



Article Moving Object Detection in Freely Moving Camera via Global Motion Compensation and Local Spatial Information Fusion

Zhongyu Chen ^{1,2,3}, Rong Zhao ^{1,2,3}, Xindong Guo ^{1,2,3}, Jianbin Xie ¹ and Xie Han ^{1,2,3,*}

- ¹ School of Computer Science and Technology, North University of China, Taiyuan 030051, China; b20220703@st.nuc.edu.cn (Z.C.); b1907073@st.nuc.edu.cn (R.Z.); gxd@sxau.edu.cn (X.G.); kedachang@126.com (J.X.)
- ² Shanxi Key Laboratory of Machine Vision and Virtual Reality, Taiyuan 030051, China
- ³ Shanxi Province's Vision Information Processing and Intelligent Robot Engineering Research Center, Taiyuan 030051, China
- * Correspondence: hanxie@nuc.edu.cn

Abstract: Motion object detection (MOD) with freely moving cameras is a challenging task in computer vision. To extract moving objects, most studies have focused on the difference in motion features between foreground and background, which works well for dynamic scenes with relatively regular movements and variations. However, abrupt illumination changes and occlusions often occur in real-world scenes, and the camera may also pan, tilt, rotate, and jitter, etc., resulting in local irregular variations and global discontinuities in motion features. Such complex and changing scenes bring great difficulty in detecting moving objects. To solve this problem, this paper proposes a new MOD method that effectively leverages local and global visual information for foreground/background segmentation. Specifically, on the global side, to support a wider range of camera motion, the relative inter-frame transformations are optimized to absolute transformations referenced to intermediate frames in a global form after enriching the inter-frame matching pairs. The global transformation is fine-tuned using the spatial transformer network (STN). On the local side, to address the problem of dynamic background scenes, foreground object detection is optimized by utilizing the pixel differences between the current frame and the local background model, as well as the consistency of local spatial variations. Then, the spatial information is combined using optical flow segmentation methods, enhancing the precision of the object information. The experimental results show that our method achieves a detection accuracy improvement of over 1.5% compared with the state-of-the-art methods on the datasets of CDNET2014, FBMS-59, and CBD. It demonstrates significant effectiveness in challenging scenarios such as shadows, abrupt changes in illumination, camera jitter, occlusion, and moving backgrounds.

Keywords: motion object detection; inter-frame transformations; local spatial; optical flow

1. Introduction

With the widespread use of surveillance cameras in recent years, the number of recorded videos has been increasing dramatically. Surveillance videos typically contain complex scene information, and it is critical to reasonably analyze and process the interesting part. In most cases, MOD serves as the foundation for advanced vision tasks, such as video surveillance [1], object tracking [2,3], and human–computer interaction surveillance [4]. It is an important component in intelligent security surveillance video because it provides basic technical support for massive video analysis and processing.

MOD techniques vary depending on whether the camera is moving or not. Numerous methods have been researched for the detection of moving objects when using fixed cameras, including the background subtraction (BGS) methods [5–9], the frame difference methods [10,11], the optical flow methods [12], and deep learning [13–18]. They primarily detect motion objects by acquiring inter-frame information or constructing background



Citation: Chen, Z.; Zhao, R.; Guo, X.; Xie, J.; Han, X. Moving Object Detection in Freely Moving Camera via Global Motion Compensation and Local Spatial Information Fusion. *Sensors* 2024, 24, 2859. https:// doi.org/10.3390/s24092859

Received: 23 February 2024 Revised: 24 April 2024 Accepted: 28 April 2024 Published: 30 April 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/). models. Accurate detection results can be obtained when the background is completely static or the scene is relatively simple. However, in the case of a freely moving camera, the background is typically dynamic and extremely complex. The pixel intensity at a specific location in the background may undergo irregular variations, which may invalidate algorithms designed for fixed cameras, posing a new challenge to MOD.

