







Article

Hardware-Accelerated Infrared Small Target Recognition Based on Energy-Weighted Local Uncertainty Measure

Xiaoqing Wang ^{1,2}, Zhantao Zhang ³, Yujie Jiang ³, Kuanhao Liu ³, Yafei Li ⁴, Xuri Yao ¹, Zixu Huang ², Wei Zheng ², Jingqi Zhang ^{3,*} and Fu Zheng ^{2,*}

- ¹ Center for Quantum Technology Research and Key Laboratory of Advanced Optoelectronic Quantum Architecture and Measurements (MOE), School of Physics, Beijing Institute of Technology, Beijing 100081, China; 3120225785@bit.edu.cn (X.W.); yaoxuri@bit.edu.cn (X.Y.)
- ² Key Laboratory of Electronics and Information Technology for Space Systems, National Space Science Center, Chinese Academy of Sciences, Beijing 100090, China; huangzixu22@mails.ucas.ac.cn (Z.H.); zhengwei@nssc.ac.cn (W.Z.)
- ³ School of Integrated Circuits and Electronics, Beijing Institute of Technology, Beijing 100081, China; 3120231354@bit.edu.cn (Z.Z.); 3120231367@bit.edu.cn (Y.J.); 3220231954@bit.edu.cn (K.L.)
- ⁴ School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China; 1120211326@bit.edu.cn
- * Correspondence: zhangjq@bit.edu.cn (J.Z.); fzheng@nssc.ac.cn (F.Z.)

Abstract: Infrared small target detection is a key technology with a wide range of applications, and the complex background and low signal-to-noise ratio characteristics of infrared images can greatly increase the difficulty and error rate of small target detection. In this paper, an uncertainty measurement method based on local component consistency is proposed to suppress the complex background and highlight the detection target. The method analyzes the local signal consistency of the image. It then constructs a confidence assignment function and uses the mutation entropy operator to measure local uncertainty. Then, the target energy information is introduced through an energy-weighting function to further enhance the signal. Finally, the target is extracted using an adaptive threshold segmentation algorithm. The experimental results show that the algorithm can effectively detect small infrared targets in complex backgrounds. And, the algorithm is at the leading edge in terms of performance; the processing frame rate can reach 3051 FPS (frame per second), 96 FPS, and 54 FPS for image data with a resolution of 256×256 , 1920×1080 , and 2560×1440 , respectively.

Keywords: infrared; small target detection; FPGA



Citation: Wang, X.; Zhang, Z.; Jiang, Y.; Liu, K.; Li, Y.; Yao, X.; Huang, Z.; Zheng, W.; Zhang, J.; Zheng, F. Hardware-Accelerated Infrared Small Target Recognition Based on Energy-Weighted Local Uncertainty Measure. *Appl. Sci.* **2024**, *14*, 8798. <https://doi.org/10.3390/app14198798>

Academic Editor: Pedro Couto

Received: 8 September 2024

Revised: 26 September 2024

Accepted: 27 September 2024

Published: 30 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background

Infrared small target detection identifies and tracks a few pixels that make up a small target in infrared images. It finds use in aerospace, ocean monitoring, and disaster warning. This technology has important application needs in aerospace, ocean monitoring, disaster warning, and other fields. However, infrared small target detection faces significant challenges due to its characteristics. Small targets consist of few pixels with low signal strength, leading to a very low signal-to-noise ratio. As image distance increases, target size and signal strength decrease further. Additionally, cluttered backgrounds with similar characteristics make distinguishing targets difficult. To address these challenges, the current research focuses on algorithms for complex background suppression and small target enhancement for infrared small targets.

1.2. Related Work

Infrared small target recognition can be divided into single-frame image processing and multi-frame image processing. Li [1] used multi-frame image processing and proposed

a Track-Before-Detect method for target recognition through candidate extraction, object enhancement, and trajectory tracking. Wu [2] used single-frame image processing and proposed a target-detection algorithm based on an adaptive threshold. In general, single-frame processing is more concise and multi-frame processing can better handle background information. In the way of implementation, some scholars use CPU to realize infrared small target detection. Chen [3] used high-pass filtering based on the difference of local neighborhood gray values for background suppression and achieved target recognition and tracking by Unscented Kalman filtering on an FPGA and DSP. Bo [4] implemented a local contrast method inspired by the human visual system and derived kernel models on an FPGA. Zhang [5] implemented morphological filtering and local threshold adaptive filtering on an FPGA and ARM. In addition, some scholars adopt the way of hardware and software cooperation to achieve target detection. Roy [6] proposed a kind of hardware and software codesign for real-time inspection. Initial target positioning is accelerated using the FPGA, followed by in-depth processing of the region of interest through software. In addition to traditional image-processing algorithms, some scholars have also tried to use neural networks and other methods to realize infrared small target recognition. Hu [7] transforms the small-target-detection task into an image-to-image conversion problem, using convolutional neural networks to convert infrared small targets into target maps.

1.3. Motivation and Contribution

The infrared small target and its background in the image correspond to the fluctuating background, noise, and target signal in different regions, bringing uncertainty to the spatial characteristics of the observed data. Due to the uncertainty of observation caused by the spatial variation of each component, the previous work did not analyze the characteristic uncertainty of dim target detection, which can significantly impact the detection of infrared small targets. Moreover, the parallel processing capability and reconfigurability of the FPGA enable it to exhibit a superior performance in infrared small target detection.

In this paper, considering the relationships between signals of different components and between signals of the same component, we propose the Energy-weighted Local Uncertainty Measure (ELUM) based on signal component consistency. This method is especially suitable for the rapid detection of small single-frame infrared targets in complex scenes on the FPGA. Single-frame image processing is mainly based on the spatial-domain information of the image and utilizes single-image data for target recognition, while multi-frame image processing introduces time-domain information on this basis and utilizes multi-frame images with temporal continuity for target recognition. Multi-frame image processing can obtain the target's motion data with the help of spatio-temporal information to analyze and predict its motion trajectory, and this method is more effective when the background is more similar to the target [8]. However, when the target moves too fast or the background is not fixed, the effect of multi-frame image processing will be greatly affected [9]. In addition, the computational complexity of multi-frame image processing is large, and the computational efficiency is greatly limited. Therefore, single-frame image processing has a wider range of application scenarios. The method used in this paper adopts single-frame image processing, which has great advantages in processing efficiency and can better meet the efficiency and real-time needs of target recognition. The main contributions of this paper are as follows:

1. This paper first analyzes the characteristic uncertainty of dim target detection and emphasizes its critical importance for enhancing the performance of infrared small target detection.
2. This paper introduces a method that constructs a local consistency structure window to analyze the consistency of signal components within a small target's detection space, which is particularly effective in complex backgrounds or under low signal-to-noise ratio conditions.
3. This paper employs a variable entropy operator to measure uncertainty in the local region, which is essential for improving the robustness of small target detection.

The algorithm further incorporates an energy-weighting function designed with a Gaussian matching filter.

4. A classical adaptive threshold segmentation algorithm is used to extract the target, demonstrating its capability to process complex backgrounds using real data.

2. Background and Basic Principle

2.1. Analysis of Imaging Characteristics for Small Targets

Photonic imaging is a complex process subject to interference from multiple factors. The degradation and deterioration of images are inevitable at various stages of the entire imaging system chain [10]. Specifically, this section simplifies the analysis of the full-chain photonic imaging system, describing the process as follows: target signals are transmitted through atmospheric radiative transfer and modulated by the optical system before reaching the focal plane of the remote sensing detector. The detector then integrates and samples these signals to obtain digital image signals [11].

In addition, the interaction between atmospheric radiation and the optical system inevitably leads to significant degradation of the target signal. Due to the diffusion of light energy, the optical system acts as a spatial low-pass filter during imaging, filtering out high-frequency components of the signal while retaining the lower frequencies. The imaging produced by the optical system is often not ideal [12,13].

The Point Spread Function (PSF) of an optical system describes how a point source's energy spreads on the image plane. This is often represented using functions like the Bessel or Gaussian function, noted as PSF_{opt} .

The principle of a PSF imaging simulation involves convolving the original two-dimensional light-distribution image with the optical system's PSF to obtain the corresponding 2D light distribution of the optical simulation image. The mathematical expression for this operation is

$$I(x, y) = O(x, y) \times PSF_{opt}(x, y) \quad (1)$$

In real-world cases, fast-moving weak and small targets show noticeable motion relative to the detector. This motion causes energy changes, leading to variations in the target image's diffusion through the imaging chain [14]. Figure 1 illustrates an example of the image blur for a weak and small moving target.

Assume the radius of the blur spot of the target on the focal plane is r pixels and the number of pixels occupied by the blur spot is N_0 . After t seconds of exposure, due to relative motion, the weak and small moving target will experience motion blur, causing the number of pixels occupied in the image to change:

$$dx = \frac{v \cdot t}{\xi} (\text{pixel}) \quad (2)$$

$$N = N_0 + 2r \cdot dx = N_0 + \frac{2r \cdot v \cdot t}{\xi} \quad (3)$$

Here, t represents the relative displacement of the target during the exposure time, measured in units of pixel count; N represents the number of pixels occupied by the target under motion blur; v is the speed of motion; and ξ is the pixel size.

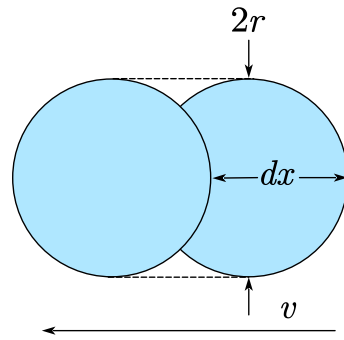


Figure 1. Example diagram of image spot of weak and small moving target.

