EFFECTIVE SMALL DIM TARGET DETECTION BY LOCAL CONNECTEDNESS CONSTRAINT

Keren Fu[A,B] *Kai Xie*[A] *Chen Gong*[A] *Irene Y.H. Gu*[B] *Jie Yang*[A*]

[A] Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China
[B] Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden

ABSTRACT

The main drawback of conventional filtering based methods for small dim target (SDT) detection is they could not guarantee sufficient suppression ability towards trivial high frequency component which belongs to background, such as strong corners and edges. To overcome this bottleneck, this paper proposes an effective SDT detection algorithm by using local connectedness constraint. Our method provides direct control for target size, ensure high accuracy and could be easily embedded into the classical sliding-window based framework. The effectiveness of the proposed method is validated using images with cluttered background.

Index Terms— Small dim target detection, Connectedness constraint, Region growing, Sliding window

1. INTRODUCTION

Small dim target (SDT) detection in infrared (IR) image is deemed as a key technique for applications such as infrared searching and tracking (IRST), accurate guidance for missiles and satellite remote sensing. Due to atmospheric transmission and attenuation, a long-distance imaging target usually accounts for only one or a few pixels in the imaging plane (may be modeled using *two-dimensional Gaussian* [1, 2, 3, 4]). Therefore conventional image features like corners and edges are nearly unavailable. Most traditional SDT detection methods [4, 5, 6, 7, 8, 9] solve the problem by implementing various kinds of filters upon input image. Methods as such can be categorized into two groups: direct high-pass filtering based (e.g. directional filters [4], LoG [5]) and oblique low-pass filtering based [6, 7, 8, 9]. The diagrams of the above two subcategories are summarized in the lower part of Fig.1.

In general, although the filtering based approaches have advantages such as simple implementation and good performance in homogeneous background, they suffer from two typical drawbacks in cluttered scenes. First, they could not totally handle the trivial high frequency component that belongs to background, such as corners and edges. These troublesome staffs may come from the sky-sea lines in the water-surface scene or the corners of a building in the land scene. It becomes nearly incapable of tuning appropriate threshold to filter the remaining false positives. Second, edges or corners may even gain unwanted enhancement that is even more obvious than that of targets. This fact brings obstacle into further identifying the targets according to their energy. The above disadvantages are illustrated by Fig.2 using a synthetic test image. Note the edge false positive of Max-mean [6] and TDLMS [9] and the corner false positive of Max-median [6] and Top-hat [7]. Notice that in Fig.2 among the conventional methods, the Max-median [6] performs best, since it can theoretically maintain the lines of four directions after filtering. Unfortunately, it would still degenerate the lines beyond four directions as well as corners.

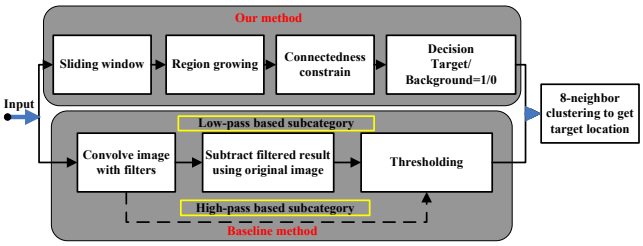

Fig. 1. Comparison of processing diagrams between our method and the conventional filtering based approaches.

This paper proposes an effective small dim target detection method based on local connectedness constraint. Our method could overcome the aforementioned drawbacks that traditional methods suffer from. Our motivation comes from the phenomenon that we have observed: *small dim targets usually present the signature of isolation, i.e. discontinuity from their neighbor regions*. This assumption is somewhat shown in the test image by the red rectangle in Fig.2. In contrast, the edges and corners could easily connect to their neighborhood that locates outside the yellow boxes. This connectedness character is an important signature that helps distinguish SDT from those high frequency distracters. Our method is based on the local connectedness constraint and accurately catches the target in Fig.2 (see the last image). The overview of our method is shown in the upper part of Fig.1.

*This research is partly supported by National Science Foundation, China (No: 61273258, 61105001), Ph.D. Programs Foundation of Ministry of Education of China (No. 20120073110018). Jie Yang is the corresponding author (email: jieyang@sjtu.edu.cn).

