Optimization of Point Target Tracking Filters

CHARLENE E. CAEFER
JERRY SILVERMAN
JONATHAN M. MOONEY
Air Force Research Laboratory

We review a powerful temporal-based algorithm, a triple temporal filter (TTF) with six input parameters, for detecting and tracking point targets in consecutive frame data acquired with staring infrared (IR) cameras. Using an extensive data set of locally acquired real-world data, we used an iterative optimization technique, the Simplex algorithm, to find an optimum set of input parameters for a given data set. Analysis of correlations among the optimum filter parameters based on a representative subset of our database led to two improved versions of the filter: one dedicated to noise-dominated scenes, the other to cloud clutter-dominated scenes. Additional correlations of filter parameters with measures of clutter severity and target velocity as well as simulations of filter responses to idealized targets reveal which features of the data determine the best choice of filter parameters. The performance characteristics of the filter is detailed by a few example scenes and metric plots of signal to clutter gains and signal to noise gains over the total database.

Manuscript received January 27, 1997; released for publication September 24, 1999.

Refereeing of this contribution handled by J. J. Wharton.

IEEE Log No. T-AES/36/1/02585.

This work was carried out under Air Force Task 2305C1.

Authors' address: Air Force Research Laboratory, AFRL/SNHI, 80 Scott Dr., Hanscom AFB, MA 01731-2909.

0018-9251/00/$10.00 2000 IEEE

I. INTRODUCTION

We are investigating the potential for staring focal plane infrared (IR) arrays to carry out an Infrared Search and Track (IRST) mission [1]. As the centerpiece for such a function, modern staring IR technology promises passive day and night operation, high sensitivity, and high frame rate. The rapid frame rate increases the correlation in time of evolving clutter (such as clouds). Hence, temporal signal processing algorithms should offer superior clutter rejection to that of the standard spatial processing approaches [2]. The desire to maximize target detection range focuses attention on algorithms for detecting and tracking point targets. Targets of interest and current staring camera parameters, such as IFOV and frame rates, translate into point targets which move only fractions of a pixel per frame time. Our search for and development of the requisite temporal algorithms is closely guided by real-world data with airborne targets of opportunity. Such data has been acquired with on-hand, state of the art staring IR cameras.

We have identified and reported a new class of temporal filters [3, 4] whose ease of hardware implementation and inherent lack of response to evolving cloud clutter offer great potential for the IRST application. A survey of the historical development of tracking point targets in the IR is presented in these earlier reports. We tested many of these approaches in our effort to implement a staring-based IRST. Our temporal filter approach is in the spirit of recent techniques which use multiple frames in retaining and updating target probability values for each pixel with no premature thresholding [5, 6]. This is achieved through a recursive implementation with a weighting of past history.

The design and test of these filters has been closely controlled by real-world data which has dictated a somewhat empirical approach. The first reports [3, 4] of our temporal filters focused exclusively on the evolving cloud scenario and compared their performance with that of our initial approach: a bank of spatio-temporal velocity filters [7]. In later work [8], we concentrated on the temporal filters emphasizing issues and trade-offs in choosing their free parameters. We identified parametric variations with better signal to temporal noise (as contrasted with signal to cloud clutter) response which gave improved performance in clear sky scenes but showed increased cloud clutter leakage in other scenarios.

Here we address a problem of increasing importance. As algorithms with parametric variations are used to process real data in real-time hardware, how can one systematically and effectively incorporate such data in the design (not just the test) of such

algorithms? Algorithm types which incorporate training sets into their design, such as neural nets, attempt to meet this need. The approach discussed here is to use the Simplex algorithm to find the most successful parameter set for a given algorithm working on a given data sequence in terms of optimum target to nontarget response. We have used very extensive real-world IR data with moving airborne targets and complex evolving cloud clutter in an attempt to optimize these promising temporal filters for point target tracking. The data includes clear sky scenes, both night and day, and a variety of challenging evolving cloud scenes. The targets of opportunity contained in the data base exhibit wide variation in target focal plane sampling as well as focal plane velocities: from about 0.02 to 0.50 pixels/frame (p/f). By examining the correlations among the filter parameters over our data base, we have identified significantly improved forms of our algorithm. Further, simulations of filter performance on ideal embedded targets as well as correlations with clutter severity and target velocity have led to a clearer picture of filter/data interactions and internal filter parameter constraints. While six parameters are explicitly present in the filter description, the parameter choice of the main period of the filter tightly constrains the optimum choice for the other five. While we hoped to find one set of parameters "for all seasons," our results showed clustering about two parameter sets: one optimum for clear sky and benign (stationary) cloud scenes, the other for challenging cloud scenes.

Our presentation is organized as follows. Review of the temporal filters and their parameter base is presented in Section II. Description of the process of optimization is found in Section III. The analysis and conclusions of the optimization results are in Section IV. Performance characterizations of the improved tracking filters is in Section V, and conclusions and future work plans are presented in Section VI.

II. TRIPLE TEMPORAL FILTER

The generic form of our filter weights the data of each pixel in time by a zero-mean damped sinusoid as shown in Fig. 1. The zero-mean property insures that it will only respond to pixels changing in time. The filter responds very weakly to changes that are monotonic over long time spans, but strongly to temporal profiles which rise and fall over briefer periods. Fig. 2 shows the typical monotonic temporal profile of a drifting cloud edge as contrasted with the temporal profile of a target, which rises and falls as the target passes through the pixel. The intensity of the target scales the profile height, while the inverse of the target velocity scales the width. Therefore, the

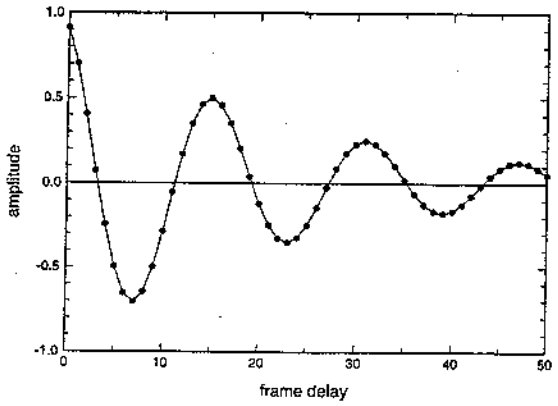


Fig. 1. Example of generic temporal filter: zero-mean damped sinusoid of period 16 frames and half-damping 16 frames. Circles indicate discrete frame intervals.