2.2. Analysis of Background Characteristics in Complex Scenes

Interference elements from various scenes reduce the visibility of targets and increase the difficulty of detecting weak and small targets. The main challenges associated with scenes arise from their irregularity, dynamics, and potential visual similarity to the objects of interest. Given an optical image I of size $M \times N$, the variance of its grayscale values measures the severity of changes in the grayscale distribution. This variance is defined as follows:

$$\sigma^2 = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N |I(i, j) - \mu|^2 \tag{4}$$

The smoothness (R) of an image measures the smoothness and coherence of the image background. A higher value of R indicates a smoother image. The smoothness is defined as follows:

$$R = 1 - \frac{1}{1 + \sigma^2} \tag{5}$$

Entropy can effectively measure the average information content in an image. A larger entropy value indicates a more complex image, which is more likely to contain complex background components. In contrast, a smaller value suggests a more uniform background. Assuming the given image has a total of N_g gray levels, and the probability of the gray level value x appearing is p_x , the image entropy is defined as follows:

$$E = - \sum_{x=0}^{N_g-1} p_x \log p_x \tag{6}$$

Additionally, an image consistency metric can be computed to measure the smoothness of the image. A lower consistency value indicates greater complexity and irregularity in the image. The consistency is defined as follows:

$$U = \sum_{x=0}^{N-1} (p_x)^2 \tag{7}$$

The rank of an image can represent the richness of information, redundancy, and noise present in the image. To quantitatively perceive the complexity of an image from a global perspective, we introduce a method based on the rank-deficiency ratio of the image matrix. For image data I of dimensions $M \times N$, the Rank-Deficiency Rate (RDR) is defined as follows:

$$RDR = \frac{\#(Q > thro)}{\min(M, N)} \tag{8}$$

Here, Q represents the singular values of matrix I and $thro$ denotes the tolerance value for determining the rank of the matrix. $\#(Q > thro)$ represents the count of singular values in matrix I that are greater than $thro$. The RDR quantifies the complexity of the image on a global scale. However, the inherent difficulties in searching for or extracting objects

of interest from the image background are influenced by many local factors [15]. Local edges, contours, and other texture information in images significantly affect target-detection results. Therefore, it is necessary to consider multiple descriptive factors when qualitatively and quantitatively analyzing image complexity [16]. Utilizing research on image complexity analysis, the Gray-Level Co-occurrence Matrix (GLCM) can provide texture features and multi-dimensional parameters that describe scene complexity. However, these feature parameters may overlap and be redundant. In this study, four easily calculable and less correlated texture parameters—energy, contrast, Inverse Difference Moment (IDM), and correlation—are selected to assess the complexity of local scenes.

Considering an $M \times N$ scene I with N_g gray levels, let (i_1, j_1) and (i_2, j_2) be two pixels in scene I , where they are located at a distance d along direction θ . The GLCM of the scene is calculated as follows:

$$P(x, y, d, \theta) = \#\{(i_1, j_1), (i_2, j_2) \in M \times N | I(i_1, j_1) = x, I(i_2, j_2) = y\} \tag{9}$$

The Angular Second Moment (ASM) is used to describe the uniformity of the background distribution. A lower ASM value indicates that the elements are concentrated near the diagonal of the GLCM, suggesting a more uniform gray-level distribution and finer texture. Conversely, a higher ASM value indicates an uneven gray-level distribution, implying a coarser texture:

$$ASM = \sum_{x=0}^{N_g-1} \sum_{y=0}^{N_g-1} P(x, y, d, \theta)^2 \tag{10}$$

Contrast (CON) is a statistical measure that describes the variation or roughness of texture and reflects the clarity of the image texture. In our application, the clearer the texture details and the more pronounced the contrast differences, the more likely they are to cause significant interference with target recognition:

$$CON = \sum_{x=0}^{N_g-1} \sum_{y=0}^{N_g-1} (x - y)^2 P(x, y, d, \theta) \tag{11}$$

Correlation (COR) is used to measure the similarity of the GLCM elements in either the row or column directions. When rows or columns are similar, the COR value is higher, indicating lower background complexity:

$$COR = \sum_{x=0}^{N_g-1} \sum_{y=0}^{N_g-1} \frac{(x - \mu_1)(y - \mu_2)}{\delta_1 \delta_2} \tag{12}$$

Here, μ_1 and μ_2 represent the means of the normalized GLCM elements along the row and column directions, respectively. δ_1 and δ_2 denote their variances.

The IDM is a statistical feature parameter that reflects the local texture of the scene. When the IDM value is larger, the textures in different regions of the background scene are more uniform:

$$IDM = \sum_{x=0}^{N_g-1} \sum_{y=0}^{N_g-1} \frac{P(x, y, d, \theta)}{1 + (x - y)^2} \tag{13}$$

In summary, for any scene image, eight parameters can be extracted and combined into a background feature vector, as shown below:

$$BF = [RDR, R, E, U, CON, ASM, COR, IDM] \tag{14}$$

2.3. Analysis of Target-Complex Background Coupling

Background clutter and noise may interfere with the radiative diffusion of the target energy signal, causing changes in the energy distribution, which are described here as the result of the interaction between the target and the background. The interaction between the target and the background is mainly reflected in the spatial and temporal changes in energy and distribution, including changes in the energy value in the image element when the background is occluded by the target and changes in the local area perturbation due to the target entering different background regions [17].

The SCR used to evaluate weak targets in complex scenes is based on the difference between the target and the background, defined as follows [18]:

$$\text{SCR} = \frac{\mu_T - \mu_B}{\sigma_B} \quad (15)$$

In this equation, μ_T and μ_B refer to the average grayscale signal of the target and background regions, respectively, and σ_B is the standard deviation of the grayscale signal of the background region.

The standard evaluation of the signal-to-heterodyne ratio places the background in the global field of view and ignores local significance [19]. Therefore, a localized signal-to-heterodyne ratio is introduced on top of the signal-to-heterodyne ratio, which describes the difference between the target and the background in a localized region [20]:

$$\text{LSCR} = \frac{\mu_T - \mu_{LB}}{\sigma_{LB}} \quad (16)$$

In this equation, μ_{LB} refers to the average grayscale signal of the background region within a specified range L near the target and σ_{LB} is the standard deviation of the grayscale signal of the corresponding background region.

The Target Significant Rate (TSR) is a statistical parameter that reflects the significance of the target in the whole scene; generally speaking, the value of the TSR is between 0 and 1, and when the TSR equals 1, it indicates that the target has the highest energy in the whole scene:

$$\text{TSR} = \exp\left(\frac{\mu_T - \mu_B}{\max(B) - \mu_B} - 1\right) \quad (17)$$

In this equation, $\max(B)$ is the maximum gray value within the global background region B.

In summary, for the optical data of any scene, the following target feature vectors are adopted in this paper as the main quantitative metrics for target saliency assessment as follows:

$$\text{TF} = [\text{TSR}, \text{SCR}, \text{LSCR}] \quad (18)$$

Figure 2 shows a schematic of the signal information of a typical target in a real scenario.

The first step of target detection in a complex scene is to identify all regions of interest in the image, including the actual target region, as well as interfering regions that are similar to the target. These interfering regions show high feature similarity with the actual target, which may lead to confusion when recognizing the real target. In this study, a target-confusion-element assessment metric is proposed in anticipation of assessing the degree of difficulty of target detection from the perspective of confusable elements [21].

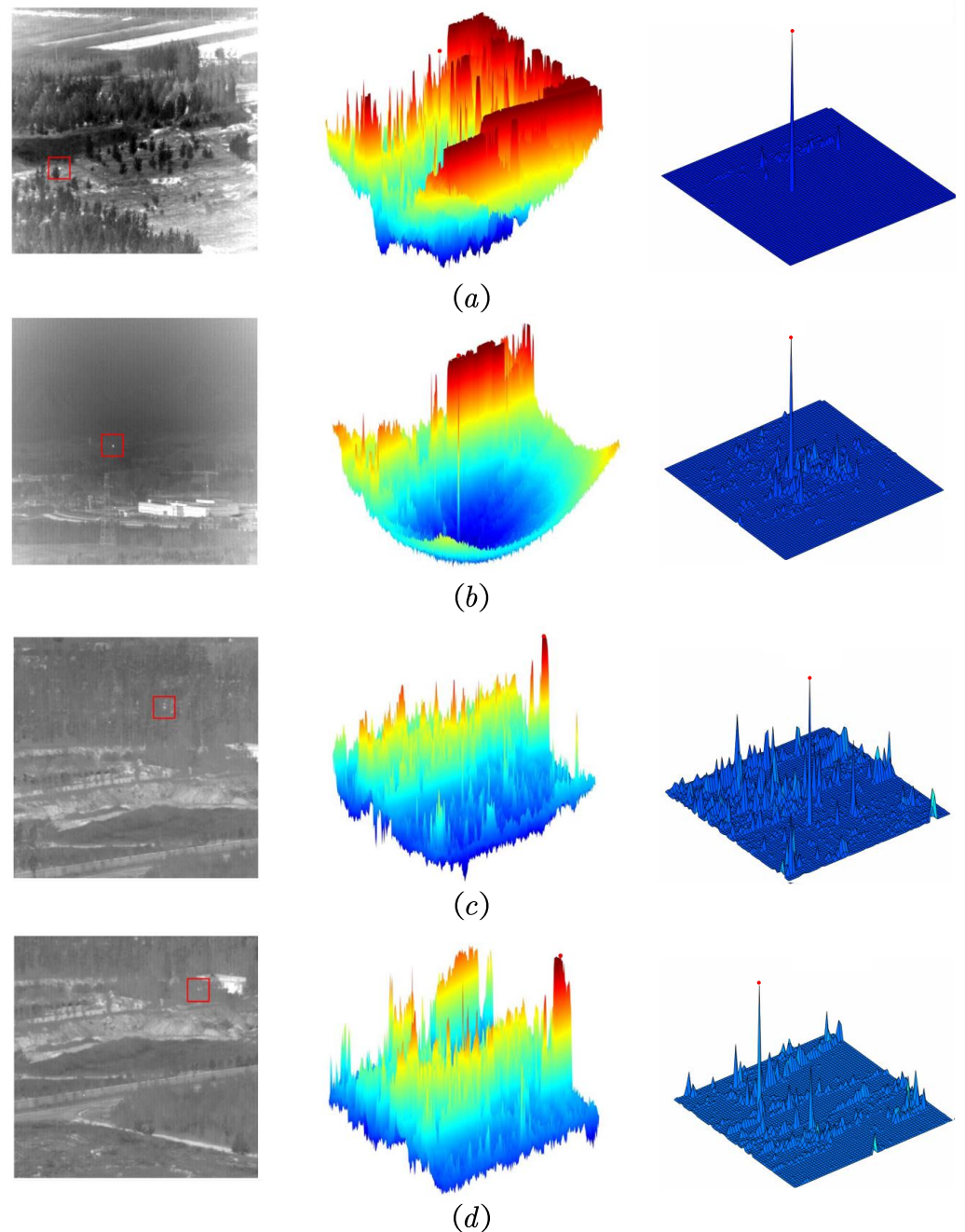


Figure 2. Typical target information in real-world scenarios (a–d). Note: From left to right are the original image, the 3D grid map of the whole image, and the grid map within the local area of the target.