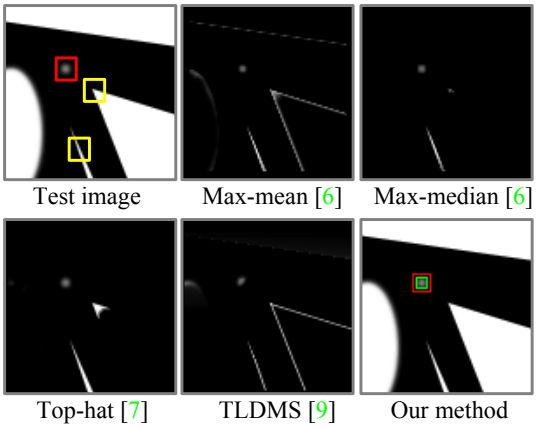

Fig. 2. Comparison of baseline methods (low-pass filtering based) with our method on a test image containing a SDT as well as strong corners and edges.

2. THE PROPOSED METHOD

To begin with, our method employs the pixel-wise sliding window strategy that is also widely used by the traditional filtering based methods. The output of our method is a decision map on which value "1" represents the corresponding window covers target and "0" for vice versa. In each sliding window, connectedness constraints are proposed to analyze the growed region. The following subsections will first introduce our connectedness constraints and then demonstrate our choice for region growing algorithm.

2.1. Connectedness Constraint

Since a SDT is usually an isolated blob that overshoots from its neighborhood whereas edges and corners are not, after implementing region growing, the growed region is supposed to be constrained (bounded) by a local window. Let the ith sliding window be W_i and the gray level and position of its central point be I_i and p_i. We propose to constrain the region's connectedness to be located only inside the window. The procedure is as following: in each window W_i, we take the central point p_i as seed (indicated by the red point in Fig.3) and run a region growing algorithm that is specially-designed. Based on connectedness, we judge whether the growed region is totally constrained by window W_i. If so, then this window is deemed as covering a target and its location p_i on the output decision map is set to value "1". Otherwise if the growed region touches the *window boundary*, the covered part is more likely to belong to homogenous background (second image in Fig.3) or edge/corner (third image in Fig.3). The corresponding location on the decision map is then set to "0".

Here, the function of the size of W_i should be highlighted. Since W_i is used to constrain the connectedness, its size determines the largest target (*upper bound*) our system could handle. If the target's size is equal or larger than the window size, according to the connectedness constraining strategy above, the region growing could reach the window boundary and then it would be discarded. Specifically, W_i is set to be a bit larger than the desired targets.

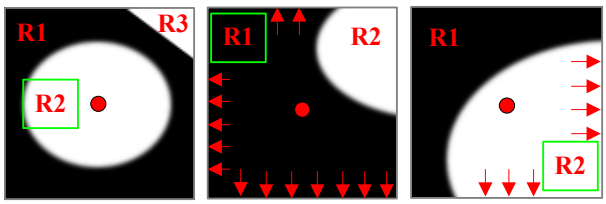

Fig. 3. The connectedness constraining strategy. The above images are three window examples from a synthetic input image, where $R_i, i = 1, 2, 3$ denote different regions in the windows. From left to right, the windows are respectively located on: an proper SDT candidate, homogenous background, edge/corner. The green boxes highlight the regions that contain central points while the red arrows indicate the growed regions touch the window boundaries. Only the first case will be identified as covering a target.

However, in some applications, too small target may belong to noise and should be rejected as well. To meet this demand, we extend our method by using double-window strategy, i.e. considering both outer window W^{out} and inner window W^{in}. The rationale for inner window is to provide *lower bound* of size for the target we desire. Hence, a double-window couple should simultaneously satisfy the following two conditions so as to be a "target-covering" window:

Condition (*i*): The connectedness region should be constrained inside the outer window W^{out}.

Condition (*ii*): The connectedness region should fill the inner window W^{in}.

Because some small size clutter parts fail to fill the inner window, they would be ultimately discarded. The double window strategy is illustrated using Fig.4, in which only Target 2 could be detected successfully. The advantage of using double windows is that it provides direct control for target's size, which could hardly be guaranteed by the typical filtering based methods. Although our method adopts nearly the same sliding window strategy that the filtering based methods usually do, in contrast, ours could easily filter out edges and corners using such connectedness constraints.