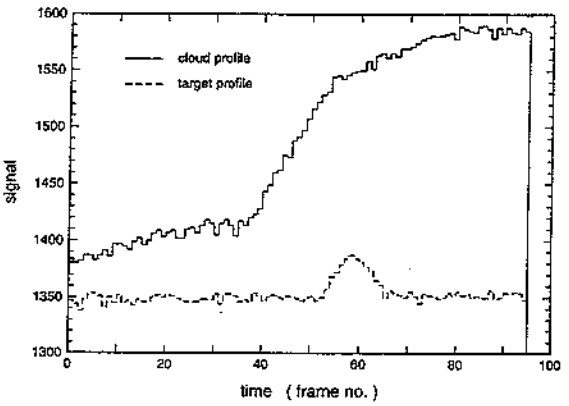


Fig. 2. Temporal profiles over 95 consecutive frames of two pixels in daytime cloud scene: one traversed by moving point target, the other with entering cloud edge.

filter inherently responds to moving points and rejects stationary objects and moving edges.

In order to minimize hardware complexity, we implement the damped sinusoid recursively by means of the expression

$$z_{k+1} = z_k \alpha e^{i\theta} + d_{k+1} e^{i\phi} \tag{1}$$

where z_k is the complex output of the kth iteration, α is the damping constant (memory persistence) between zero and one given by $e^{\log(0.5)/l}$, l is the full width at half max of the exponential decay, θ is the angular shift per iteration in the sinusoid filter given by $\theta = 2\pi/P$ where P is the period of the sinusoid in number of frames, d_{k+1} is the next pixel data value, and the phase ϕ is given by

$$\tan(\phi) = (1 - \alpha\cos(\theta))/(\alpha\sin(\theta)). \tag{2}$$

ϕ is that phase shift of the sinusoid which ensures that the real part of z_k has zero response to a constant intensity signal. Since we would like to threshold the results to clearly distinguish point targets from residual clutter, the filter is completed by taking the absolute value of the real part of z_k. By applying

D_2	V_T	D_1
H_L	P	H_R
D_3	V_B	D_4

Fig. 3. Notation used for 3 by 3 neighborhood about center pixel P.

a second sinusoidal filter to the output of the first filter, we found that the resulting composite filter was sensitive to a broader velocity range, had better S/N response and had a weaker response to clouds. Finally, a third decaying exponential (averaging) filter was applied to the output of the second filter to dampen the ringing intrinsic to sinusoids. The resulting composite triple temporal filter (TTF) is specified by five parameters: $F_1(P_1,l_1)F_2(P_2,l_2)F_3(l_3)$, where the subscripts refer to the first, second, and third filters, respectively. The optimization of these five parameters is the main subject of this work. Until now filter parameter values have been selected in conjunction with real data largely by trial and error.

The filter so far described is a one-dimensional filter; each pixel is filtered independently in time. We have found edge suppression for data with evolving strong cloud edges to be a useful spatial adjunct to the temporal algorithm. An obvious spatial discriminant between a true point target and a point-like cloud feature is the likely presence of the latter at an edge. A measure of the edge strength S_{MAX} associated with a given pixel can be taken as the maximum absolute value obtained from the set of 4 subtractions within the 3 by 3 neighborhood which straddle the pixel: $H_L - H_R$; $V_T - V_B$; $D_1 - D_3$; $D_2 - D_4$ (Fig. 3). We reduce the output of the first filter of the TTF by the edge factor $\lfloor S_{\mathrm{MAX}}/E \rfloor + 1$, where E is an edge parameter which scales the edge strength and $\lfloor\ \rfloor$ represents truncation to the next lower integer. Such a spatial adjunct is an effective discriminant unless the target is on several pixels, in which case it is typically closer and stronger and can survive reduction by the edge factor. E is the sixth parameter included in a given TTF specification.

Our early work focused on detecting point targets in evolving cloud clutter [4]. We developed and characterized the TTF specified by the following six parameters: $F_1(16,16)E(64)F_2(10,10)F_3(5)$. This "tracker" was implemented in hardware and was found to have good clutter suppression characteristics but limited signal to temporal noise response, particularly for slow targets. The shortcomings of the $F_1(16,16)E(64)F_2(10,10)F_3(5)$ TTF were supported by the results of a simulation of idealized slow targets embedded into Gaussian white temporal noise. We have developed TTF filters that have better S/N response characteristics and improved slow target detection performance, as reported recently [8]. However, to achieve reasonable clutter suppression, these filters required very strong edge suppression, i.e, small values of E, which cause targets in front of clouds to be lost.

Until now we have chosen the parameters largely from intuition and experience, incorporating certain reasonable constraints such as $P = l$ to simplify the search. The richness of the parameter space plus the quantity and variety of real data made such a semiempirical approach problematic, since we had hoped to find one tracker for all conditions and a wide variety of point targets. We next describe the process which enabled us to analyze the interaction of the data with the algorithm and to extract improved parameters.

III. OPTIMIZATION PROCESS

We sought to find an optimal set of parameter values for the TTF averaged over a representative sample of our extensive database. Our approach was to use an automated routine to find the six best parameters for each data set and then to analyze the resulting parameter sets to see if an average set could be identified or a correlation between parameters found.

Two PtSi IR cameras, both fabricated in our laboratory, were used to acquire the data. The focal plane array (FPA) of camera 1 was obtained from David Sarnoff Research Center; the FPA of the second camera was obtained from FLIR Systems Inc.. The format of both arrays is 320 by 244 pixels with square pixel sizes of 40 μm for the first and 24 μm for the second array. Data was taken with the first camera using a 50 mm and a 200 mm lens, while that taken with the second camera used a 100 mm and a 200 mm lens. These combinations gave total fields of view of 14.6, 4.4, 3.7 and 2.2 deg, respectively. Frame rate for each camera is 30 s^{-1} with standard interlace format. The performance of both cameras is similar to those previously characterized [9].

The data from the cameras was recorded to 12 bit precision using a digital cassette recording system or a laboratory-developed frame-grabber. Sequences of 95 consecutive frames were selected for algorithm evaluation and added to an archive of real-world test data on UNIX work stations. The requisite hardware and software for data recording, transfer, and algorithm development were developed in our laboratory.

The "optimum" filter parameter values for a given data set was determined from the Simplex routine [10]. A given data set consisted of about 3 s of consecutive frame data (95 frames) with one moving point target. For scenes with multiple targets, a separate optimization is run for each target. The optimization process was run on a representative subset of all our real data (some 22 data sets). At least one sequence of data from each day or evening 12 bit digital recording session was used.

