an important part of the development and Fig. 3 shows its general view. The windows on the top show the raw image, the whitened image at the output of the CLS filter, and target tracking results. Comments to each window provide estimates of various important parameters such as mean intensities, variances, jitter amplitudes (x, y shifts), detection thresholds, number of detections (blips) part of which are false, type of the best CLS filter in specifically encountered conditions, etc. The two windows at the bottom allow for a more detailed (magnified) view of the input frames and the results of processing. One can switch between several modes to view pure tracks, whitened frames with tracks, etc. More detailed results are documented in files that can be previewed off-line. In Fig. 3 squares correspond to instantaneous detections (part of which are false), which appear and disappear with each frame, and solid lines correspond to estimated and confirmed target tracks.

Let σ_b^2 and ρ denote the variance and correlation coefficient of clutter, respectively. The clutter-to-noise-ratio will be defined as $\mathrm{CNR} = \sigma_b/\sigma_N$. In Tables I–V, we present the results of simulations for the following three different scenarios:

1) Relatively weak clutter with moderate spatial correlation: $\mathrm{CNR} = 5.4$, $\rho = 0.85$.
2) Moderately intense clutter with very low spatial correlation: $\mathrm{CNR} = 25.8$, $\rho = 0.2$.
3) Very intense clutter with high spatial correlation: $\mathrm{CNR} = 78.4$, $\rho = 0.95$.

In all three simulation cases, the jitter was about half of a pixel size.

It is seen that spatial-temporal algorithms work well in all scenarios. However, there is no single optimal algorithm for all conditions, which suggests using a bank of CLS filters with a reconfigurable architecture. Spatial-only algorithms perform moderately well only for weak and relatively correlated clutter. In other conditions, spatial algorithms completely or almost