The short-time Fourier transform can be used to analyze local signals with the help of a window. The Gabor transform is a type of short-time Fourier transform, and by introducing a Gaussian window, the spectral characteristics of a signal can be analyzed within a specific window range. The energy distribution of a weak target consists of its energy and morphological characteristics, and Gabor filtering in a fixed direction not only maintains the energy structure of the target in a specific direction but also enhances the specific frequency content in that direction while suppressing other frequency components. Here, the use of the Gabor transform for null domain filtering allows for the enhancement of point-like structures similar to the target while preserving significant edge and texture

features in the image, forming a basis for evaluating potentially similar elements. The standard 2D Gabor function is described as follows:

$$g_{\sigma, \omega_0}(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)} e^{j\omega_0(x+y)} \tag{19}$$

where σ_x and σ_y are the standard deviations of the Gaussian window function in the horizontal and vertical directions, respectively, and ω_0 is the spatial frequency of the plane wave.

Considering the size of the weak target, in order to enhance the adaptability to different sizes of targets as small as possible, the filter kernel parameters $\lambda = 2$ and $\theta = 0, 45^\circ, 90^\circ$, and 135° are chosen in this paper. The design of the filter kernel is shown in Figure 3.

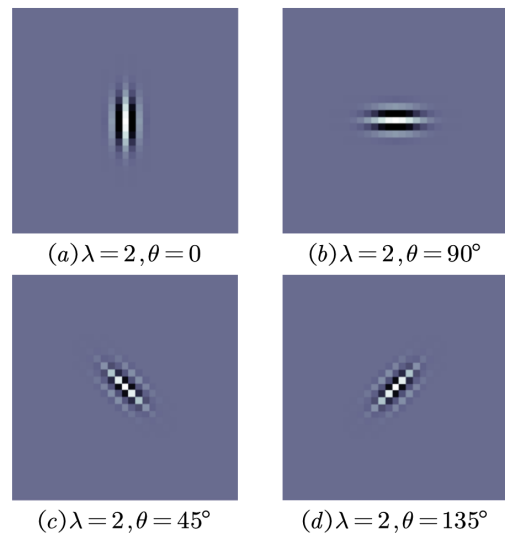


Figure 3. Four-directional Gabor filter kernel.

The original image is convolved using a filter kernel to obtain a resultant map of the features of the image in the specified direction:

$$I_{gabor}^i = I \times gabor(\theta_i) \tag{20}$$

$$I_{Gabor} = \sum_{i=1}^N \frac{I_{gabor}^i}{N} \tag{21}$$

where $N = 4$ is the filtering result of different angles and I_{Gabor} is the resultant map of image features fused with filtering features of different directions based on I_{Gabor} to calculate the evaluation index of the region similar to the target based on the potential similarity point elements.

Firstly, the concept of a High Pixel Rate (HPR) is defined here to estimate the potential target confusion that may be caused by the highlighted elements in the whole image by counting the number of pixels in the image that are higher than the target signal strength:

$$HPR = \frac{\#(I_{Gabor} > \mu_{T_{gabor}})}{\#I_{Gabor}} \tag{22}$$

where $\#(I_{Gabor} > \mu_{T_{gabor}})$ denotes the number of pixels in the optical data with an intensity greater than the target signal and $\#I_{Gabor}$ denotes the total number of pixels in the image.

Further, R, E, U, ASM, CON, COR, and IDM are computed from I_{Gabor} considering texture features retained after filtering.

In summary, for the image data of any scene, eight point-similarity-based parameters for evaluating target confusion elements are extracted from the point of view of confusion-prone elements and combined into the following confusion feature vectors:

$$CF = [HPR, R_{cf}, E_{cf}, U_{cf}, CON_{cf}, ASM_{cf}, IDM_{cf}, COR_{cf}] \tag{23}$$

Typical scenes were analyzed using a measure of the confounding factors contained in the background of the image, and the results are shown in Table 1.

Table 1. Analysis results of background confusion characteristics in typical scene images.

| | HPR | R_{cf} | E_{cf} | U_{cf} | CON_{cf} | ASM_{cf} | IDM_{cf} | COR_{cf} |
|-----|-----------------------|----------|----------|----------|------------|------------|------------|------------|
| (a) | 0.0019 | 0.4677 | 1.4847 | 0.4330 | 0.2737 | 0.5653 | 0.4060 | 0.8675 |
| (b) | 3.81×10^{-6} | 0.7733 | 2.7218 | 0.1737 | 0.2856 | 0.8575 | 0.1764 | 0.8683 |
| (c) | 0.2610 | 0.8991 | 3.0797 | 0.1653 | 0.5424 | 0.8800 | 0.1959 | 0.8127 |
| (d) | 0.0652 | 0.9836 | 4.7025 | 0.0487 | 4.8836 | 0.8385 | 0.0177 | 0.5227 |

2.4. Detectability Assessment

In order to assess the detectability of the target, this section will discuss the performance of three aspects of the target’s own feature assessment, the background’s own feature assessment, and the background element confusion assessment in a comprehensive manner and study the quantitative assessment techniques of the detectability of the weak target.

The contrast between the target and background directly affects the performance of the detection algorithm. In order to improve the tracker’s ability to recognize targets in the local background region, a number of methods have been proposed to assess the degree of fluctuation of the augmented response map and the confidence level of the detected targets. The Target Saliency Degree (TSD) is used to measure the difficulty of recognizing targets in local and global background regions and is defined as follows:

$$TSD = TSR \times SCR \times LSCR \tag{24}$$

$$TSD = \frac{1}{1 + TSD} \tag{25}$$

where SCR and LSCR are the results after normalization by the Norm function. The Norm function maps the values of x to between 0 and 1, and the distribution density of the values can be adjusted by the tuning parameter k [22]. The selection of k affects the smoothness and the trend of the values of x , and it can be applied to contrast enhancement and feature scaling in image processing:

$$Norm(x, k) = \frac{x}{x + k} \tag{26}$$

The value of k can be determined with reference to actual SCR and LSCR usage experience; k is set to one for SCR and three for LSCR.

In a single-frame image, the background is the main factor affecting the target-recognition accuracy. Complex edges and texture elements in the background can cause significant interference in target detection and recognition. The Background Complexity Degree (BCD) is defined as follows and combines global and local measures of background complexity:

$$BCD = \frac{RDR + R + E + CON}{U + ASM + IDM + COR} \tag{27}$$

where E, U, and CON are normalized using $Norm(\cdot, 1)$ for normalization.

In addition, beyond the characteristics of the target and the background itself, it is necessary to evaluate the target-like elements in the background scene. In this paper, we propose the target intra-frame confusion degree to reflect the evaluation of the intra-frame

confusion degree caused by the similarity between the target and the global background, and the Target Confusion Degree (TCD) is defined as follows:

$$\text{TCD} = \frac{\text{HPR} + \text{R}_{\text{cf}} + \text{E}_{\text{cf}} + \text{CON}_{\text{cf}}}{\text{U}_{\text{cf}} + \text{ASM}_{\text{cf}} + \text{IDM}_{\text{cf}} + \text{COR}_{\text{cf}}} \quad (28)$$

where E_{cf} , U_{cf} , and CON_{cf} are normalized using $\text{Norm}(\cdot, 1)$ for normalization.

On the basis of the above analysis, this paper integrates the three partial parameters—target characteristics, background characteristics, and background element confusion—to construct the feature matrix. The evaluation of the difficulty of the weak-target-detection task is obtained, and the Target Detectability (TD) is expressed as follows:

$$\text{TD} = \frac{\text{Norm}(\text{BCD}, 1) + \text{Norm}(\exp(\text{TCD}), 1)}{\text{TSD}} \quad (29)$$

By normalizing the above equation, the difficulty of the weak-target-detection task in a single-frame image can be calculated as follows:

$$\text{TD} = \frac{\tanh(8 \times (\text{Norm}(\text{TD}, 1) - 0.5))}{2 \times \tanh(4)} + 0.5 \quad (30)$$

3. Spatial Enhanced Detection Algorithm

The complex background seriously interferes with the signals of weak targets in the airspace. It is difficult to suppress the background effectively in the detection of low signal-to-clutter ratio targets in complex scenes, which tends to lead to false alarms, and too strong background suppression often leads to partial loss of the target signal. In this chapter, we examine methods to enhance the target signal while effectively suppressing the background. Based on the coupling characteristics of dim targets and complex environments, this chapter introduces an optical imaging mechanism into spatial-domain processing and combines the difference between target and background characteristics in the imaging process to build a single-frame processing method for signal enhancement and the background suppression of low signal-to-clutter ratio targets in complex scenes [23]. Inspired by the observation uncertainty caused by the spatial component changes in different elements in a space, this section proposes a Local Uncertainty Measurement based on the principle of component consistency to suppress complex scenes and detect small targets in complex backgrounds. On the basis of suppressing background clutter, the target signal is better retained and enhanced. The experimental results show that the proposed algorithm achieves a better detection performance in complex scenarios and has an excellent real-time performance.