Currently, there are various MOD methods used to deal with complicated background sceneries, such as camera jitter [19], small object recognition [20,21], abrupt changes in illumination [22], shadow removal [23], and video with noise or a damaged image frame [24]. According to whether the background is modeled or not, it can be broadly divided into two categories, i.e., model-based and motion-based algorithms. The first category combines motion compensation methods [25] to extend MOD under fixed cameras, establishing local or global background models. Model-based algorithms [26–28] align several adjacent image frames using affine transformations and then extract foreground objects using background modeling or background subtraction. Since the motion between adjacent image frames is localized, it is challenging to detect slowly moving or temporarily stationary objects. Moreover, this approach fails when the video contains huge amounts of global motion (e.g., panning, tilting, rotating, and jittering). It is due to the fact that the motion vector between consecutive frames does not adequately depict global motion and does not give sufficient information to rebuild the current frame. The second category analyzes foreground objects by estimating the consistency of pixel point or image block motion using the optical flow method [29]. In this case, the MOD results are hardly affected by background variations as the motion-based algorithms [4,30,31] do not require the construction of an explicit background model. These methods perform well when there are relatively large objects and the background is moving in a globally consistent manner on the image. The detection performance, however, is heavily dependent on the accuracy of the motion vector between the foreground object and the background. Moreover, in the presence of local spatial noise and illumination interference, such as changes in the position of light sources, shadows, etc., the algorithm may extract inaccurate motion information, which can affect the subsequent calculation and analysis.

In this paper, we present a hybrid approach that combines the advantages of motion compensation and optical flow for motion object detection. The primary focus is on enhancing the accuracy of global motion compensation and mitigating the interference of complex backgrounds on local foreground areas. Therefore, we propose a MOD algorithm based on global motion compensation and local spatial information fusion on a freely moving camera. On the premise of eliminating inter-frame motion transformation with large errors, we convert the relative projection transformation problem into a global-based absolute projection transformation problem and perform global optimization to ensure accurate transformation. Furthermore, in the local spatial, the optical flow segmentation approach is combined to enhance the foreground object contour and reliably detect the foreground object. The contributions are summarized as follows:

- (1) A novel motion compensation framework for global optimization is constructed. This framework aims to estimate inter-frame transformations effectively and improve global matching accuracy by optimizing the residual matching and drawing on the use of back-end optimization methods [32] and STN [33].
- (2) A novel method for local spatial information fusion is proposed, which utilizes several types of local spatial boundary information, to effectively handle challenges such as foreground false alarms, hollow foreground objects, and unclear contours.
- (3) An end-to-end MOD method is proposed through the collaborative work of the two modules mentioned above. The proposed approach is validated on three datasets, and a comparison is made with existing methods. The results show state-of-the-art performance and confirm the efficacy of the proposed method. Furthermore, the method covers a wide variety of camera motions, which increases its practical utility.

2. Related Works

The process of extracting and identifying changing areas from the background image of a video image sequence is known as MOD. Object detection methods for dynamic scenes are critical as cameras are increasingly used on mobile cameras. Existing methods for detecting motion objects with moving cameras are classified into three types: model-based algorithms, motion-based algorithms, and hybrid algorithms.