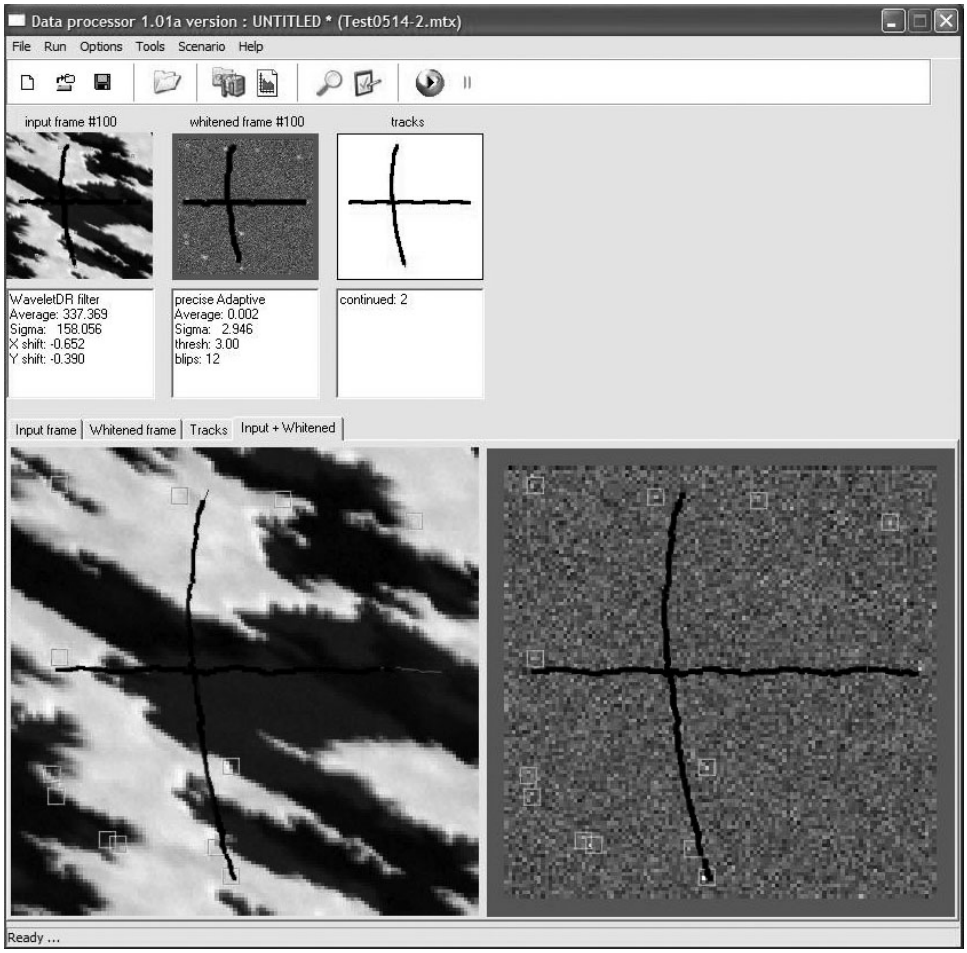


Fig. 3. Graphical user interface

TABLE I

$G-$VALUES FOR THREE OBSERVATION CONDITIONS

Filter	*No Filtering*	*Spatial*	*DRbil*	*Fourier*	*Wavelet*	*DRspl*	*Pol*
Scenario 1	5.40	1.73	1.25	1.07	1.06	1.06	1.20
Scenario 2	25.80	28.90	1.96	1.08	1.05	2.56	1.68
Scenario 3	78.4	4.29	1.26	1.26	1.18	1.16	1.22

TABLE II

$Q-$VALUES FOR SCENARIO 1: CNR= 5.4, $\rho = 0.85$, $T_{mem} = 20T_0$

Target Velocity	*No Filtering*	*Spatial*	*DRbil*	*Fourier*	*Wavelet*	*DRspl*
0.2 pix/frame	0.18	0.38	0.53	0.70	0.72	0.73
0.5 pix/frame	0.18	0.38	0.62	0.82	0.83	0.88

TABLE III

$Q-$VALUES FOR SCENARIO 2: CNR= 25.8, $\rho = 0.20$, $T_{mem} = 20T_0$

Target Velocity	*No Filtering*	*Spatial*	*DRbil*	*Fourier*	*Wavelet*	*DRspl*
0.2 pix/frame	0.04	0.02	0.27	0.61	0.60	0.30
0.5 pix/frame	0.04	0.02	0.35	0.71	0.72	0.38

completely fail. For example, if clutter has a very high spatial variation (low correlation) it is even better not to apply any filtering than to use spatial processing. See Table III for moderately intense and weakly correlated clutter. The results in Tables I–IV correspond to the best spatial filter among those tested, which is a 3×3 matrix filter with specially designed adaptive weights given by (23).

The data in Table V show that the proposed jitter compen-

TABLE IV
$Q-$VALUES FOR SCENARIO 3: CNR= 78.4, $\rho = 0.95$, $T_{mem} = 20T_0$

Target Velocity	*No Filtering*	*Spatial*	*DRbil*	*Fourier*	*Wavelet*	*DRspl*
0.2 pix/frame	0.01	0.15	0.36	0.56	0.57	0.72
0.5 pix/frame	0.01	0.15	0.50	0.63	0.68	0.83

TABLE V
MSE OF JITTER ESTIMATION IN THE UNITS OF THE PIXEL SIZE

Filter	*Wavelet*	*Fourier*	*DRspl*	*DRbil*	*Pol*
Scenario 1	0.007	0.008	0.036	0.069	0.024
Scenario 2	0.003	0.003	0.021	0.028	0.025
Scenario 3	0.056	0.058	0.014	0.013	0.026

sation algorithms are highly efficient and typically allow for residual jitter in the range of $1-3$ percent of a pixel size.

Figures 4-5 illustrate the results of clutter suppression and target detection and tracking for various observation conditions. The following parameters were used in simulated experiments: Geostationary orbit; FOV (field-of-view) 128×128 pixels with pixel size of $30-35$ microradians; instantaneous FOV fluctuates around the reference FOV because of LOS vibrations (vibration amplitude about one-half of a pixel size); point targets move with constant velocity $0.5-1$ pix/sec; frame rate 1 fps; Gaussian PSF. The efficiency of the CLS algorithms and software has been demonstrated by processing data sets obtained from the simulator (the imitation model of images) that takes into account generalized hardware/sensor parameters, clutter and LOS vibrations. We have simulated various scenarios under a variety of conditions.