As the double windows offer the lower and upper bound (denoted as $[r^{in}, r^{out}]$, in which r^{in} and r^{out} are respectively the window radius for inner and outer window) for target's size, for each detected candidate, sometimes we want to know its exact size. Hence after a connectedness region satisfies the above two conditions, we continue the following step to obtain the actual target's size range: gradually shrink the gap, i.e. reduce the upper bound and increase the lower bound to make the following deviation E_i achieve its minimum:

$$E_i = \min_{\widehat{r_i^{in}}, \widehat{r_i^{out}}} |\widehat{r_i^{out}} - \widehat{r_i^{in}}| \tag{1}$$

$$s.t. \begin{cases} r^{in} \le \widehat{r_i^{in}} \le \widehat{r_i^{out}} \le r^{out} \\ Condition\ (i) \\ Condition\ (ii) \end{cases} \tag{2}$$

where $\widehat{r_i^{in}}$ and $\widehat{r_i^{out}}$ respectively represent the dilated and

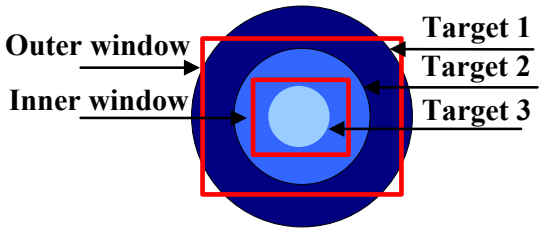

Fig. 4. Double window strategy. Since only Target 2 satisfies the both conditions, it would be the only one among these three targets that can be detected successfully.

eroded window radiuses. The narrow bound $[\widehat{r^{in}}, \widehat{r^{out}}]$ provides the estimated size range at position p_i for each target.

2.2. Region growing

Although some sophisticate region growing algorithms for weak boundary region discovery in gray images [11, 12, 13] can be employed, they usually get quite a few parameters to be tuned and are also computationally expensive, for we need to run such an algorithm for pixel-wise sliding windows. Instead we propose a simple but effective region growing strategy to meet this task. Our motivation is based on that SDT-like isolated blob is able to be popped out by using a threshold value that is a bit lower than its peak value (see W_1 in Fig.5). This local threshold value T_i^l for each window W_i is adaptively chosen as:

$$T_i^l = \min\{\frac{I_i + I_i^b}{2}, I_i - T^o\} \tag{3}$$

$$F_j = \begin{cases} 1 & \text{if } I_j \geq T_i^l \\ 0 & \text{otherwise} \end{cases} \quad s.t.\ p_j \in W_i \tag{4}$$

where I_i^b denotes the intensity average of W_i^{out}'s boundary pixels and is treated as a rough intensity estimation of the background nearby (owing to W_i^{out} is usually a bit larger than the desired targets). Using the average of I_i^b and I_i as threshold could highlight most prominent part of a SDT. *In case of small overshoot being detected* (see W_2 in Fig.5), overshoot threshold T^o is introduced as well. Only blobs with the overshoot that is larger than T^o could be detected. F_j indicates the resulting binary pixel. Three typical cases of applying this operation are illustrated using 1-D examples in Fig.5.

In Fig.5, W_1, W_2, W_3 represent three windows that cover a target candidate, a blob with too small overshoot and an edge respectively. In W_1, the central part of the target candidate could be directly segmented out (see the pink line labeled parts). For W_2, because the overshoot is too small to exceed T^o, the whole pixels in the window pop out. For the edge presented in W_3, only the right part is above the threshold. As analyzed before, since SDT has the signature of isolation, the corresponding binary region could be totally bounded by window W_1. In contrast since W_2 and W_3 cover nearly flatten and edge areas, after thresholding, the binary regions could *not* be

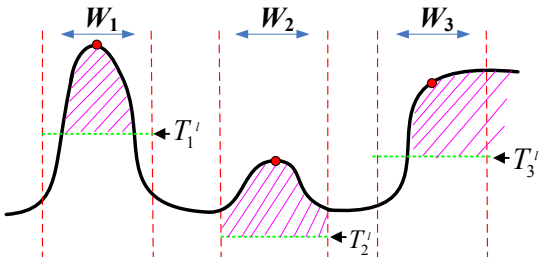

Fig. 5. Local thresholding operation in 1-D case. W_1, W_2, W_3 respectively represent three windows which cover a target candidate, a blob with too small overshoot and an edge. The red point in each window represents the corresponding central point. Here note for W_2, although the blob can be partially thresholded out by using the average of I_i and I_b, it is still too small to pass T^o. So T_2^l actually takes the second term in the minimum function.