2.1. Model-Based Algorithms

The model-based algorithm focuses on finding invariant areas to construct a statistical model of the scene by means of motion compensation. Liu et al. [26] proposed an unsupervised learning framework based on the foreground model that computes motion using homography transformations across consecutive frames to find a limited number of keyframes. The motion cues obtained from the Markov Random Field (MRF) are combined with the appearance of the Gaussian Mixture Model (GMM) to achieve motion object segmentation. Similarly, Zamalieva et al. [27] proposed a method to adaptively change the geometric transformation matrix to estimate the background motion caused by camera motion. They utilized Bayesian learning to model the appearance of the background and foreground. Recently, some studies [34] proposed a Poisson-fusion-based background orientation area reconstruction approach to obtain foreground objects by computing the difference between the original orientation area and the reconstructed orientation area. In other methods, low-rank matrix factorization is employed for subspace learning. Eltantawy et al. [20] proposed a new principal component pursuit (PCP) [35] method for modeling moving objects as moving sparse objects based on multiple local subspaces. Others [36] used the developed candidate background model to iteratively update the background model, and then performed background subtraction based on the consistency of local changes. Some methods also focus on establishing background models in the subspace. Chelly et al. [37] used Robust Principal Component Analysis (RPCA) [38] to generate a local background map for background reduction. However, this method is more sensitive to camera rotation and image perspective translation, not to mention that it is not an end-to-end learning approach, and its applicability is limited. By embedding affine transformation operators in online subspace learning, He et al. [19] obtained a more extensive subspace alignment. However, the above algorithm cannot handle non-rigid dynamic backgrounds, and its performance may suffer if complicated backgrounds cannot be approximated as planes.

2.2. Motion-Based Algorithms

Motion-based algorithms use the difference in motion patterns between foreground and background objects. It is classified into two types: point-based and layer-based. Pointbased methods detect and track sparse feature points before executing segmentation on the same type of object. Some studies [39] deconstruct the motion trajectory matrix and classify foreground objects at the pixel level using trajectory information. Some [40,41] extract foreground areas from the video by clustering the obtained long feature trajectories. Manda et al. [42] combined optical flow information between neighboring frames to extract features of the object in both spatial and temporal dimensions. Others utilized [43] lengthy trajectory features and a Bayesian filtering framework to estimate motion and appearance models. Although these methods are robust in handling large scenes of camera motion, they only create sparse point segmentation that must be post-processed to produce dense segmentation.

The layer-based method computes dense optical flows and then clusters them based on motion consistency. Shen et al. [30] used dense optical flow to calculate the trajectory of each pixel point and then used a two-stage bottom–up clustering approach to identify the final motion object. Some researchers utilize initial motion boundary information generated by dense optical flow fields to detect moving objects. Sugimura et al. [31] employed the initial foreground and background label boundary information in conjunction with the area segmentation method to recognize foreground objects. The initial labels are computed using the motion boundaries of two different flow field sizes and directions. Additionally, Zhang et al. [44] used the weighted local difference in orientation of the optical flow to detect the moving object's rough contours, and it is used to guide the selection of prospect objects. In contrast, Bideau et al. [4] combined the optical flow angle and size to optimize the difference between object motions. Due to quick camera movements and video noise, such methods may create some mistakes in measurements. Singh et al. [45] used the similarity of image and optical flow feature information to generate edge weights, and then self-supervised trained the network using a graph cut mask. The optical-flow-based approach is further constrained by the initialization process, which necessitates the development of a priori information such as contour lines and object numbers. Recently, deep learning methods [46–52] have been used to extract optical flow features to detect moving objects.

2.3. Hybrid Algorithms

With the intention of improving detection, hybrid approaches integrate information such as motion, color, and appearance. In this regard, Elqursh et al. [43] relied on long trajectory information in low-dimensional space, considering non-spatial trajectories as foreground objects. They utilized long trajectories, motion, and appearance models combined within a Bayesian filtering framework to obtain the final foreground objects. Delibacsouglu et al. [53] proposed a moving object detection method based on background modeling and subtraction, which represents the background as a model with features such as color and texture. Similar to the above method, Cui et al. [54] used Markov stochastic models for trajectory, appearance, and spatiotemporal cues for the detection of moving object trajectories and background trajectories generated by optical flow. Reference [55] created foreground probability estimates by fusing the motion and appearance modules and then used graph cuts to generate the final segmentation masks. Some researchers also proposed to maximize the relationship between foreground and background clues as much as possible. Makino et al. [56] combined the fraction obtained from the difference between the background model and the current frame with the motion fraction obtained from the local optical flow calculation to more accurately detect moving objects. This method, however, is ineffective when the moving object is too large. Particularly, some studies [57] modeled multiple foreground objects appearing in the scene at different levels, treating it as a semantic segmentation problem. They estimated the motion and appearance models of these objects and employed a Bayesian filtering framework to infer a probability map. In another research work, Zhao et al. [58] used the GMM to obtain the confidence of the foreground cue for each pixel point after motion compensation. The confidence in the foreground cue combines with the confidence in the background pixel points obtained by feature point matching. Finally, they integrated the two confidences to achieve an accurate segmentation result.