The Simplex routine features an iterative algorithm which minimizes a selected function of a set of input variables. Our chosen variables were the 6 input parameters of a TTF filter and our chosen function was the clutter to signal (C/S) ratio of the TTF filtered output. C/S for each frame is taken as the ratio of the largest non-target-related pixel output to the largest target-related output. C typically stems from a cloud edge for cloud scenes or a noise fluctuation for clear sky scenes. The final C/S for a given filter parameter set is the average over the set of frames after reaching a steady-state result (steady-state is achieved by the last 40 frames). The optimization process works as follows. The initial parameters are sent to the Simplex routine; the TTF filtered output is calculated and its corresponding C/S is determined. The Simplex routine determines six new parameters. Once again, the TTF output and its corresponding C/S is calculated. This process is repeated until C/S is no longer changing within a predetermined limit or a set number of iterations has been reached. A 200 MHz Pentium-Pro-based computer takes about 30 min to complete one optimization.

We should emphasize that this application of the Simplex algorithm involves a very complex multiminimum function space in which some 7.4 million data points (320)(244)(95) are processed by a 6 parameter filter. The iterative minimum reached is a function of the starting parameters to a degree dependent upon the particular data set. We have run extensive optimizations with both the $F_1(35,10)E(64)F_2(35,10)F_3(10)$ TTF (identified previously by trial and error as having improved signal to noise response) and the previous baseline tracker: the $F_1(16,16)E(64)F_2(10,10)F_3(5)$ TTF as starting points. We present the results for the former as lower C/S ratios are generally achieved and a coherent picture emerges with this start, although we have no assurance of having reached a global minimum.

IV. ANALYSIS OF OPTIMIZATION RESULTS

Each optimization on a sequence of data gave a new set of "optimized" filter parameters as well as an improved C/S output value. We have examined the optimized filter parameters in the form of pairwise correlation plots. The correlations taken in conjunction with idealized filter responses are then used to elucidate the relationship among the scene clutter, the target velocity, and the filter parameters. Finally, we use the correlation plots to choose improved filter parameters.

A. Analysis of Correlation Plots

Fig. 4 shows a plot of the half-damping constant of the first filter l_1 versus the period of the first filter P_1 (correlation coefficient $r = -0.73$). Values of the clutter-dominated scenes are differentiated from the noise-dominated scenes. In the former, maximum nontarget filter response stems from cloud evolution. In the latter, the highest nontarget filter response stems from temporal noise fluctuations. A clear sky or a sky with stagnant or dim clouds are examples of noise-dominated scenes. The three problematic clutter points are numbered and are discussed later on. Note that, within the linear trend, there are distinct clusters of noise data points and clutter data points, the "centers" of which are indicated in Fig. 4 and are also discussed later. Fig. 5 shows a plot of the period of the second filter P_2 versus the period of the first filter P_1. A similar linear trend is observed, although the scatter is considerably greater ($r = -0.18$). l_2 and l_3 when plotted versus P_1 (not shown) scatter in value from 8 to 13 with a slight upward trend with period. There is a smooth linear correlation between edge E and the period of the first filter P_1 ($r = 0.95$) as is shown in Fig. 6. As P_1 increases, a larger E value is needed, i.e., weaker edge suppression (see Section II).

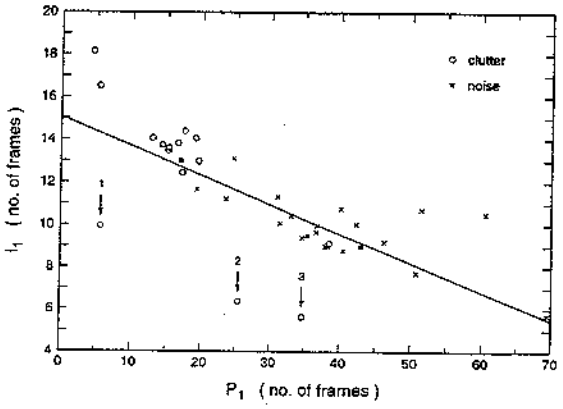


Fig. 4. Correlation of l_1 versus P_1. Data sequences where nontarget leakage is dominated by clouds (o) are distinguished from those dominated by temporal noise (×). Linear regression line here and in succeeding figures use minimum square distances perpendicular to line (not those along one of the axial directions). See text for discussion of points labeled 1, 2, 3 and the two black rectangular points.

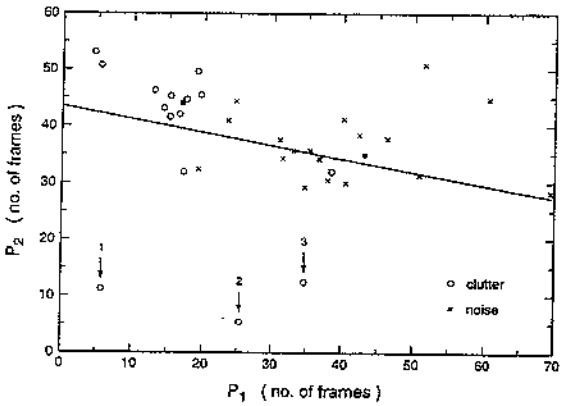


Fig. 5. As Fig. 4 for P_2 versus P_1.

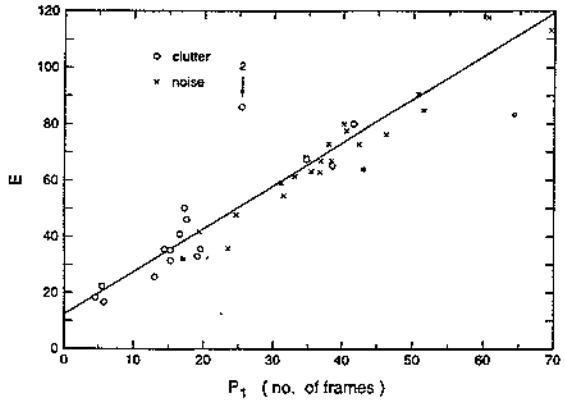


Fig. 6. As Fig. 4 for E versus P_1.

Although we anticipated some relationship between the parameters, the observed correlations were unexpected. We had expected that slower targets would be best detected by longer period filters, but instead found that target velocity is not the leading factor in determining the optimal filter parameters. The filters are nonlinear and thus we endeavored to understand the filter operation and the correlations by simulation.

The simulations were divided into three parts: targets, clutter, and temporal noise. For each of the simulations we examined the output of F_1, F_2, and F_3 (we did not simulate the edge filter). Ideal targets with velocities of 0.25, 0.1, and 0.05 p/f were simulated. Cloud clutter was modeled by a linear ramp that increased in intensity by one unit per frame. The noise was computer-generated white Gaussian noise.