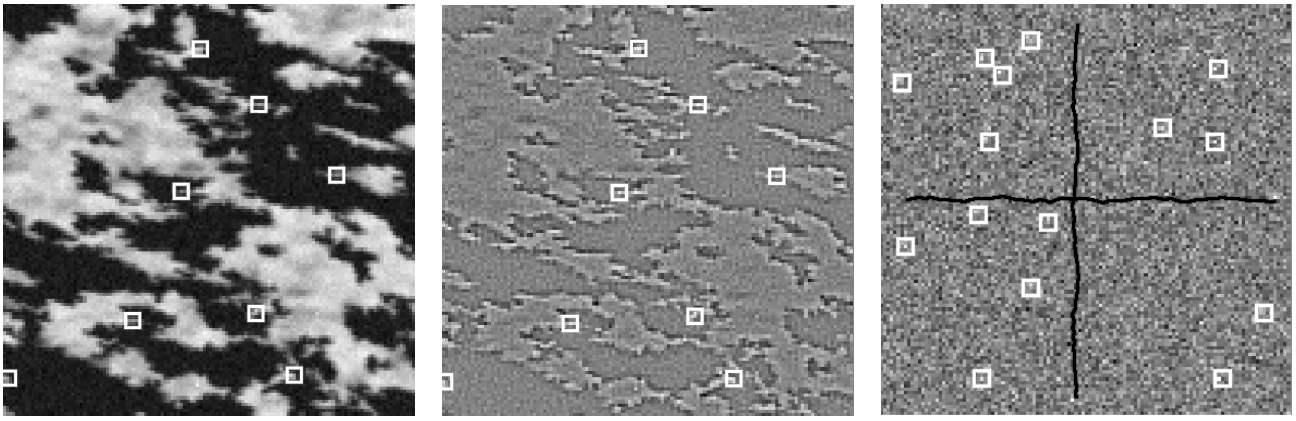
Fig. 4 illustrates the tracking of two weak targets with S(C+N)R = 0.1. In this particular example the simulated sequence corresponds to the following conditions:

- *Generalized sensor parameters:* pixel size 35 μrad, standard deviation (STD) of Gaussian sensor noise 3 quants of analog-to-digital converter (ADC).
- *Meteorological conditions:* two-layer clouds — height of the upper boundary of clouds (HUBC) of the lower layer $\approx$ 3 km, HUBC of the upper layer $\approx$ 10 km, STD of clutter ≈ 150 quants of ADC.
- *Targets:* 2 targets moving along perpendicular lines with constant velocity ≈ 0.5 pix/frame.
- *LOS vibrations:* amplitude of vibrations 1 pix.

In this example, the auto-selection algorithm has chosen the *Fourier* filter as the best filter from the bank. This filter almost completely suppressed clutter (STD of residuals $\approx$ STD of sensor noise). Targets were immediately detected and tracked from the beginning to the end, as can be seen from Fig. 4(c). Fig. 4(b) shows the results for the Spatial-only CLS filter (23). The spatial filter fails: targets are not tracked when this filter is used. In Fig. 4, boxes represent instantaneous detections (mostly false), which appear and disappear with each frame, and solid lines correspond to estimated target tracks which were confirmed.