3.1. Energy-Weighted Local Uncertainty Measure

In the process of the ultra-long distance detection of small moving targets, due to the interference of atmospheric conditions, noise, and other factors, as well as the limitation of detection equipment and precision, the target information will be polluted when transmitted through the signal acquisition channel. As the accuracy of the observation information decreases gradually, it becomes more difficult to accurately identify the target position from the detection data.

The object and its background in the image correspond to different component signals, and the occurrence of a fluctuating background, noise, and target signals in different regions brings uncertainty to the spatial characteristics of the observed data. Common methods use the local entropy operator to calculate the image complexity of the local region to measure the difference between the target and context, suppressing the cloud background. However, it is difficult to process complex background data without considering the relationship between different component signals and the relationship between the same component signals. Based on the uncertainty thought driven by the spatial variation of each component, this paper first analyzes the characteristic uncertainty of dim target detection and proposes

a Local Uncertainty Measure (LUM) based on the principle of component consistency, hoping to achieve dim target detection with a complex background.

The uncertainty measurement method evaluates local component consistency. This helps distinguish the target from the background. The method is divided into two stages: the local component uncertainty measurement and signal enhancement based on the energy-weighting function. In order to suppress the background clutter, the component consistency confidence is assigned by analyzing the component consistency of the local signal, and the uncertainty in the local region is measured by the local entropy operator. Then, the target energy information is introduced through the energy-weighting function to further enhance the target signal. Finally, an adaptive threshold segmentation algorithm is used to extract the target.

Figure 4 shows the detailed ELUM processing flow. Increasing the size of the computational unit results in a quadratic increase in resource consumption, while excessively small computational units struggle to capture critical image information. Consequently, the minimum unit of the computational module is defined as 3×3 . (1). Construction of the three-layer nested sliding window: First, the algorithm constructs a three-layer nested sliding window structure centered on the target-detection area. This structure expands outward from the central window to form a 5×5 multi-level window, comprising the central layer, the neighborhood layer, and the outermost environmental layer. The central layer directly corresponds to the target area, while the neighborhood and environmental layers are designed to capture the contextual information surrounding the target. (2). Assessment of local signal component consistency: The environmental layer is utilized to evaluate the consistency of signals within the neighborhood layer. The objective of this step is to identify signal patterns that significantly differ from background noise or other interfering factors by comparing the signal characteristics of the various components in the neighborhood layer. Through this evaluation, the method effectively distinguishes target signals from surrounding environmental noise, thereby enhancing the accuracy of subsequent processing. (3). Generation of the uncertainty distribution map: Based on the consistency assessment results obtained from the neighborhood layer, confidence levels regarding component consistency are assigned to each measurement point. These data are utilized to generate an uncertainty distribution map, which illustrates the strength of the signal consistency at each point, providing a foundation for subsequent filtering and target-confirmation processes. (4). Gaussian template matching filtering: A 3×3 Gaussian template matching filter is applied within the three-layer nested window. This step enhances the contrast between the target signal and the background by performing local weighting and smoothing on the signal, thereby rendering the signal characteristics of the target area more pronounced. The Gaussian template utilized is given by $[1/16, 2/16, 1/16; 2/16, 4/16, 2/16; 1/16, 2/16, 1/16]$, where each coefficient's denominator is a power of two. This configuration of the Gaussian template facilitates effective matched filtering while reducing computational overhead on the hardware. (5). Generation of the energy-weighted uncertainty map: The local energy-weighting factor is calculated based on the filtered data to produce the energy-weighted uncertainty map. This map reflects the likelihood of the presence of targets in various regions after signal enhancement, providing a quantitative basis for the final target detection and localization. (6). Threshold segmentation and target detection: Finally, threshold segmentation techniques are employed to process the energy-weighted uncertainty map, allowing for the separation of high-probability target areas. Through this stage of processing, the algorithm is capable of accurately identifying and localizing targets within the image.

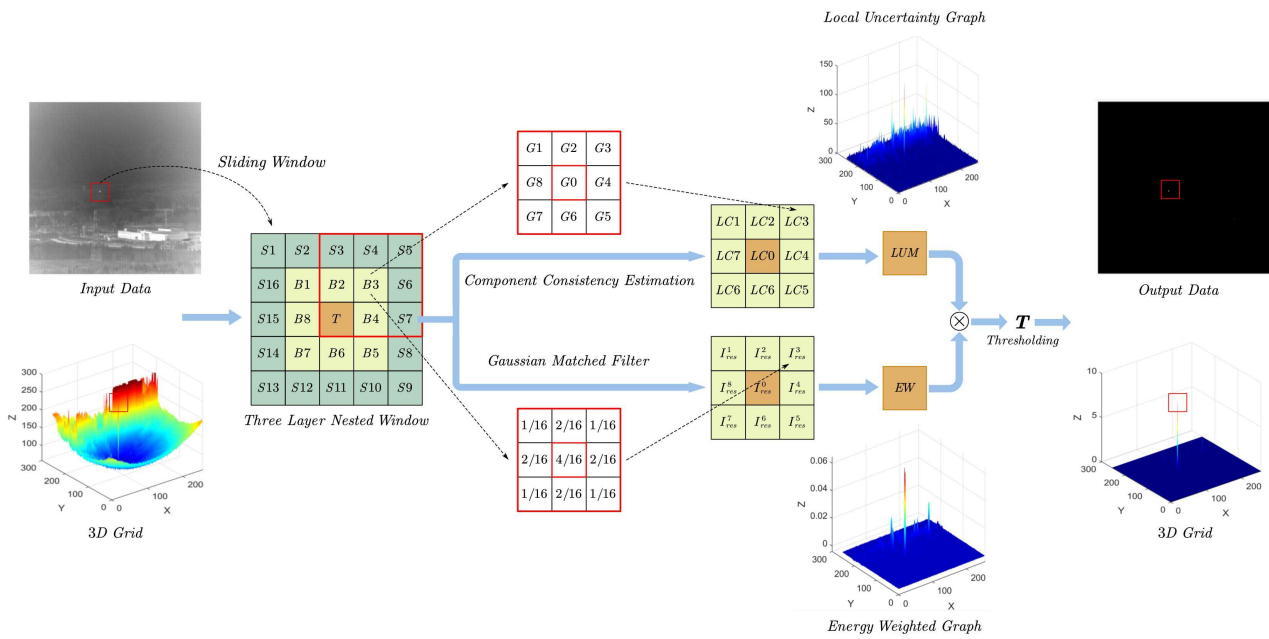


Figure 4. Schematic diagram of the ELUM method.

3.2. Local Uncertainty Measurement

In this paper, an uncertainty measurement method based on a local-component-consistency-evaluation assignment for target signal enhancement is proposed to help distinguish the target from the background. The uncertainty measurement is conducted in two steps: First, a confidence assignment function is constructed based on the results of the local component consistency evaluation. Then, the local component uncertainty of the image is measured according to the feature consistency of the local region of the image. Similar to methods inspired by the human visual system, the LUM can be calculated by sliding a window. The size of the sliding window is determined by the center window; the optimal size of the center window should be able to wrap the target signal, with the complete window structure being 5×5 in size. By adjusting the size of the center window, one can effectively deal with different sizes and shapes of the target.

In a complex background, there are different characteristics between the target and background and different types of backgrounds. Assume that a small window containing only similar-image background signals is relatively stable when the following occurs:

1. When the window only covers the background signal, the local gray value is stable and the consistency is high;
2. When the target signal is wrapped in the window, the local gray value is also relatively stable, but the energy in the window should be significantly higher than that in the background area;
3. When the window-wrapping target is at the interface with the background, the local gray value has an obvious gradient and low consistency.

In this paper, a local signal's gray consistency evaluation standard is defined, which is used to evaluate the signal component consistency between the central region and the surrounding eight neighborhood regions:

$$LC_{ij} = \frac{G_{ij}^0}{LB_{ij}} \times \frac{G_{ij}^0}{\max(G_{ij}^k)} \quad (31)$$

$$LB_{ij} = \frac{1}{K} \sum_{k=1}^K G_{ij}^k \quad (32)$$

where LC_{ij} represents the consistency evaluation result of the pixel at coordinate (i, j) and the signal components of the surrounding eight neighborhood regions. G_{ij} coordinates (i, j) as the center of the 3×3 area. G_{ij}^k corresponds to the first number k , a neighborhood grayscale average. The value of k is eight.

The component consistency evaluation LC_{ij} shows the following results:

- If the location of the G_{ij} region center is in high consistency with its eight neighboring regions, then $LC_{ij} \approx 1$;
- If the energy at the center of the G_{ij} region is low, then $LC_{ij} < 1$;
- If the energy in the center of the G_{ij} region is higher, $LC_{ij} > 1$.

In information theory, entropy is usually used to represent the uncertainty of random variables; that is, to calculate the expected amount of information, the following equation is used:

$$H(X) = - \sum_{x \in X} p(x) \times \log p(x) \tag{33}$$

where $p(x)$ is the occurrence probability of the event x . In evidence theory, the confidence assignment function is often used to assign $p(x)$. Based on this, a local component uncertainty method based on a variation entropy operator is proposed, which uses local consistency as the confidence assignment function.

When the window is crossed over the target region, the component uncertainty in the region is high because the local component consistency is low and the consistency confidence assignment function is low.

The uncertainty of the component consistency of the local spatial signal can be measured by the variation entropy operator. The proposed mutation entropy operator is expressed as follows:

$$U_{ij} = - \sum_{k=0}^K M_{ij}^k \times \log \left(1 - \left(M_{ij}^k \right)^2 \right) \tag{34}$$

where U_{ij} is the uncertainty of the position of the pixel (i, j) . M_{ij}^k is the component consistency confidence value assigned to each block in the window structure centered on (i, j) .