Fig. 7 illustrates the general trends observed in our simulation. In Fig. 7(a) and 7(b) we show the output of F_1 to an ideal target moving at 0.1 p/f and the ideal clutter (ramp) for the parameter values $F_1(17,13)$ and $F_1(43,9)$ (chosen from the "centers" in Fig. 4). Not shown is the response of the two filters to temporal noise which is about 15% larger for $F_1(17,13)$ than for $F_1(43,9)$. While the response to the targets is somewhat greater for $F_1(43,9)$, its response to (ideal) clutter is also much greater. The trends seen in Fig. 7 generally hold up over the simulated range of velocity and after application of the full TTF algorithm leading us to the following broad conclusion. The simulations predict that the severity of the clutter will be the main driving force in determining the optimal parameter values of the first filter. The target velocity is a significant factor only in cases were the temporal noise dominates.

Returning to real data, we note that Figs. 8 and 9 support these predictions. In Fig. 8, P_1 is plotted as a function of clutter severity. The clutter severity is quantified using the standard deviation in time of each pixel over the entire image sequence [11]. Only the data from the cloud-clutter dominated scenes is plotted in Fig. 8. We see that the image sequences with the largest clutter values are strongly correlated with the

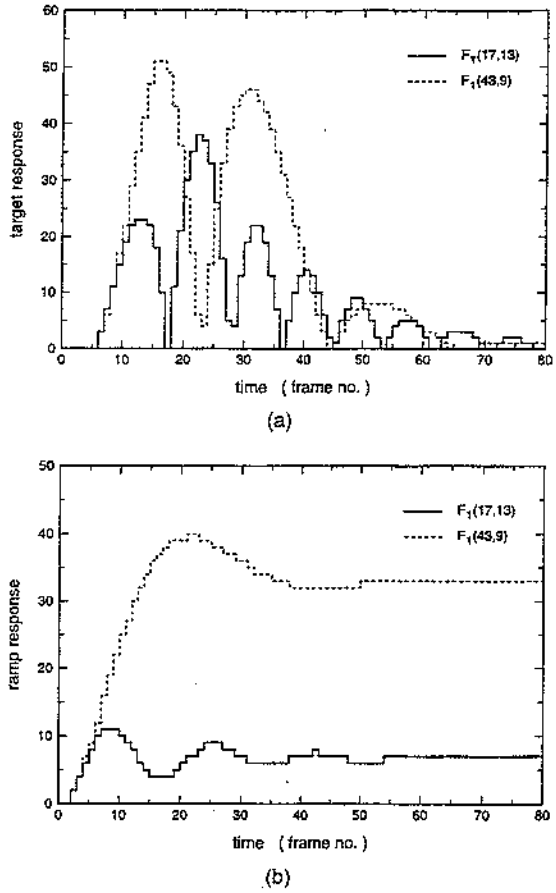


Fig. 7. Simulated temporal responses after application of single damped sinusoids, $F_1(17,13)$ and $F_1(43,9)$ to: (a) idealized target with velocity of 0.1 p.f, and (b) idealized cloud simulated by linear ramp.

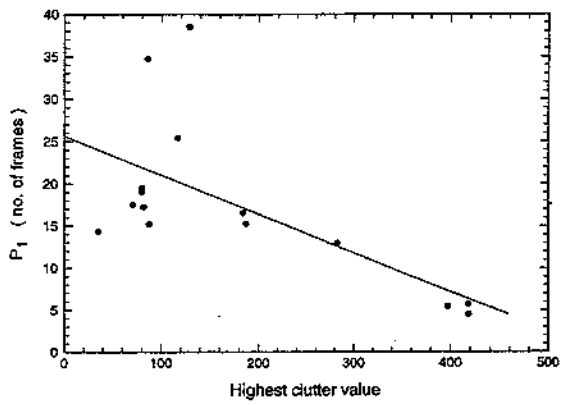


Fig. 8. Correlation of P_1 versus measure of severity of clutter evolution for clutter-dominated data sequences.

lowest values of P_1. Those image sequences with the smallest clutter values are not as strongly correlated with P_1.

In Fig. 9, P_1 is plotted versus target velocity. The target velocity is measured by dividing the number of pixels traversed by the number of frames for both horizontal and vertical directions and adding the two values in quadrature. Only the data from the noise-dominated scenes is plotted in Fig. 9.

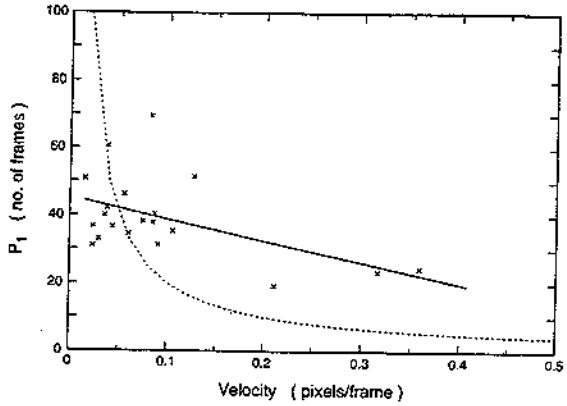


Fig. 9. Correlation of P_1 versus measure of target velocity for noise-dominated data sequences. Solid line is linear regression; dotted line is from simple model.

The image sequences with the largest velocities are correlated with the lowest values of P_1. The anticipated correlation of longer periods with slower targets is generally found (correlation coefficient of -0.57). The data used in this plot was largely acquired at night when weak, slow targets (≤ 0.1 p/f) were abundant and cloud evolution was absent. The predicted trend (dotted line) is from a simple model which equates the time width of the sinusoid lobes $P_1/2$ with that of reciprocal velocity $1/v$ in frames/pixel. Much of the lowest velocity data are well below the expected period; however, many weak, slow targets have jagged temporal profiles (due to camera motion) with segments which mimic the profiles of faster targets.

Further examination of ideal profiles for the sequence of component filters which constitute the TTF elucidate the internal dynamics of the filter and the influences constraining parameter choice after P_1 is selected. Shorter period P_1 values need a longer period P_2 (see Fig. 5) to capture multioutputs of the first filter. As the first period increases, the output concentrates in fewer peaks which are slightly narrower than the initial temporal profile, favoring a second filter slightly lower in P value than the first. The narrow ranges found for l_1 and l_2 values, generally from 8–14, are a trade-off between keeping the integrity of the first few sinusoid lobes (longer damping constants) and avoiding the excess noise which longer damping constants introduce. The dependence of the edge parameter on P_1 shown in Fig. 6 is a consequence of the edge filter enhancing performance in clutter and degrading performance in temporal noise. Small edge parameter values (strong filter output suppression) are required in heavy clutter and coincide with small P_1 values.

Finally, note the three numbered deviations from the correlation trends in Figs. 4 and 5. Point 1 has the difficult combination of very severe clutter and very slow target velocity (≈ 0.01 p/f) setting up conflicting requirements for P_1. Point 2 is in front of clouds. Point 3 remains unexplained. Note that in the correlations of Fig. 6, only point 2, requiring a milder edge suppression, remains anomalous. None exhibit anomalous behavior in the correlation with clutter (Fig. 8).