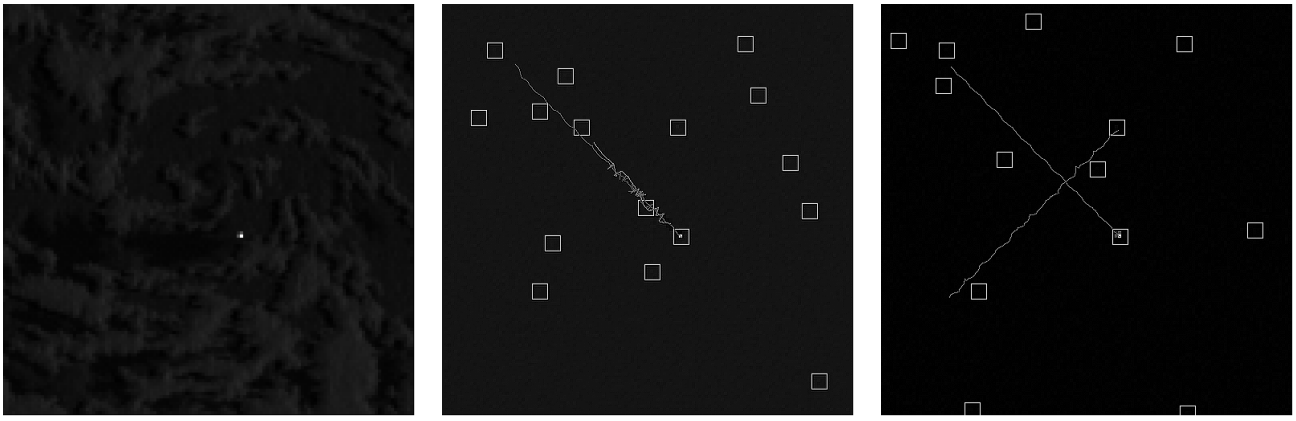
In the course of experimentation we discovered an interesting effect related to the influence of bright targets (or other point-like sources) on the CLS algorithm performance. It turns out that in the presence of bright sources the performance degrades dramatically. In particular, weak targets cannot be detected and tracked in the presence of bright targets. Therefore, a procedure for strong signal compensation that intends to eliminate the influence of bright targets is needed. This is crucial for improving the algorithm's performance in the presence of strong signals. The development of such a procedure has been performed but is out of the scope of this paper and will be presented elsewhere. Here we only mention that the signal compensation includes nonlinear estimation of strong signals and subsequent subtraction of these resulting estimates from the raw data. More specifically, the signal estimates $\hat{\mathbf{s}}_n$ are subtracted from the raw data to form the statistic $\widehat{\mathbf{Z}}_n = \mathbf{Z}_n - \hat{\mathbf{s}}_n$. Spatial-temporal CLS algorithms are then applied to the sequence of statistics $\widehat{\mathbf{Z}}_{n-m+1},\dots,\widehat{\mathbf{Z}}_n$ in the time window m to estimate the background. The resulting estimate $\hat{\mathbf{b}}_n$ is subtracted from $\mathbf{Z}_n$ to form the residuals $\widetilde{\mathbf{Z}}_n = \mathbf{Z}_n - \hat{\mathbf{b}}_n$ which are the output of the CLS filters. The data at the outputs of all filters in the bank are analyzed by the auto-selection block to choose the best configuration for the current conditions, as was discussed in Section II. In the experiments below we used an ad hoc algorithm of bright signal estimation that is based on a simple thresholding of raw data (with thresholds higher than those used for the target detection).

To illustrate the aforementioned effect, we present the results of tracking one very strong point target with S(C+N)R = 100 and one weak target with S(C+N)R = 0.04. The corresponding results are shown in Fig. 5 where the squares with no dots inside and no tracks attached represent instantaneous detections part of which are false. False detections, however, did not lead to real false alarms, since no false tracks were formed. The squares with dots inside and with tracks attached represent true target detections that were confirmed and tracked. The polynomial (*Pol*) filter was used for clutter rejection — with and without compensation of the strong signal. A strong target was detected and tracked in both cases. The weak target was tracked only when strong signal compensation was used. Figures 5(b) and 5(c) also illustrate the difference in the jitter compensation accuracy when there is no strong signal compensation and when strong signal compensation is used. This can be judged based on the "fuzziness" of the tracks. Indeed, the track in Fig. 5(b) is substantially more fuzzy



(a) Original data frame (b) Output of the *Spatial-only* CLS filter (c) Output of the *Fourier* CLS filter

Fig. 4. Target detection and tracking. Two weak targets: SNR=5, CNR=50, target velocity=0.5pix, jitter=1pix



(a) Original data frame (b) Output of the *Pol* CLS filter without strong signal compensation (c) Output of the *Pol* CLS filter with strong signal compensation

Fig. 5. Target detection and tracking. One strong and one weak target: SNR(strong)=10000, SNR(weak)=4, CNR=100, target velocity=0.5pix, jitter=0.5pix

(even for a very bright target) because of the residual jitter as compared to Fig. 5(c).