The confidence assignment function can be expressed as

$$M_{ij}^k = \frac{LC_{ij}^k}{\sum(LC_{ij})} \tag{35}$$

In this way, the confidence of each block in the window structure is processed so that the sum is one. The Local Uncertainty Measurement results are scaled to the same scale to ensure that the uncertainty results measured at different sliding window positions can be compared.

Figure 5 shows the distribution curve of the variation entropy function used in the case of binary information sources. Compared with the information entropy curve, the variation entropy used changes more sharply at both ends and more gently in the middle.

In a scenario based on local component consistency as a reliability distribution function, if the estimates of local component consistency in the sample region are close, the variation entropy function will give a lower entropy value, reflecting that the elements in the region have a lower uncertainty. Conversely, if the estimates of local component consistency differ significantly within the sampling region, the value of the variation entropy function is higher, indicating significant variation and high uncertainty in the region.

The entropy operator is not sensitive to the composition difference in a small range but shows a high sensitivity to the difference in a large range. Regions with significant differences in local component consistency usually appear at the target's location and around areas with extreme changes in background edges and corners. The high sensitivity of variation entropy means that it is able to ignore those small changes in composition and

focus more on features with significant differences. This property can effectively suppress background noise and improve the overall performance of the algorithm.

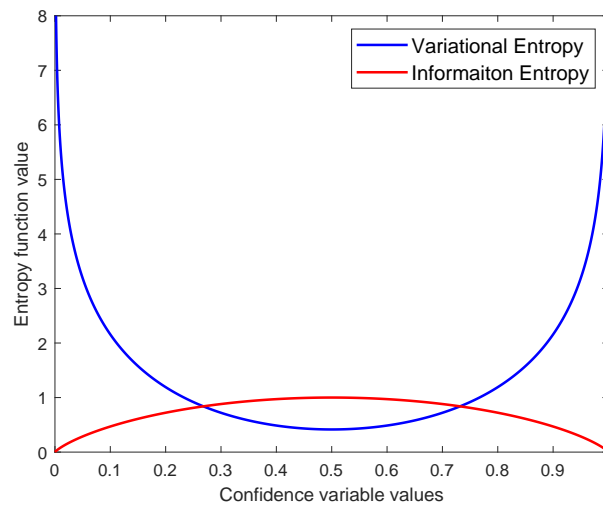


Figure 5. Entropy operator curves proposed under binary information sources.

Just as the information entropy follows the maximum entropy principle, the proposed mutation entropy operator follows the minimum entropy theorem, and the minimum value is obtained when $M_{ij}^k = 1/(K + 1)$:

$$Entropy_{min} = -\log\left(1 - \left(\frac{1}{K + 1}\right)^2\right) \tag{36}$$

Based on the above, to suppress the background to a greater extent, the measured component uncertainty is modified into the following form:

$$LUM_{ij} = U_{ij} - Entropy_{min} \tag{37}$$

When the signal component consistency is higher in the local region, the uncertainty is lower. The smaller the value of the variation entropy operator is, the closer the performance of the modified uncertainty operator is to 0.

3.3. Energy-Weighting Coefficient

In the process of the uncertainty measurement, the uncertainty of the local pixel component consistency is calculated. However, the calculation of the variation entropy does not strengthen the energy difference between different regions. Here, background estimation is performed by Gaussian filtering, and the residual signal is obtained.

The energy-weighting factor is designed based on local energy differences. Based on the uncertainty graph, the energy information contained in the target is strengthened and the target signal is improved. Considering that the weak target signal has a two-dimensional Gaussian shape, the background signal can be smoothed by Gaussian filtering:

$$I_{gaus}(i, j) = \sum_{x=-1}^1 \sum_{y=-1}^1 GK(x, y)I(i + x, j + y) \tag{38}$$

where I is the raw image data. I_{gaus} is the result of the Gaussian filtering of the original image. GK is the Gaussian template, here set as $GK = [1/16, 2/16, 1/16, 2/16, 4/16, 2/16, 1/16, 2/16, 1/16]$.

After the Gaussian template matching convolution, the residual difference between the original image and the Gaussian filtering image can be obtained:

$$I_{res}(i, j) = I(i, j) - I_{gaus}(i, j) \tag{39}$$

The target center energy decreases due to the accumulation of neighboring pixel energy, and the pixel energy increases due to the accumulation of neighboring pixel energy. Using the same sliding window as the component consistency evaluation process, the local energy difference in the residual image can be calculated as the signal energy weighting:

$$EW_{ij} = \max\{0, I_{res}(i, j) - I_b(i, j)\} \tag{40}$$

$I_b(i, j)$ is the residual mean of $I_{res}(i, j)$ in the surrounding eight neighborhood locations. After calculating the LUM and the weighting factor, the ELUM can be defined as

$$ELUM(i, j) = EW(i, j) \times LUM(i, j) \tag{41}$$

4. Hardware Accelerator Design

4.1. Overall Hardware Architecture

In the hardware implementation, we used both pipeline and parallel designs. These designs help to optimize the processing speed and resource utilization. The overall architecture is shown in Figure 6.

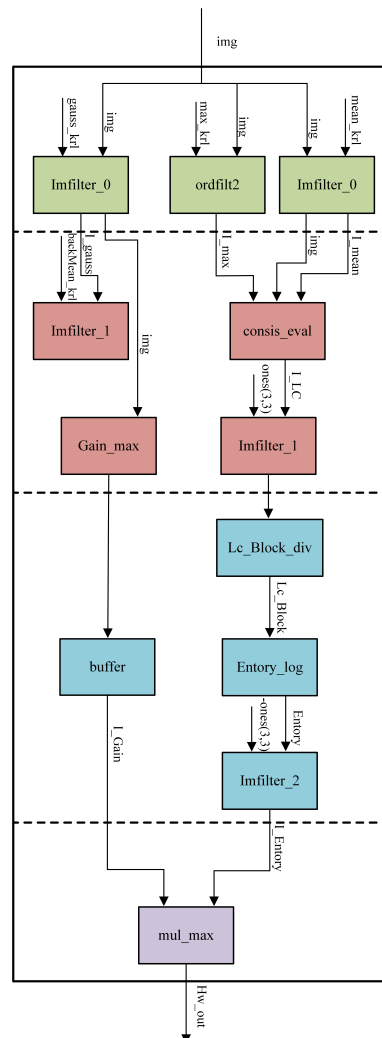


Figure 6. Overall hardware architecture.

By adopting a pipeline design, we split the task into four phases so that each phase can work independently. This allows the data to be processed in one phase while other phases can process different data in parallel. The requirements of the algorithm in this paper include background-suppression operations and target-enhancement operations. This naturally provides the conditions for hardware parallel implementation. Consistent with the algorithm, the overall hardware architecture of this paper is also divided into two independent computing paths. The primary task of the first computation path is to generate $LUM(i, j)$. Its core function is to realize background suppression to reduce the interference of background noise during target detection. At the same time, the second computation path synchronously generates $EW(i, j)$. Its main function is target enhancement based on background suppression. Through parallel design, the demand for memory access is effectively reduced, thus reducing the pressure on storage resources. In the design proposed in this paper, the computational complexity of the first line is higher than that of the second line. To solve this problem, we inserted a buffer after the calculation of the second line. By temporarily storing the calculation result, the second line can wait for the first line to complete the calculation of $LUM(i, j)$.

In the first stage of pipeline processing, the input image is scanned. This design realizes the filtering of different kernels through the reuse of the Imfilter module. In the first stage, we use Imfilter_0 to achieve filtering without extra precision. Among them, the mean filter is realized by using the mean filter kernel. At the same time, the Gaussian filter is realized by using the Gaussian filter kernel. In the first stage, we also use the Ordfilt2 module for filtering. The maximum filtering is implemented using the maximum filtering kernel.

In the second stage of the pipeline, combined with the data obtained in the first stage, we continue to scan the image. In the parallel algorithm background-suppression circuit, this design adopts the special Consis_eval module to achieve the component consistency measurement. This module includes the multiplication operation and the division operation, allowing us to calculate $LC(i, j)$. Then, we use Imfilter_1 to achieve double-precision filtering. We use the Imfilter_1 module to filter the previous $LC(i, j)$ through the cumulative kernel. In the parallel algorithm of the target-enhancement line, this design first runs the Imfilter_1 module. In the Imfilter_1 module, the results of the Gaussian filtering obtained in the first stage are filtered again; that is, the decentralized mean filtering. Then, the Gain_max module is designed to realize the energy-maximum calculation. The Gain_max module includes subtraction and comparison operations to calculate $EW(i, j)$.

In the third stage of the pipeline, we only execute the background-suppression algorithm. Firstly, the Lc_Block_div module is designed. The Lc_Block_div module is used to compute the confidence assignment function, including the division operation. We then import the result of the resulting confidence assignment function into the Entory_log module. The Entory_log module is used to calculate the mutation entropy operator, including multiplication and division and comparison and log operation, which needs to be called the Cordic IP core. In the second stage of the pipeline, the target-enhancement algorithm is implemented. Therefore, in the third stage, we only pass the result of the second stage through the buffer and wait for the end of the background-suppression algorithm to ensure the continuity of the entire pipeline processing.

In the fourth stage of the pipeline, this paper combines the background-suppression algorithm with the target-enhancement algorithm to obtain the final output result. In this paper, a special Mul_max module is designed to realize the combination of background suppression and target enhancement. The Mul_max module includes multiplication and comparison operations to calculate the final output result.

4.2. Submodule Design

4.2.1. Filter Module

The filter modules include two kinds, which realize Gaussian filtering and maximum value filtering, respectively.

The Imfilter_0, Imfilter_1, and Imfilter_2 modules can realize Gaussian filtering; these three modules are designed for different levels of precision, and their internal logic structure is basically the same, as shown in Figure 7.