Algorithm 1 Effective SDT Detection

Input: r^{in}, r^{out}, T^o and input IR image I;
Output: decision map D, location list L and size list S;

while for each sliding window-couple W_i^{in} and W_i^{out} **do**
  Binarize all pixels in the window W_i^{out} using T_i^l, resulting in binary map B;
  Run a binary region growing algorithm on the binary map B using central point as seed;
  if Condition (*i*) and Condition (*ii*) are satisfied **then**
    $D_i = 1$;
    Shrink the outer window radius $\widehat{r^{out}}$ and increase the inner window radius $\widehat{r^{in}}$ to estimate actual target size range $U_i = [\widehat{r^{in}}, \widehat{r^{out}}]$ (U_i is also an interval);
  else
    $D_i = 0$
  end if
end while
Run an 8-neighbour connectedness detection on D;
while for each foreground region R_j **do**
  Insert the region center location into L;
  The range $\bigcap_k U_k, p_k \in R_j$ is inserted into S;
end while

constrained totally inside the window and is able to connect to the *window boundary* easily. Hence after thresholding, we only focus on the positive region (value equals 1) in each window and take the window's central point as seed, based on which a binary region growing algorithm (Flood Fill) [10] is applied. Practically, this specially-designed region growing method works well and is also more time efficient than those in [11, 12, 13].

In general, our method gets three parameters to be determined, respectively are inner and outer window size r^{in} and r^{out}, and overshoot threshold T^o. When they are set according to specific demands, the sliding window based strategy is used to detect target candidates. Note that since our method guarantees direct size range control, it is naturally multi-scale. The **pseudo-code** for our technique is shown in Algorithm 1:

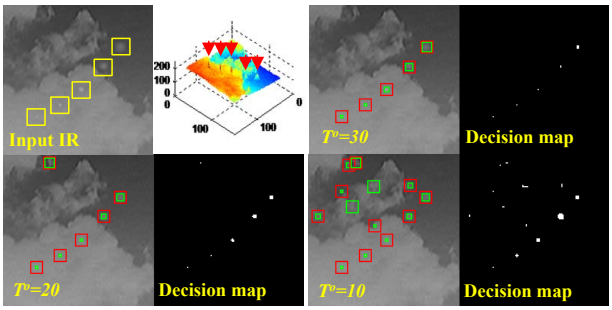

Fig. 6. The effect of changing T^o. The upper left sub-image shows the original image with cluttered cloud background and its 3D lateral view. Five targets are labeled in yellow boxes. Their sizes are respectively 1×1, 3×3, 5×5, 7×7 and 9×9 pixels. The remaining three sub-images show results under varied T^o. The size of red rectangles equals r^{out} and the size of green rectangles equals $\widehat{r^{out}}$ (the estimated target size). Note that as T^o becomes smaller, more and more false positives are generated.

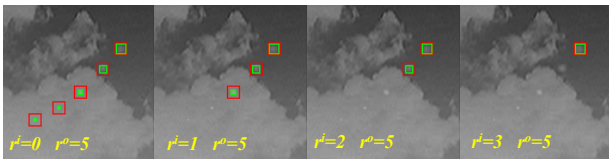

Fig. 7. Tuning r^{in} and r^{out} can control the target's size in Fig.6.

3. EXPERIMENTS AND RESULTS

Experiment with threshold T^o: T^o may determine the energy gap between the window central point and the local threshold. In order to successfully detect the desired targets, this gap should not be larger than the extent that targets overshoot out from their local neighborhood. Otherwise, the targets' surroundings will also be thresholded out according to (3) and (4) and cannot be constrained by the outer window, leading to entire target being rejected. Hence T^o should keep smaller. On the other hand, if T^o is too small, it will cause more false positives, for a very small overshoot would be detected as target. The effect of changing T^o is shown in Fig.6, where the r^{in} and r^{out} are set to 0 and 5 respectively. In practice, T^o is trained empirically through grid search using background images without any target and also images with manually labeled targets to achieve a trade-off between false positive and false negative.

Experiment with scale selection: Our method provides direct control on target's size by tuning r^{in} and r^{out}. Fig.7 shows the experimental results. As the lower bound r^{in} increases, more and more smaller targets are rejected.

Comparison with the baseline methods on detecting multi-scale targets: The top two rows of Fig.8 shows the results of conventional filtering based methods including max-mean [6], max-median [6], top-hat [7] and TLDMS [9]. Their window size is set as two times the maximum target size [4] to achieve good performance. It is obvious that in such cluttered background, traditional approaches could not offer sufficient suppression ability towards cloud edges and corners. Their