B. Comparison with Standard Differencing Methods

We performed a detailed comparative study of the "industry standard" differencing method with our clutter rejection techniques. The differencing clutter rejection method simply subtracts two consecutive frames. It is therefore equivalent to our temporal window-limited clutter rejection filter (implemented in the bank of CLS filters) with the window size of 1 frame. See (24) with $m = 1$.

We first comment on this algorithm in detail. The first important observation is that the differencing algorithm may work well if, and only if, almost complete stabilization (sub-sub-pixel) is performed. It also requires an independent stabilization algorithm in contrast to most our CLS filters where stabilization is performed in the course of and jointly with clutter rejection. The reason is obvious — this algorithm is not only robust but, by contrast, it is quite sensitive to any changes that occur between images. In addition, a simple subtraction of frames doubles the intensity of sensor noise. Therefore, small weak targets will not be detected and tracked even if clutter is suppressed.

To be more specific, consider the following ideal situation. Assume that the platform is completely stable (or stabilized) so that there is no jitter, and also the background does not change, i.e., completely "frozen." Then the two subsequent observations, for $k = n$ and $n + 1$, are

$$Z_k(\mathbf{r}_{ij}) = S(\mathbf{r}_{ij} - \mathbf{r}(k)) + b(\mathbf{r}_{ij}) + \xi_k(\mathbf{r}_{ij}),$$

where $\mathbf{r}(k)$ is the location of the target at time k. Assuming further that the target moves fast enough, so that the signals do not intersect in successive frames, we obtain that at the output of the differencing CLS filter we observe

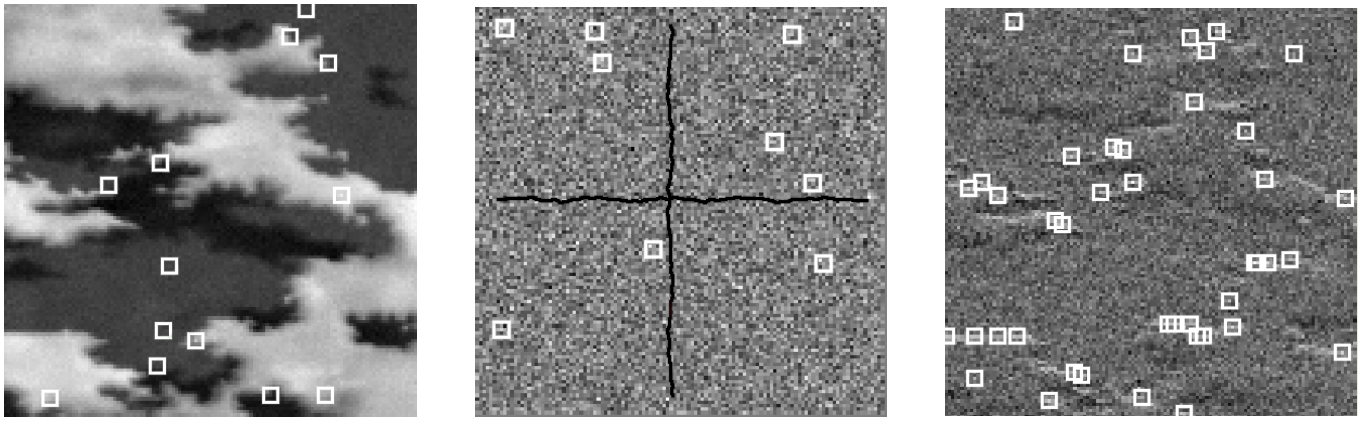
$$\widetilde{Z}_{n+1}(\mathbf{r}_{ij}) = S(\mathbf{r}_{ij} - \mathbf{r}(n+1)) + \xi_{n+1}(\mathbf{r}_{ij}) - \xi_n(\mathbf{r}_{ij}).$$

Therefore, clutter will be completely suppressed but the intensity of sensor noise will be doubled, since

$$\mathbf{E}[\xi_{n+1}(\mathbf{r}_{ij}) - \xi_n(\mathbf{r}_{ij})]^2 = 2\sigma_N^2,$$

where σ_N^2 is the variance of noise. This means that the SNR is $S^2/2\sigma_N^2$, i.e., twice lower than the potential maximal SNR S^2/σ_N^2.