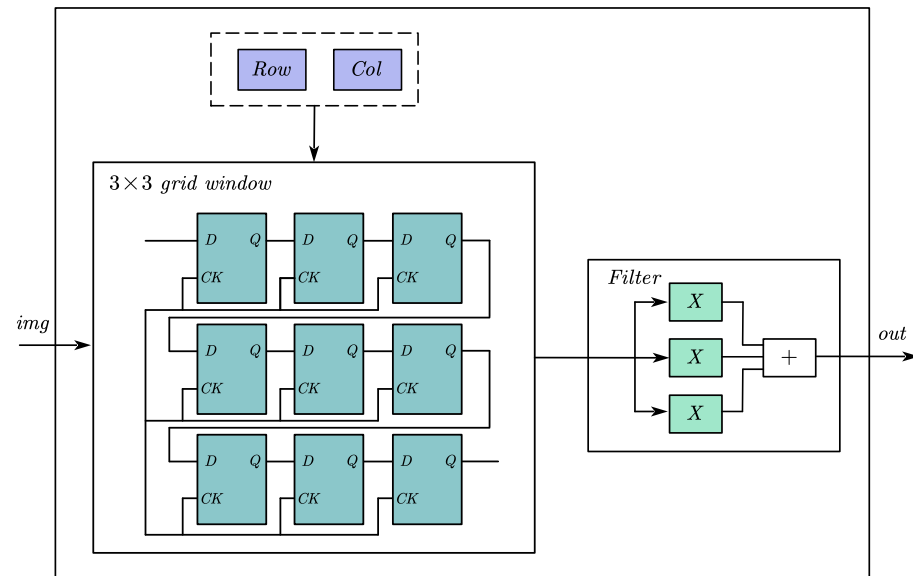


Figure 7. Filter module design.

The Imfilter module consists of three main parts: The first part is a row and column counter that counts the row and column pixels of the input image data and indicates the current element position for the calculation. The second part is used to generate a 3×3 window, consisting of two RAM-based shift registers with the depth of the RAM being the number of columns of the input image and the output of which is a 3×3 -sized filter window for calculating the value of the current element, containing nine windows of data. The third part implements the actual filtering function and contains mainly multiplication and addition logic, which first multiplies the input nine window data corresponding to the values of the convolution kernel and then adds the nine multiplication results. This part uses a pipelined design, where the result of the multiplication and the intermediate result of the addition are saved in registers to form a three-stage pipeline. The convolution kernel can be parameterized as required, and by configuring a specific convolution kernel, this module can also implement mean value filtering.

The Imfilter_0 module is designed for input image data with no extra precision, and the input data bit width is eight. The Imfilter_1 module and Imfilter_2 module are designed for input image data with $1 \times$ fixed and $2 \times$ fixed precision, respectively, and the input data contain extra fixed bits and $2 \times$ fixed decimal places.

The Ordfilt2 module can achieve maximum filtering; its internal structure of the row and column counter and window generator part is the same as the Imfilter module, and the difference is that the filtering function part of the multiply–add logic is replaced by two levels of comparison logic to achieve the maximum filtering function.

4.2.2. Component Consistency Measurement Module

The calculation of component consistency is implemented by the Consis_eval module, whose hardware structure is shown in Figure 8.

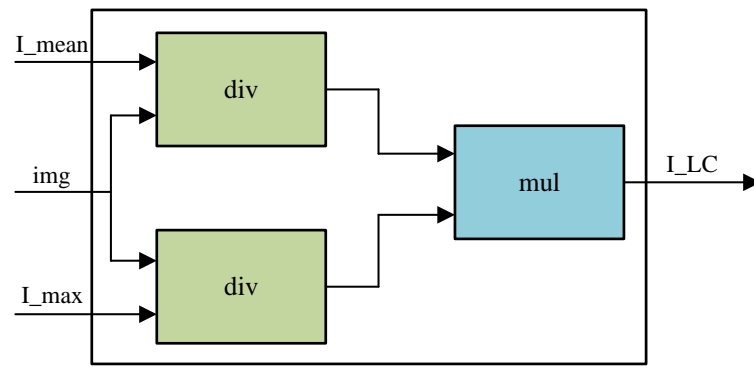


Figure 8. Composition consistency calculation module design.

Calculating the component consistency requires the use of the mean filtered and maximum filtered data of the image. Component consistency results are obtained by dividing the image data by the mean filter data and the maximum filter data, respectively, and then multiplying the two division results. The division and multiplication in the module are implemented using IP cores, and the two division operations are executed in parallel using pipeline division IP cores. The design of the module needs to pay attention to the alignment of the elements because the division and multiplication are conducted for the corresponding pixels.

4.2.3. Confidence Assignment Function Module

The confidence assignment function is implemented by the Lc_Block_div module, whose inputs include the delayed component-consistent data and the result of processing the component-consistent data by an Imfilter_1 module, where the values of the convolution kernel of the Imfilter_1 module are all set to one and the processed data are actually the result of the summation of the component-consistent data. The Lc_Block_div module mainly implements the division operation of component-consistent data with the summation result, using the pipelined division IP core.

4.2.4. Mutation Entropy Function Module

The mutation entropy function is implemented using the Entory_log module, whose hardware structure is shown in Figure 9.

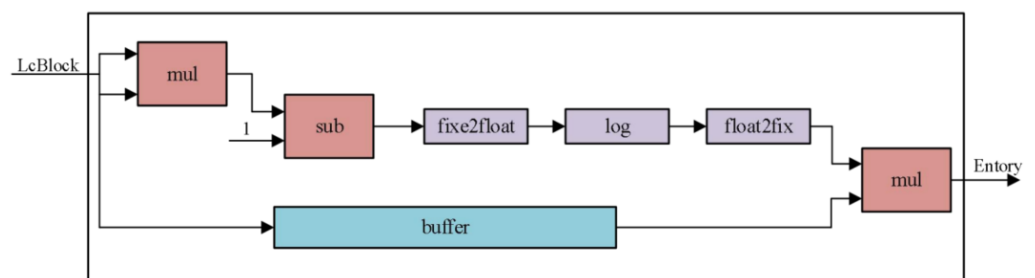


Figure 9. Mutation entropy function module design.

The modules in red are fixed-point operations, including two multiplication operations and one subtraction operation, while the modules in purple are floating-point operations, including the conversion between fixed-point and floating-point data and floating-point logarithmic operations. The multiplier is used to square the result of the confidence assignment function, subtract the squared result from one, use the logarithmic operation, and finally multiply it with the delayed result of the confidence assignment function. The delay of the data is managed using a shift register in the blue section of the buffer.

The complete mutation entropy function also requires a summation of the multiplication result, and an Imfilter_2 module is inserted after the Entory_log module to complete the summation operation, with its convolution kernels all configured with a value of one.

4.2.5. Remaining Operational Modules

The Gain_max module is mainly used for the calculation of the energy-weighting coefficient; the module implements the subtraction operations and comparison operations, subtracts the image data and the Gaussian convolution data to obtain the residual data, compares the residual data with 0, and uses the larger value as the final energy-weighting coefficient. Since the residual data are signed numbers, it is only necessary to judge whether the highest bit of the data is 1 to determine whether the data are greater than 0.

The Mul_max module implements the multiplication of the local uncertainty and the energy-weighting factor to obtain the final uncertainty result with energy weighting.

5. Experimental Results and Comparison

5.1. Experimental Results

The hardware architecture needs to be calculated in the fixed-point format. In contrast, the accuracy of the software simulation is double-precision floating point. Therefore, in the process of hardware implementation, due to the existence of the truncation error, the implementation result of the algorithm will be biased, which can be solved by choosing the appropriate digit width of the fixed-point number. To address this, the precision loss data under different fixed-point bits are analyzed first. Based on these considerations, the relative error calculation method is designed in this paper:

$$error = \frac{\sum |Result_{software}(i, j) - Result_{hardware}(i, j)|}{\sum |Result_{software}(i, j)|}, (Result_{software}(i, j) > 16) \quad (42)$$

Since the background of this paper pays more attention to the highlighted points (the higher gray value), the elements with a gray value of 16 and below are not considered here.

The results are shown in Figure 10. It can be seen that with the increase in fixed, the relative error shows a decreasing trend. After 11 digits, there is no obvious change, and the relative error stabilizes at 0.87%.

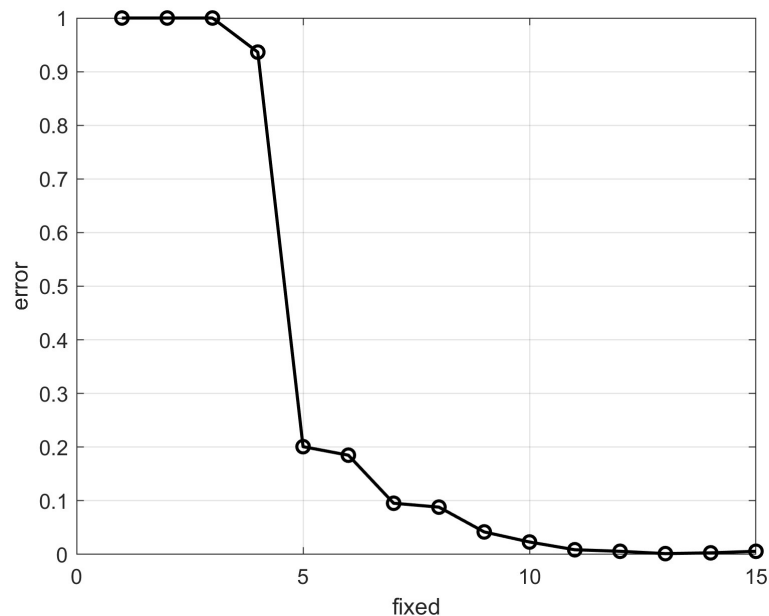


Figure 10. Relative error with the increase in fixed. Fixed refers to the number of decimal places in a fixed-point number.