B. New Choices of Filter Parameters

The correlation plots served as a guide in choosing new filter parameters. To start the process with Fig. 4, we note that clutter-dominated data points tend to cluster toward one end of the range of values found and noise-dominated data points toward the other end. The choice of two new filters at the positions indicated at the solid rectangles in Fig. 4 (and the two following figures) is designed to span this range as well as favor the clusters. The first filter for cloud-dominated scenes (cloud tracker, (CT)) is taken as $F_1(17,13)$ and for noise-dominated scenes (noise tracker, (NT)) is taken as $F_1(43,9)$.

With P_1 fixed, we follow a similar philosophy in using Fig. 5 to choose P_2 as 44 for the CT and 35 for the NT. Plots not shown indicate narrow scatter for l_2 and l_3 and little trend with P_1 or P_2. The value of 10 is selected for both trackers for l_2. A value of 8 is chosen for l_3, because it is easier to implement in hardware. Finally the edge value strengths are chosen from Fig. 6 (shifted to the nearest power of two again for hardware convenience). In summary, the two new TTF filters are $F_1(17,13)E(32)F_2(44,10)F_3(8)$ for CT and $F_1(43,9)E(64)F_2(35,10)F_3(8)$ for NT.

V. PERFORMANCE OF IMPROVED FILTERS

The improved performance of the NT and CT filters, obtained by the optimization process, is detailed by a few typical examples after which metric plots for our total data base are shown.

A. Representative Examples

In this section, we illustrate some typical performance improvements by two example scenes from our data set. The presentation is based on input and output images. The two input images consist of a single frame of the scene with target locations indicated by rectangular outlines, and a linearly displayed image of the standard deviation of the intensity of each pixel over 95 temporal frames. The latter image serves as a relative measure of target versus clutter changes with time. Results are indicated by output images of filtered frames displayed on a linear scale, typically, the TTF-filtered result after 95 frames of data processing.

Example 1 is a challenging example of cloud clutter; there is one target moving at 0.2 p/f in the horizontal and 0.02 p/f in the vertical direction in clear sky among many brightly sunlit clouds moving

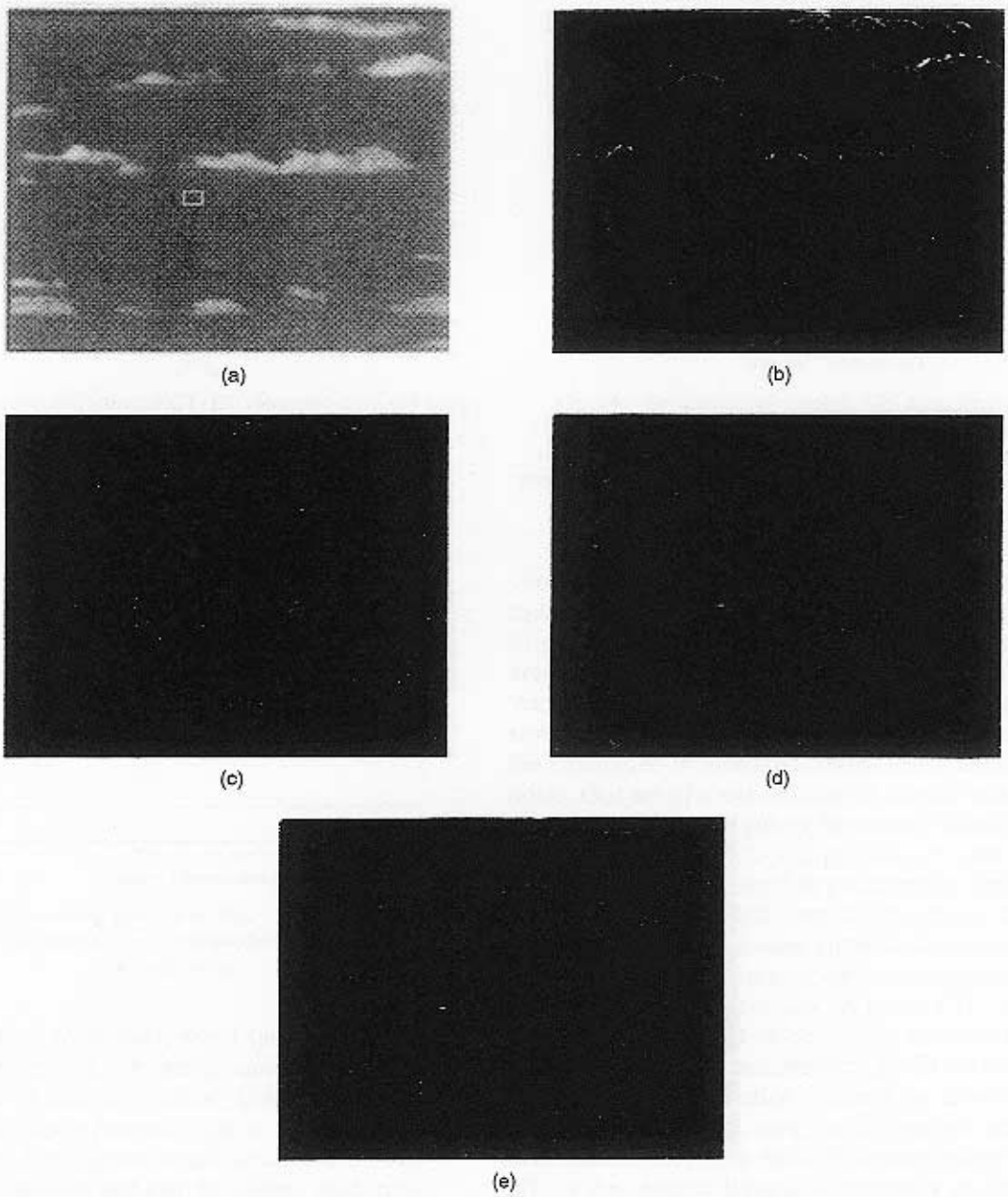


Fig. 10. Results of three tracking filters applied to daytime cloud scene. (a) Example input frame with point target indicated by white outline. (b) Temporal standard deviation after 95 input frames displayed as linear image, white outline indicates region of target movement over frames. (c) NT result. (d) CT result. (e) ET result.

horizontally about 0.03–0.04 p/f. The results of the following three filters are shown in Fig. 10: the two new tracking filters, CT and NT and our previous baseline filter, $F_1(16,16)E(64)F_2(10,10)F_3(5)$, designated as the empirical tracker (ET).