On the other hand, for most of our CLS algorithms the SNR will be close to S^2/σ_N^2. For example, for our temporal



(a) Original frame (b) Spatial-temporal *Wavelet* filter (c) *Differencing* CLS algorithm

Fig. 6. The results of clutter suppression and target tracking using spatial-temporal Wavelet and differencing CLS filters

algorithm (24) with the window size of m frames we have

$$\widetilde{Z}_{n+1}(\mathbf{r}_{ij}) = S(\mathbf{r}_{ij} - \mathbf{r}(n+1)) + \xi_{n+1}(\mathbf{r}_{ij}) - \frac{1}{m}\sum_{k=n-m+1}^{n} \xi_n(\mathbf{r}_{ij}),$$

i.e., the variance of noise at the output is

$$\sigma_N^2 + \sigma_N^2/m = \frac{m+1}{m}\sigma_N^2.$$

For $m \geq 10$, the SNR is $mS^2/(m+1)\sigma_N^2 \approx S^2/\sigma_N^2$.

We now confirm these points by experiments with synthetic data. In order to "enhance" the capability of the differencing algorithm we used ideal conditions in terms of stabilization — without jitter. As we will see, even in these conditions the algorithm performs poorly, as can be expected from the above argument.

We simulated an image sequence with moderately intense clutter and sensor noise STD $\sigma_N = 3$. Two weak targets were inserted in the sequence. We first used the *Wavelet* spatial-temporal filter with window of 20 frames. The results were very successful — the standard deviation of the residual clutter plus noise was about 3 and both targets were tracked, as can be seen in Fig. 6(b). By contrast, the differencing method was not able to track targets, as seen from Fig. 6(c). As before, squares with no tracks attached represent instantaneous detections part of which are false, while solid lines correspond to confirmed target tracks.

V. Conclusions

1. The results of experiments and realistic simulations obtained with a Software Simulator show that the developed adaptive spatial-temporal clutter suppression and image stabilization algorithms, in particular spatial-temporal Wavelet and Fourier parametric methods, allow for efficient clutter rejection in all tested situations, especially when supplemented with the additional strong signal compensator. These algorithms completely remove heavy clutter (to or even below the level of sensor noise) in the presence of substantial jitter and *do not require expensive sub-pixel jitter stabilizers*.

2. Experimental study shows that the proposed jitter compensation algorithms guarantee that the residual jitter is less than $1/10$-th of a pixel in all considered scenarios. Typically the residual jitter can be made as low as $1-3\%$ of a pixel. Note that this result is somewhat unique. Indeed, our algorithmic stabilization is better than can be achieved by the best available commercial mechanical and/or electronic stabilizers. In fact, the use of very expensive super-stabilizers is not necessary and allows for *cost reduction*.

3. There is no single optimal CLS filter in all situations. For this reason, a reconfigurable, adaptive architecture was designed that allowed us automatic selection of the best filter from the bank in specific conditions.

4. Spatial-only filtering can be used only for weak and relatively correlated (smooth) clutter. Spatial methods have poor performance for intense and relatively smooth clutter. Spatial-only processing methods completely fail for heavy and non-smooth clutter.

5. An industry standard differencing method may perform relatively well if, and only if, an extraordinary jitter compensation accuracy is achieved, and even in this case it performs poorly when tracking relatively weak targets.

Acknowledgements

The research of Alexander Tartakovsky was supported in part by the U.S. Office of Naval Research grants N00014-99-1-0068 and N00014-95-1-0229 and by the U.S. Army Research Office MURI grant W911NF-06-1-0094 at the University of Southern California and by the Missile Defense Agency SBIR contract FA8718-04-C-0059 at ADSANTEC. The research of James Brown was supported by the Air Force.