To further validate the robustness and practical performance of our proposed method, this study employs a comprehensive and diverse infrared image dataset [24]. This dataset is specifically designed for the detection and tracking of small low-altitude flying targets and is generated through field recordings and post-processing, making it particularly suitable for the detection of fixed-wing unmanned aerial vehicles in complex backgrounds. The data collection encompasses a variety of environmental conditions, including sky and cluttered backgrounds, to simulate the diverse challenges encountered in real-world scenarios. The entire dataset consists of 22 image sequences, 30 trajectories, 16,177 frames, and 16,944 annotated target instances. Each target is associated with a clearly defined position label, and each image sequence corresponds to a detailed annotation file. This dataset not only provides a rich resource for the analysis of infrared target characteristics but also holds significant value for research in small target detection and precision guidance. By leveraging this extensive and substantial dataset, we are able to evaluate the performance of our algorithm under varying levels of noise, atmospheric conditions, and other real-world variables. These experimental results contribute to a more comprehensive demonstration of the effectiveness and adaptability of our design, and they also provide a robust benchmark for future research in related fields, as shown in Figures 11–13. Each test dataset generates four graphs. The images include the original drawing, the MATLAB R2021a double-calculation result, the simulation hardware calculation result (fixed = 11), and the actual hardware simulation result.

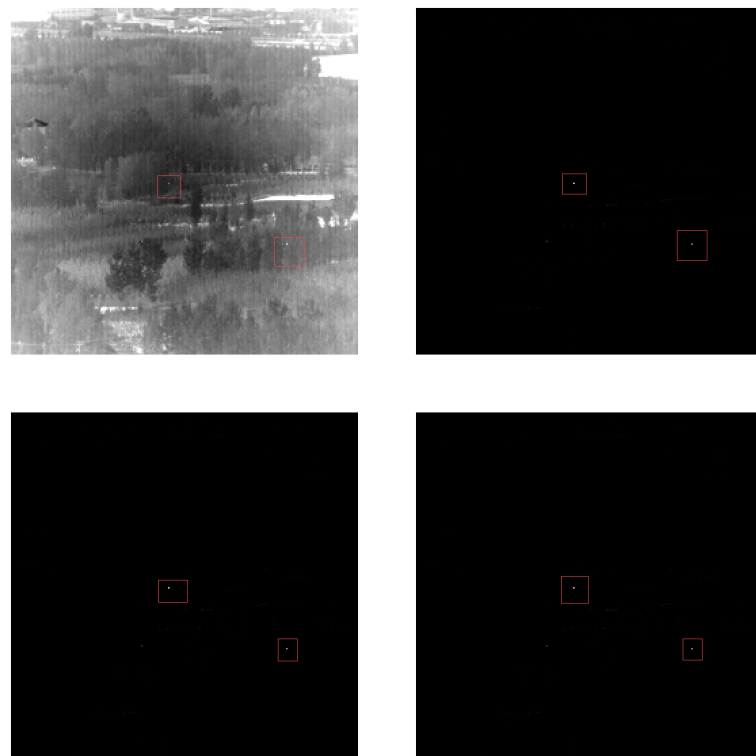


Figure 11. Test data example 1. The targets are highlighted with red boxes.

Through extensive testing on the dataset, the algorithm implemented in this paper demonstrates a false positive rate of only 7.69%, indicating a high level of recognition accuracy. This algorithm is capable of reliably identifying infrared small targets across various types of backgrounds.

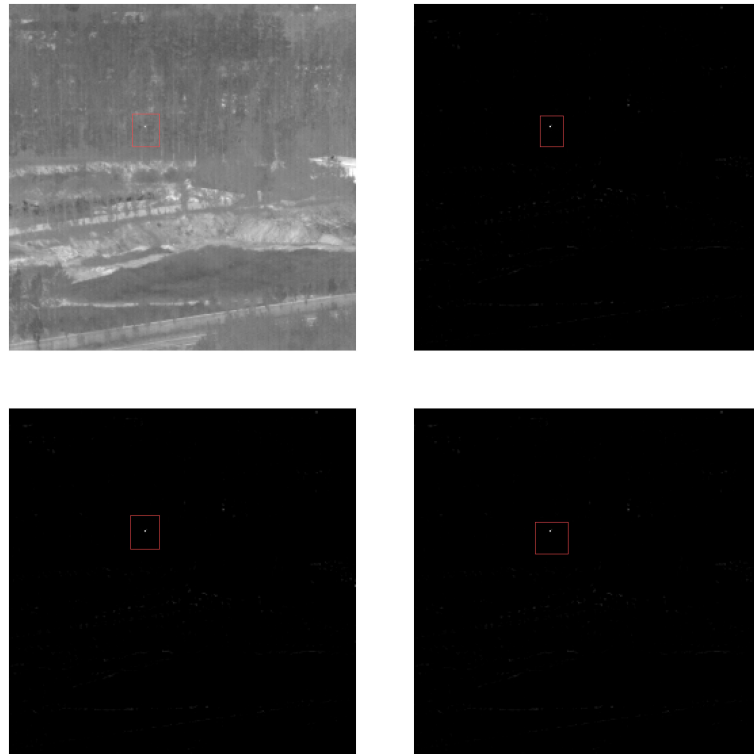


Figure 12. Test data example 2. The targets are highlighted with red boxes.

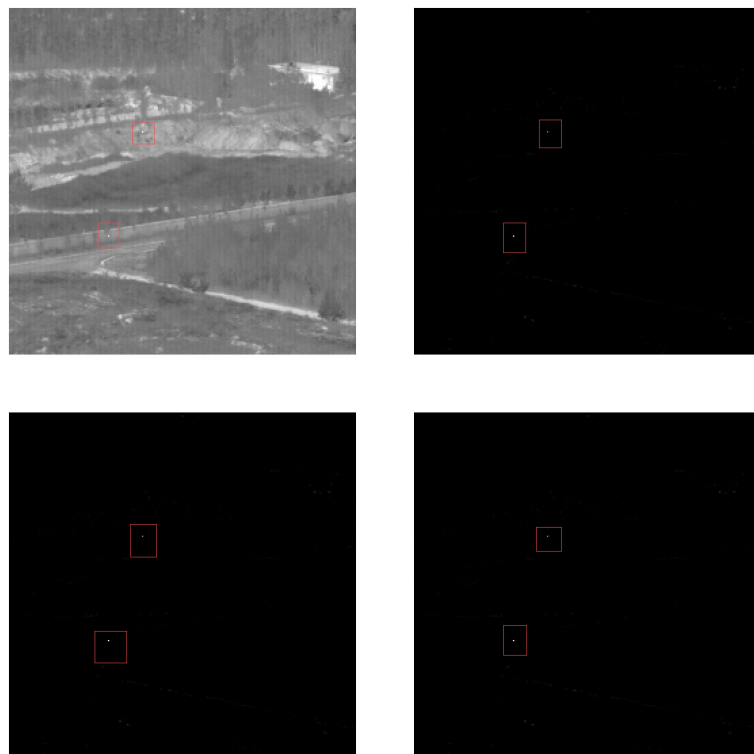


Figure 13. Test data example 3. The targets are highlighted with red boxes.

5.2. Comparison

This section presents a systematic analysis and discussion of the research findings. First, the evaluation criteria and specific parameters employed in this study are outlined. These metrics provide an objective basis for assessing the performance of the methodology.

Subsequently, based on these criteria, a detailed evaluation of the ELUM framework proposed in this paper is conducted, followed by a comprehensive comparison with existing methods to verify its superiority. The evaluation criteria and methods used in this study are described below.

LUT resources are defined as the configurable logic blocks within an FPGA that implement combinational logic functions. Each LUT can perform any logic operation by programming its inputs and outputs. LUT utilization reflects the flexibility and efficiency of hardware design in implementing complex logic functions and serves as a critical metric for assessing the design logic density and resource optimization. The LUT count is a key indicator of how efficiently a design uses the FPGA's logic capabilities and can directly influence the overall performance and scalability of the system.

Throughput is defined as the volume of data processed or transmitted by the hardware within a given time interval. It reflects the system's capacity to handle large quantities of data efficiently and serves as a critical performance metric. Throughput is a key indicator of the system's ability to manage data flows and its operational capacity. In this paper, throughput is represented by the number of infrared image pixels processed per unit time, as expressed by the following:

$$\text{Throughput} = \text{Resolution} \times \text{FPS} \quad (43)$$

We implemented the hardware design of the ELUM architecture on a Xilinx Kintex-7 series FPGA and conducted a comprehensive performance comparison with other existing designs. Table 2 presents the implementation results of our design after placement and routing, alongside the results of some recently proposed designs. The ELUM design is configurable to accommodate input images of varying resolutions by adjusting its parameters. Table 3 presents the processing speed of the proposed design at different input image resolutions. The design was proven to achieve a resolution of 3112×2048 pixels at a frame rate of 30 fps, ensuring that the system maintains a sufficient processing speed while preserving high image quality. Given that other designs handle images with varying resolutions, we use the previously mentioned throughput as a unified metric for comparison.

Table 2. Implementation results.

| Ref | Resolution | FPS | Throughput (Pixels/s) | LUT | DSP | Power (mW) |
|------|------------|-------|-----------------------|--------|-----|------------|
| [4] | 320 × 240 | 25 | 1,920,000 | 2587 | | |
| [5] | 500 × 500 | 104.2 | 26,050,000 | 8440 | 54 | 372 |
| [7] | 256 × 256 | 56 | 3,670,016 | 35,573 | | 48.7 |
| [1] | 320 × 240 | 204.1 | 15,674,880 | 19,401 | 4 | |
| [6] | 640 × 480 | 30.3 | 9,308,160 | 24,936 | | |
| [25] | 760 × 576 | 25 | 10,944,000 | 28,848 | | |
| our | 256 × 256 | 3051 | 199,950,336 | 5301 | 38 | 292 |

Note that our pixel depth is 8-bit.