The NT result for such a scene is dominated by clutter; the highest target value is less than the highest cloud leakage value. The CT result suppresses the clouds and the target value is about three times the cloud leakage value. The ET result also brings the target above the cloud leakage, but only by 1.8 times. Increasing the edge suppression of the ET (not shown) slightly improves the results, but the CT result is still superior.

Our next example features a noise-dominated case with two targets. The "fast" target (left) is moving at $v_x = 0.33$, $v_y = 0.05$ p/f and the slow target (right) is moving at $v_x = 0.04$ p/f, $v_y = 0.02$ p/f. The same three filters were used as in the first example. The NT gave superior results as is shown in Fig. 11. The left target response was 7 times the largest noise value and the right target was 6 times the largest noise fluctuation. The CT and the ET were able to bring the targets above the noise, but to a far less extent than the NT. The CT results were 1.5 and 2.3 and the ET results were 1.7 and 2.6 times the largest noise fluctuations for the left and right targets, respectively.

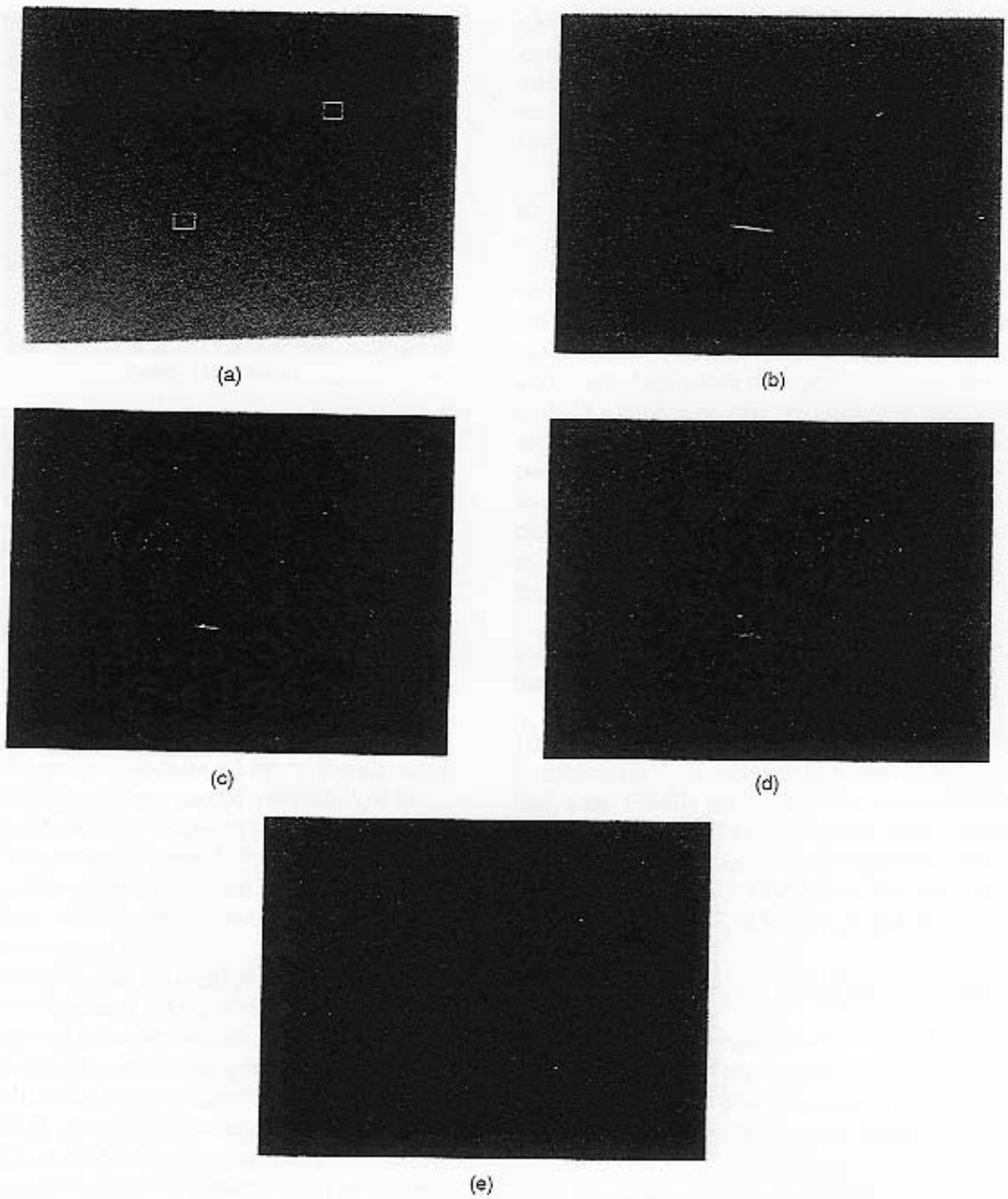


Fig. 11. Results of same three tracking filters applied to noise-dominated scene with two targets. Images correspond to those in Fig. 10, except no white outline is used in image (b). Peak in (b) at right edge, middle, is from noisy bad pixel.

The trends in these two examples: namely, improved signal-to-noise sensitivity of the NT filter, and improved signal-to-cloud-clutter performance of the CT filter, are representative of the overall trends as shown next.

B. Metric Plots

A common technique for evaluating the clutter suppression capability of an algorithm is in terms of signal-to-clutter (S/C) ratio gains expressed in dB [12] and defined by

$$\text{Gain} = 20\log\frac{(S/C)_{\text{out}}}{(S/C)_{\text{in}}}. \tag{3}$$

In earlier work [11], we introduced two new performance metrics more relevant to temporally processed real target data than the previous metrics which were designed for spatially processed data with embedded targets. Here, we use the second of these proposed metrics, the antimedian metric (AM), as it treats input and output symmetrically for both signal and clutter and is objective in its application. Essentially, the values required in the above equation are obtained by using the antimedian filter as a measure (not as a data filter) of the point target to clutter strength for both input frames and filtered output. Our form of the antimedian filter takes the absolute difference between the center pixel and the

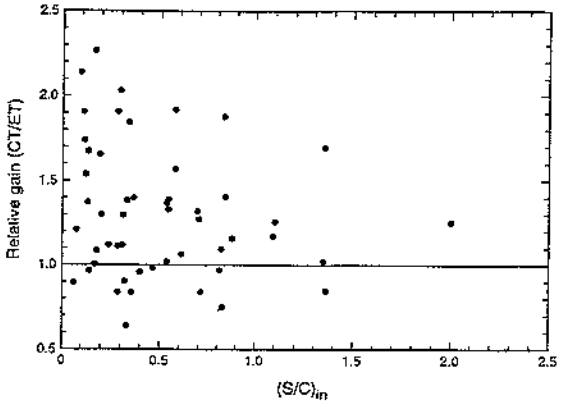


Fig. 12. Relative S/C gains of CT:ET computed from (3) using AM plotted versus $(S/C)_{in}$. Our total database of clutter-dominated data sequences is used in this plot.