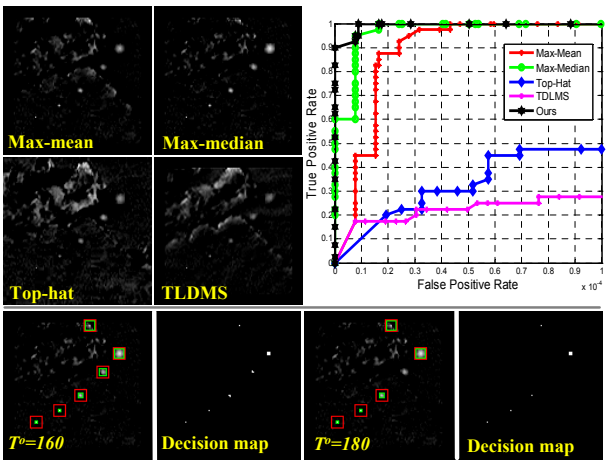

Fig. 8. Both visual (top left) and quantitative (top right) comparisons with baseline methods. The last row shows that Max-median [6], which is the best among traditional approaches, enhances both targets as well as corners. When we use it as pre-processing for our method and then gradually increase T^o, a corner still remains even after the 7×7 target is rejected.

results contain large amount of noises and many of them are also target-like blobs (most happens on corners). Using a post-thresholding may lead to lots of ambiguous detections. The top right graph in Fig.8 shows the ROC curves [4] on tens of test IR images containing about fifty targets with varied sizes. By using varied threshold value to segment the resulting maps, different false positive and negative rates are obtained. Note for our method, we switch T^o to adjust performance. It is observed from Fig.8 our method achieves the best ROC curve and the highest AUC (Area Under ROC Curve) score. Top-hat [7] and TDLMS [9] perform worst due to their weak background suppression ability (Fig.8).

The last row in Fig.8 shows experiment where we use max-median [6] as pre-processing since such filtering based methods seem to be able to enhance targets and suppress background. We gradually increase T^o to filter out the false positives one by one. Unfortunately, the result is that a desired target with size 7×7 is first rejected when there still remains a false positive. This implies that after applying max-median filtering, some corners may gain unwanted enhancement which is even more obvious than targets themselves.

4. CONCLUSION AND FUTURE WORK

This paper proposes an effective algorithm for SDT detection by local connectedness constraint. Two constraint conditions are imposed to analyze the growed region generated by using local thresholding. Double-window strategy guarantees both lower bound and upper bound of size for desired targets. Comparing with conventional methods, our method could pick up targets from cluttered scene more accurately.

As our method has not considered the intensity gradient, in the future, we plan to incorporate more cues like distribution fitness and gradient extent to enhance the performance.

5. REFERENCES

[1] J.P. Ardouin. “Point source detection based on point spread function symmetry”, Optical Engineering, 32(9), pp. 2156-2164, 1993.

[2] T. Soni, J.R. Zeidler, W.H. Ku. “Performance evaluation of 2-D adaptive prediction filters for detection of small objects in image data”, IEEE Trans. on Image Processing, 2(3), pp. 327-340, 1993.

[3] D.S.K. Chan, D.A. Langan, D.A. Staver. “Spatial processing techniques for the detection of small targets in IR clutter”, in Proc. SPIE, 1990

[4] S. Kim, J. Lee. “Scale invariant small target detection by optimizing signal-to-clutter ratio in heterogeneous background for infrared search and track”, Pattern Recognition, 45(1), pp. 393-406, 2012.

[5] S. Qi, J. Ma, C. Tao, C. Yang, J. Tian. “A Robust Directional Saliency-Based Method for Infrared Small-Target Detection Under Various Complex Backgrounds”, IEEE Geoscience and Remote Sensing Letters, 10(3), pp. 495-499, 2013.

[6] S.D. Deshpande, M.H. Er, V. Ronada, P. Chan. “Max-mean and max-median filters for detection of small targets”, In Proc. SPIE, 1999.

[7] V.T. Tom, et al. “Morphology-based algorithm for point target detection in infrared backgrounds”, In Proc. SPIE 1993.

[8] R. Nitzberg, et al. “Spatial filtering techniques for infrared (IR) sensors”, In Proc. SPIE 1979.

[9] Y. Cao, R.M. Liu, J. Yang. “Small target detection using two-dimensional least mean square (TDLMS) filter based on neighborhood analysis”, International Journal of Infrared and Millimeter Waves, 29(2), pp. 188-200, 2008.

[10] R.C. Gonzalez, R.E. Woods. “Digital Image Processing”, 2nd Edition, Prentice Hall, Upper Saddle River, New Jersey, 2002.

[11] Y. Qiao, J. Yang. “Adaptive Region Growing Based on Boundary Measures”, in Proc. ICONIP, 2011.

[12] L. Zhou, Y. Qiao, L. Zhou, J. Yang, Y. Gao. “An Active Contour Model based on Multiple Boundary Measures”, in Proc. ICIP, 2013.

[13] S. Lankton, A. Tannenbaum. “Localizing Region-Based Active Contours”, IEEE Transactions on Image processing, 17(11), pp. 2029-2039, 2008.