We thank Dr. Leonid Volfson for useful discussions. We are also grateful to the referees and the associate editor for a comprehensive review and excellent comments that have improved the paper.

References

[1] M. Alam, J. Bognar, R. Russel, and B. Yasuda, "Infrared image registration and high-resolution reconstuction using multiple translationally shifted aliased video frames," *IEEE Trans. Instrum. Measur.*, Vol. 49, pp. 915–923, 2000.

[2] T. Aridgides, G. Cook, S. Mansur, and K. Zonca, "Correlated background adaptive clutter suppression and normalization techniques," *SPIE Proceedings: Signal and Data Processing of Small Targets*, (O.E. Drummond, Ed.), Vol. 933, pp. 32-44, Orlando, 1988.

[3] A. Aridgides, M. Fernandez, and D. Randolph, "Adaptive three-dimensional spatio-temporal IR clutter suppression filtering techniques," *SPIE Proceedings: Signal and Data Processing of Small Targets*, (O.E. Drummond, Ed.), Vol. 1305, pp. 63-74, Orlando, 1990.

[4] J.T. Barnett, "Statistical analysis of median subtraction filtering with application to point target detection in infrared backgrounds," *Proc. SPIE Symposium on Infrared Signal Processing*, Vol. 1050, pp. 10-18, Los Angeles, CA, 1989.

[5] J.T. Barnett, B.D. Billard, and C. Lee, "Nonlinear morphological processors for point-target detection versus an adaptive linear spatial filter: a performance comparison," *Proc. SPIE Symposium on Aerospace Sensing*, Vol. 1954, pp. 12-24, Orlando, FL, 1993.

[6] Y. Bar-Shalom and X.R. Li, *Estimation and Tracking: Principles, Techniques and Software*, Artech House, Boston-London, 1993.

[7] M. Basseville and I.V. Nikiforov, *Detection of Abrupt Changes: Theory and Applications*, Prentice Hall, Englewood Cliffs, 1993.

[8] S.S. Blackman, *Multiple-Target Tracking with Radar Applications*, Artech House, Dedham, 1986.

[9] S. Blackman, R. Dempster, and T. Broida, "Multiple hypothesis track confirmation for infrared surveillance systems," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, pp. 810-823, 1993.

[10] K. Brammer and G. Siffling, *Kalman-Bucy Filters*, Artech House, Norwood, MA, 1989.

[11] I. Daubechies, *Ten Lectures on Wavelets* (CBMS-NSF Regional Conference Series in Applied Mathematics), SIAM, Capital City Press, Montpelier, Vermont, 1992.

[12] M. Fernandez, A. Aridgides, and D. Bray, "Detecting and tracking low-observable targets using IR," *SPIE Proceedings: Signal and Data Processing of Small Targets*, (O.E. Drummond, Ed.), Vol. pp. 193-206, Orlando, 1990.

[13] R. Fries, C. Ferrara, W. Ruhnow, and H. Mansur, "A clutter classifier driven filter bank for the detection of point targets in non-stationary clutter," National Iris Conference, 1988.

[14] A.H. Jazwinski, *Stochastic Processes and Filtering Theory*, Academic Press, 1970.

[15] S. Kligys, B.L. Rozovsky, and A.G. Tartakovsky, "Detection algorithms and track before detect architecture based on nonlinear filtering for infrared search and track systems," Technical report # CAMS–98.9.1, Center for Applied Mathematical Sciences, University of Southern California, 1998. (Available at http://www.usc.edu/dept/LAS/CAMS/usr/facmemb/tartakov/preprints.html).

[16] R.L. Lucke and A.D. Stocker, "Filtering interpolators for frame differencing signal processors," *IEEE Transactions on Signal Processing*, Vol. 41, no. 8, pp. 2574-2582, 1993.

[17] W. Shaffer and R. Lucke, "Simultaneous registration and nonuniformity correction of space-based IR images for a scanning sensor," in *Proc. IRIS Speciality Group Meeting on Targets, Backgrounds, Discrimination*, January 28-30, 1992.