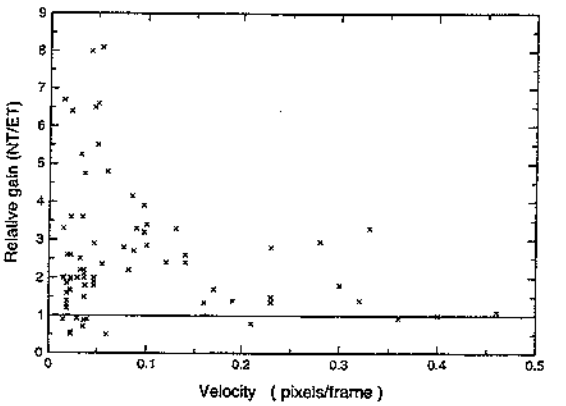


Fig. 13. Corresponding gains as in Fig. 12 for NT:ET plotted versus target velocities for our total database of noise-dominated data sequences.

median of the 5 by 5 block about (and including) this pixel. Using the AM, we can document the improved performance of the new tracking filters identified by the optimization process. This is described by the following two figures which encompasses our total target database, not just the subset used in the optimization process. In Fig. 12, gains computed from (3) for each target in the clutter-dominated scenes for CT and ET are plotted as CT/ET gain ratios versus $(S/C)_{in}$. As the clutter conditions become more severe, the superiority of the CT chosen from the optimization becomes more marked. The overall improvements are modest (typically a factor of 2 or less), not unexpectedly as the CT filter is largely a variation of the original ET choice (they differ mainly by $F_2(44,10)$ for the CT versus $F_2(10,10)$ for the ET). More dramatic improvements are achieved in noise-dominated scenes as shown in Fig. 13 where relative NT/ET gains are plotted versus target velocity. The superiority of the NT becomes very significant at lower velocity, as expected from its longer period filters.

The results in Fig. 13 on noise-dominated scenes reflect the **relative** signal-to-noise S/N gains of NT

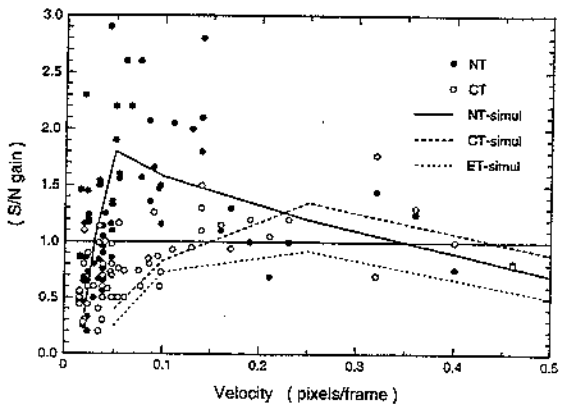


Fig. 14. Simulations of absolute S/N gains versus target velocities of NT, CT, and ET (shown as lines) compared with estimated S/N gains for noise-dominated data sequences after processing by the NT filter (closed circles) or by CT filter (open circles).

versus ET for specific real targets. It is of interest to simulate the **absolute** S/N gains of our filters. In particular, we wish to identify the range of velocities over which each filter is most effective. We avoid the complexities of characterizing and simulating real clutter and real targets by limiting the simulation to idealized targets in white Gaussian noise. Our set of idealized targets moves with one of the following purely horizontal velocities: 0.02, 0.03, 0.05, 0.1, 0.25, 0.5 p/f and are taken as one pixel in extent. Simulated target temporal profiles are triangular (ramp up and ramp down). One simulation run consists of 50 identical embedded targets added to 95 consecutive frames of computer-generated white noise at a selected input S/N. A given TTF filter is run and an equivalent output S/N is estimated from receiver operating characteristic (ROC) curves (curves of probability of detection versus false alarm rates) of typical output frames, using an estimation technique described in [13]. The ratio of the equivalent output S/N to the selected input S/N generates an estimated S/N gain for that velocity target. We have carried out this simulation for the NT, CT, and ET filters with results plotted as the lines in Fig. 14.

How well do our noise-dominated real target data reflect the trends in these simulated gains? We have computed real target S/N gains closer to the spirit of the simulation technique than is the AM, as follows. Since the simulations use equivalent output S/N values obtained from ROC curves which reflect noise at the distribution tail, we use maximum noise estimates for input and output in computing gains from (3). For the input, the temporal noise standard deviation of a single frame estimated from standard statistical methods was scaled by five to estimate N_{max}. The values obtained were consistent with previously reported camera characterization values [9]. It is difficult and somewhat arbitrary to determine the strength of a real target as it moves through sampled

imagery. As our measure of the input signal, we used the maximum change on a target-influenced pixel considering all the pixels traversed by the target over the 95 frames of a given data set. These two estimates provided $(S/N)_{in}$. We took $(S/N)_{out}$ as the largest target-to-noise pixel ratio found in any of the last 40 TTF filtered frames. Gains computed from these estimates for the NT and CT filters are plotted versus velocity for all our noise-dominated data scenes (shown as points in Fig. 14).

Within the considerable scatter found for the real data points, the broad features are nicely reproduced by the simulation predictions. These include the steep drop in sensitivity at the slowest velocities of the NT and the crossover of the NTs and CTs at about 0.25 p/f hinted at by the real (sparse) data. Many data points do have larger gains than predicted by the simulation. This could reflect the crude correspondence in temporal profiles between real and embedded targets. Each target has characteristic nuances making it different from a simulated counterpart. At the lowest velocities, in particular, the temporal profiles of real targets often have segments which mimic the profiles of faster targets. The simulation only was done for the empirical tracker ET to point out its poorer S/N gain sensitivity compared with the improved filters. In summary, the NT is most sensitive between 0.03–0.25 p/f, while the CT is most sensitive between 0.1–0.5 p/f. The useful velocity range of both filters is broader for targets in clutter provided the targets have sufficient signal-to-noise.

VI. CONCLUSIONS

In this paper, we have reviewed the input parameters of an easily implemented algorithm, the TTF, for detecting and tracking point targets in consecutive frames of IR image data from staring cameras. After application of the Simplex algorithm using real world data with targets of opportunity, we analyzed the correlations among the filter parameters which emerged from the optimization process. The correlations indicated that the value chosen for the first parameter (the period of the first of the three filters) constrains the other five parameters to narrow ranges. Examination of further correlations between this first parameter and clutter severity and target velocity, in conjunction with simulations of idealized filter responses to embedded targets, led us to the following conclusion. For scenes dominated by cloud clutter leakage, the first parameter choice correlates closely with clutter severity—shorter periods for more severe clutter. For scenes where temporal noise is the dominant nontarget leakage, a correlation of first parameter choice with target velocity becomes noticeable—shorter periods with faster targets.