Table 3. Processing speed for different input image resolutions.

| Resolution | FPS |
|-------------|------|
| 256 × 256 | 3051 |
| 1920 × 1080 | 96 |
| 2560 × 1440 | 54 |
| 3112 × 2048 | 30 |
| 4096 × 3112 | 15 |

Among all the FPGA implementations listed in the table, our design achieves the highest throughput, reaching up to 199,950,336 pixels per second, which represents a significant improvement over other hardware designs. From the perspective of logical resources, specifically LUTs, the algorithm proposed in this paper requires only 5301 LUTs

for implementation, markedly fewer than all other designs except [4]. Although the design in [4] utilizes a minimal amount of LUT resources, its throughput is limited to only 1,920,000 pixels per second. Additionally, we utilized Xilinx's power estimation tool to analyze the power consumption of the ELUM algorithm during operation. As shown in Table 2, the power consumption of the algorithm proposed in this paper is notably lower compared to other implementations with a similar performance.

6. Conclusions

In this paper, an uncertainty measurement method based on local component consistency is proposed for suppressing the complex background in infrared images and highlighting the target, thereby achieving the detection of small infrared targets. The local component consistency can reflect the feature difference information between the target and the background. Using the local component consistency as a confidence assignment function, the uncertainty of the image is measured using the mutation entropy function, which is highly sensitive to a wide range of compositional differences. The proposed method enhances the signal and suppresses background noise. The algorithm is implemented in hardware and deployed on the FPGA to improve the processing speed. After the experimental analysis, the algorithm proposed in this paper has strong detection capabilities and a strong processing performance.

The methods in this paper can be applied to areas where target recognition is required in complex environments. For example, in autonomous driving, tiny targets at longer distances can be recognized to provide an early warning, thus improving safety. In addition, the method in this paper focuses on single-frame images. By introducing time-domain information, multi-frame images can be processed to dynamically analyze tiny targets, thus realizing the recognition of small video targets.

Author Contributions: Conceptualization, X.W. and Y.L.; methodology, X.W., Z.Z., Y.J. and K.L.; software, X.W. and Y.L.; validation; formal analysis, Z.Z., Y.J. and K.L.; investigation, Y.L., X.Y., Z.H. and W.Z.; resources, Z.Z., Y.J. and K.L.; data curation, Z.Z., Y.J., K.L., X.Y., Z.H. and W.Z.; writing—original draft preparation, X.W., Z.Z., Y.J. and K.L.; writing—review and editing, X.W., Z.Z., Y.L. and K.L.; visualization, Y.L., X.Y., Z.H. and W.Z.; supervision, F.Z. and J.Z.; project administration, F.Z. and J.Z.; funding acquisition, F.Z. and J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China under grant 2023YFF07 19800.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data can be provided upon reasonable request to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| Symbol | Notation |
|--------|---|
| ELUM | Energy-weighted Local Uncertainty Measure |
| LUM | Local Uncertainty Measure |
| EWF | Energy-weighting function |
| PSF | Point Spread Function |
| RDR | Rank-Deficiency Rate |
| GLCM | Gray-Level Co-occurrence Matrix |

| | |
|-----|------------------------------|
| IDM | Inverse Difference Moment |
| ASM | Angular Second Moment |
| CON | Contrast |
| COR | Correlation |
| TSR | Target Significant Rate |
| HPR | High Pixel Rate |
| TSD | Target Saliency Degree |
| TCD | Target Confusion Degree |
| TD | Target Detectability |
| BCD | Background Complexity Degree |

References

- Li, C.X.; Lin, S.J.; Chang, K.L. FPGA Implementation of Infrared Images Small Targets Track-Before-Detect System. In Proceedings of the 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), Kyoto, Japan, 12–15 October 2021; pp. 532–533.
- Wu, Y.; Yan, H.; Wang, M. Detection algorithm of infrared dim small target based on FPGA. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 7–8 November 2020; pp. 7650–7655.
- Chen, Y.; Yu, Y.X.; Zhao, T. The method of infrared point target detection and tracking based on DSP + FPGA. *Appl. Mech. Mater.* **2014**, *457*, 1272–1277. [[CrossRef](#)]
- Bo, M.; Hui, Z.; Zheng, M.; Ang, L.; Wenyang, J.; Weijun, M. FPGA implementation of local contrast method for infrared small target detection. In Proceedings of the 2015 12th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Qingdao, China, 16–18 July 2015; Volume 3, pp. 1293–1297.
- Zhang, Y.; Gao, J.; Yang, X.; Yang, C. Hardware acceleration of infrared small target detection based on FPGA. In Proceedings of the 2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA), Chengdu, China, 16–19 December 2022; pp. 342–347.
- Roy, P.; Das, D.; Dash, P.; Chauhan, M. Hardware Software Co-Design for Real Time Detection of Small Target in IR Video. In Proceedings of the 2019 International Conference on Range Technology (ICORT), Balasore, India, 15–17 February 2019; pp. 1–5.
- Hu, K.; Sun, W.; Nie, Z.; Cheng, R.; Chen, S.; Kang, Y. Real-time infrared small target detection network and accelerator design. *Integration* **2022**, *87*, 241–252. [[CrossRef](#)]
- Kou, R.; Wang, C.; Peng, Z.; Zhao, Z.; Chen, Y.; Han, J.; Huang, F.; Yu, Y.; Fu, Q. Infrared small target segmentation networks: A survey. *Pattern Recognit.* **2023**, *143*, 109788. [[CrossRef](#)]
- Zhang, Z.; Ding, C.; Gao, Z.; Xie, C. Anlpt: Self-adaptive and non-local patch-tensor model for infrared small target detection. *Remote Sens.* **2023**, *15*, 1021. [[CrossRef](#)]
- Liu, Y. Review of infrared image complexity evaluation method. *Aviat. Weapon* **2014**, *3*, 51–54.
- Liu, R.; Wang, D.; Zhou, D.; Jia, P. Point target detection based on multiscale morphological filtering and an energy concentration criterion. *Appl. Opt.* **2017**, *56*, 6796–6805. [[CrossRef](#)] [[PubMed](#)]
- Samson, V.; Champagnat, F.; Giovannelli, J.F. Point target detection and subpixel position estimation in optical imagery. *Appl. Opt.* **2004**, *43*, 257–263. [[CrossRef](#)] [[PubMed](#)]
- Ma, T.; Wang, J.; Yang, Z.; Ku, Y.; Ren, X.; Zhang, C. Infrared small target energy distribution modeling for 2D subpixel motion and target energy compensation detection. *Opt. Eng.* **2022**, *61*, 013104. [[CrossRef](#)]
- Zou, Z.; Chen, K.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *Proc. IEEE* **2023**, *111*, 257–276. [[CrossRef](#)]
- Han, W.; Chen, J.; Wang, L.; Feng, R.; Li, F.; Wu, L.; Tian, T.; Yan, J. Methods for small, weak object detection in optical high-resolution remote sensing images: A survey of advances and challenges. *IEEE Geosci. Remote Sens. Mag.* **2021**, *9*, 8–34. [[CrossRef](#)]
- Pan, Z.; Liu, S.; Fu, W. A review of visual moving target tracking. *Multimed. Tools Appl.* **2017**, *76*, 16989–17018. [[CrossRef](#)]
- Zhao, M.; Li, W.; Li, L.; Hu, J.; Ma, P.; Tao, R. Single-frame infrared small-target detection: A survey. *IEEE Geosci. Remote Sens. Mag.* **2022**, *10*, 87–119. [[CrossRef](#)]
- Moradi, S.; Memarmoghadam, A.; Moallem, P.; Sabahi, M.F. Assessing the applicability of common performance metrics for real-world infrared small-target detection. *arXiv* **2023**, arXiv:2301.03796.
- Diao, W.H.; Mao, X.; Gui, V. Metrics for performance evaluation of preprocessing algorithms in infrared small target images. *Prog. Electromagn. Res.* **2011**, *115*, 35–53. [[CrossRef](#)]
- Soni, T.; Zeidler, J.R.; Ku, W.H. Performance evaluation of 2-D adaptive prediction filters for detection of small objects in image data. *IEEE Trans. Image Process.* **1993**, *2*, 327–340. [[CrossRef](#)] [[PubMed](#)]
- Shi, B.; Guo, J.; Wang, C.; Su, Y.; Di, Y.; AbouOmar, M.S. Research on the visual image-based complexity perception method of autonomous navigation scenes for unmanned surface vehicles. *Sci. Rep.* **2022**, *12*, 10370. [[CrossRef](#)] [[PubMed](#)]
- Singh, N.; Singh, P. Exploring the effect of normalization on medical data classification. In Proceedings of the 2021 International Conference on Artificial Intelligence and Machine Vision (AIMV), Gandhinagar, India, 24–26 September 2021; pp. 1–5.
- Qu, X.; Chen, H.; Peng, G. Novel detection method for infrared small targets using weighted information entropy. *J. Syst. Eng. Electron.* **2012**, *23*, 838–842. [[CrossRef](#)]

24. Hui, B.; Song, Z.; Fan, H.; Zhong, P.; Hu, W.; Zhang, X.; Lin, J.; Su, H.; Jin, W.; Zhang, Y.; et al. *A Dataset for Infrared Image Dim-Small Aircraft Target Detection and Tracking under Ground/Air Background*; Science Data Bank: Beijing, China, 2019. [[CrossRef](#)]
25. Rong, S.H.; Zhou, H.X.; Qin, H.L.; Wang, B.J.; Qian, K. A real-time tracking system of infrared dim and small target based on FPGA and DSP. In *Proceedings of the International Symposium on Optoelectronic Technology and Application 2014: Infrared Technology and Applications, Beijing, China, 13–15 May 2014*; SPIE: Bellingham, WA, USA, 2014; Volume 9300, pp. 304–309.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.