[18] A.P. Schaum, "Principles of interpolator design and evaluation," NRL Report 9356, Naval Research Laboratory, 1991.

[19] A.P. Schaum, "Dual difference filtering: A replacement for interpolation and subtraction to detect changes in misregistered signals," Report NRL/FR/5620-92-9522, Naval Research Laboratory, 1992.

[20] A.G. Tartakovsky, *Sequential Methods in the Theory of Information Systems*, Radio & Communications, Moscow, 1991 (In Russian).

[21] A.G. Tartakovsky, "Asymptotic performance of a multichart CUSUM test under false alarm probability constraint," *44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC05)*, pp. 320-325, December 12-16, 2005, Seville, Spain, Omnipress CD-ROM, ISBN 0-7803-9568-9.

[22] A.G. Tartakovsky, "Robust adaptive spatial-temporal algorithms for clutter rejection and scene stabilization," *ADSANTEC MDA SBIR Phase II Final Technical Report No. AFRL-VS-HA-TR-2006-1102*, 2006.

[23] A.G. Tartakovsky and R. Blažek, "Effective adaptive spatial-temporal technique for clutter rejection in IRST," *SPIE Proceedings: Signal and Data Processing of Small Targets*, Vol. 4048 , (O.E. Drummond, Ed.), Orlando, FL, 2000.

[24] A. Tartakovsky, S. Kligys, and A. Petrov, "Adaptive sequential algorithms for detecting targets in heavy IR clutter," *SPIE Proceedings: Signal and Data Processing of Small Targets*, (O.E. Drummond, Ed.), Vol. 3809, pp. 119-130, Denver, 1999.

[25] A.G. Tartakovsky and V.V. Veeravalli, "Change-point detection in multichannel and distributed systems with applications," In *Applications of Sequential Methodologies*, (N. Mukhopadhyay, S. Datta and S. Chattopadhyay, Eds.), Marcel Dekker, Inc., New York, pp. 339–370, 2004.

[26] J.B. Wilburn, "Theory of ranked-order filters with applications to feature extraction and interpretive transforms," *Advances in Imaging and Electron Physics*, (P. Hawkes, Ed.), Harcourt-Brace Academic Press, Vol. 112, pp. 233-332, 2000.

Dr. Alexander Tartakovsky is the Associate Director of the Center for Applied Mathematical Sciences and Professor in the Department of Mathematics at the University of Southern California, Los Angeles. He is also Vice President of Argo Science Corp., Rolling Hills Estates, CA. His research interests include theoretical and applied statistics; sequential analysis; change-point detection phenomena; adaptive, minimax and robust methods for overcoming prior uncertainty; pattern recognition; speech recognition and speaker identification; statistical image and signal processing; image stabilization; video tracking; detection and tracking of targets in radar and infrared search and track systems; information integration/fusion; intrusion detection and network security; and detection and tracking of malicious activity. He is the author of one book ("Sequential Methods in the Theory of Information Systems") and over 70 articles in the areas indicated above. Dr. Tartakovsky obtained an M.S. degree in electrical engineering from Moscow Aviation Institute (Russia) in 1978, a Ph.D. in statistics and information theory from Moscow Institute of Physics and Technology (Russia) in 1981, and an advanced Doctor-of-Science degree in statistics and control from Moscow Institute of Physics and Technology (Russia) in 1990. He is a member of the Institute of Mathematical Statistics, SPIE, Information Fusion Society, and a senior member of IEEE. Dr. Tartakovsky is a recipient of the *2007 Abraham Wald Award in Sequential Analysis*.

James Brown is research physicist, team leader, and mission scientist within the Space Vehicles Directorate, Battlespace Environment Division (Air Force Research Laboratory, Hanscom AFB). Started with the Air Force Cambridge Research Laboratory in 1970 where he worked on chemilumenescent reactions and cross-molecular beam dimerization reactions. Continued in the 1980s as the Air Force Geophysics Laboratory team leader for characterizing optical turbulence. Led AFRL high altitude radiance and transmission model and code development for Missile Defense Agency phenomenology program. Presently engaged in experimental measurements of space objects for Air Force Space Situational Awareness technologies.