The correlation plots served as a guide in choosing two new TTF filters: one dedicated to cloud-clutter-dominated scenes and the other to temporal-noise-dominated ones. We demonstrated the S/C improvement of the new cloud tracker and the S/N improvement of the new noise tracker as compared with the original baseline empirical tracker. We are currently running the two new forms of the TTF tracking algorithm in real time on a laboratory camera with impressive performance. In choosing the new parameters, we were very conservative. Suggestions have been made to extrapolate values of the parameters rather than choosing values from the clusters. This could further extend the S/N capability of the noise tracker which is already challenging the perception capability of human observers of the "live" IR imagery.

Further topics to be investigated with these filters include: spatial post-processing to enhance S/N sensitivity; and, ground and sea-glint clutter suppression. We plan to extend our data base to include more challenging cloud scenarios at night. In addition, work with a cooperative target is planned; this will remove some of the questions that arise when using targets of opportunity. Finally, the performance of the TTF algorithm on a moving platform needs to be investigated.

ACKNOWLEDGMENTS

The cameras used to collect the data were designed and fabricated by William Ewing, Steven DiSalvo, and John Mead of Rome Laboratory. We are grateful to the following colleagues: Virgil Vickers for software support, Steven DiSalvo for hardware support, and Linda Bouthilette for graphic assistance. We thank the following colleagues for helpful discussions and critical review of the manuscript: Virgil Vickers, William Ewing, and Paul Pellegrini.

REFERENCES

[1] Acetta, J. S. (1993)
Infrared search and track systems.
In S. B. Campana (Ed.), *The Infrared Electro-Optical Systems Handbook*, Vol. 5, *Passive Electro-Optical Systems*. Ann Arbor, MI: ERIM SPIE, 1993, ch. 4, 209–344.

[2] Reiss, D. B. (1994)
Spatial signal processing for infrared detection.
Proceedings of SPIE, **2235** (1994), 38–51.

[3] Caefer, C. E., Mooney, J. M., and Silverman, J. (1995)
Point target detection in consecutive frame IR imagery with evolving cloud clutter.
Proceedings of SPIE, **2561** (1995), 14–24.

[4] Mooney, J. M., Silverman, J., and Caefer, C. E. (1995)
Point target detection in consecutive frame staring infrared imagery with evolving cloud clutter.
Opt. Eng., **34** (Sept. 1995), 2772–2784.

[5] Barniv, Y. (1985)
Dynamic programming solution for detecting dim moving targets.
IEEE Transactions on Aerospace and Electronic Systems, **AES-21** (1985), 144–156.

[6] Barniv, Y. (1987)
Dynamic programming solution for detecting dim moving targets, Part II: Analysis.
IEEE Transactions on Aerospace and Electronic Systems, **AES-23** (1987), 776–788.

[7] Caefer, C. E., Silverman, J., Mooney, J. M., Tzannes, A. P., and Vickers, V. E. (1993)
Signal processing algorithms for point target detection in consecutive frame staring imagery.
Proceedings of SPIE, **2020** (1993), 93–107.

[8] Silverman, J., Mooney, J. M., and Caefer, C. E. (1996)
Temporal filters for tracking weak slow point targets in evolving cloud clutter.
Infrared Phys. Tech., **37** (1996), 695–710.

[9] Murguia, J. E., Mooney, J. M., and Ewing, W. S. (1990)
Evaluation of a PtSi infrared camera.
Opt. Eng., **29** (1990), 786–794.

[10] Caceci, M. S., and Cacheris, W. P. (1984)
Fitting curves to data.
BYTE (May 1984), 340–360.

[11] Silverman, J., Caefer, C. E., and Mooney, J. M. (1995)
Performance metrics for point target detection in consecutive frame IR imagery.
Proceedings of SPIE, **2561** (1995), 25–30.

[12] Chan, D. S. K., Langan, D. A., and Staver, D. A. (1990)
Spatial processing techniques for the detection of small targets in IR clutter.
Proceedings of SPIE, **1305** (1990), 53–62.

[13] Wager, N. (1994)
Automatic target recognition (ATR): Background statistics and the detection of targets in clutter.
Naval Postgraduate School, Monterey, CA, Master Thesis, Dec. 1994.

Charlene E. Caefer received a B.A. (1981) in biology from Merrimack College, North Andover, MA, a B.S. (1984) in electrical engineering from the University of Lowell, Lowell, MA, and an M.S. (1992) in systems engineering from the University of Massachusetts, Lowell.

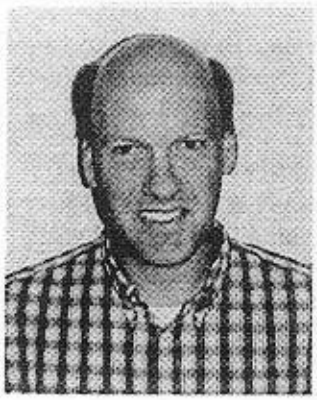
She joined Rome Air Development Center in 1984 and worked in the Radiation Effects Division. She is currently employed by the IR Sensor Technology Branch of the Air Force Research Laboratory. Her research interests have included radiation-induced defects in silicon and integrated circuits. Most recently her work has focused on designing algorithms for IR cameras as well as participating in field tests.

Ms. Caefer is a member of Sigma Xi and SPIE.

Jerry Silverman received the B.S. from Brooklyn College, Brooklyn, NY, in 1956 and the Ph.D. degree in physical chemistry from the Massachusetts Institute of Technology, Cambridge, MA, in 1960.

He joined the Air Force Research Laboratories at Hanscom AFB, Massachusetts, in 1962 and is currently employed in the IR Sensor Technology Branch. His professional activities have included research on the low-temperature acoustics of metals, x-ray crystal structure determination, discrete mathematics as applied to logic circuit design, the device physics of PtSi infrared cameras, and algorithms for the image and signal processing of IR data.

Dr. Silverman is a member of Phi Beta Kappa, Sigma Xi, and SPIE.



Jonathan M. Mooney received a B.A. in physics from the University of California at San Diego in 1981 and a Ph.D. in optical sciences from the University of Arizona, Tucson, in 1986.

He is currently employed by the Sensors Directorate of the Air Force Research Laboratory.

Dr. Mooney is a fellow of the Air Force Research Laboratory and of SPIE.