3. Method

In this section, we propose a MOD method for a moving camera. This method aims to adapt to complex environments for detecting foreground objects and enhance the overall detection performance. Traditional foreground object detection methods are often susceptible to environmental factors, such as camera motion and illumination transformations. These factors can lead to inaccurate detection results. To solve this problem, we use a novel motion compensation global optimization framework to reduce the effect of camera motion on detection results. Additionally, we address the influence of local spatial lighting variations by constructing a local spatial background model and employing a spatial information fusion strategy. An overview of the algorithm is shown in Figure 1.



Figure 1. Overview of the proposed method: (1) Input: input the video sequences. (2) The transformation between adjacent frames is optimized globally. Next, to acquire the global background model, we apply the global transform to the local background model obtained from local spatial detection. (3) Combine the coarse foreground image obtained from the global background model with the optical flow information. Then, select the corresponding object area using an adaptive method, and fuse the spatial information from both sources to obtain the final foreground object.

3.1. Inter-Frame Registration

The projection transformation establishes the correspondence between the points (x, y) and (\hat{x}, \hat{y}) in two different image frames, which is shown below:

τ

$$\hat{V} = H^T V \tag{1}$$

where $\hat{V} = [\hat{x}, \hat{y}, 1]^T$ and $V = [x, y, 1]^T$ represent the vector form of the point, and τ is a random scaling constant. The projection transformation matrix is denoted as $H \in \mathbb{R}^{3\times 3}$. Given $\eta > 3$ and $\{V_i \rightarrow \hat{V}_i\}_{i=1}^{\eta}$, the least squares method is used to estimate the optimal transformation to H, which is shown below:

$$\min_{h \in \mathcal{H}} \|\psi_h\|^2 \quad s.t. \quad h_9 = 1 \tag{2}$$

where *h* is the vectorize representation of *H*, with the constraint that the last element h_9 is equal to 1. $\psi^T = [\psi_1^T, ..., \psi_\eta^T]$. The solution of Equation (2) is the smallest right singular vector of ψ , scaled so that the last element is 1, and the following equation holds:

$$\psi_i = \begin{bmatrix} 0 & V_i^T & -\hat{y}_i V_i^T \\ V_i^T & 0 & -\hat{x}_i V_i^T \end{bmatrix} \in R^{2 \times 9}$$
(3)

During the registration process, SURF [59] is used to compute features in each frame of the image, and then RANSAC [60] is employed to eliminate point matches with significant errors, aiming to estimate *H* with minimal loss.

In order to achieve precise inter-frame matching pairs while generating more overlapping areas, it is important to ensure that the local RPCA segments are more comprehensive foreground objects. Inter-frame matching extends to each image with its next *n* frames, as shown in Figure 2. The projection transformation error ϕ is calculated for each frame with its next *n* frames, and the matching pair N_{max} with the largest error value will be rejected, as shown in Equation (4):

$$N_{\max} = \arg\max_{n} \frac{M_{I \to I_n}}{C_{I \to I_n}} \tag{4}$$

where $\phi = M_{I \to I_n} / C_{I \to I_n}$ and $\{I, I + 1, ..., I + n\}$ are the current frame and the following *n*th frame images. I_n represents the image obtained by the projective transformation of the current frame *I* with respect to the frame I + n. $C_{I \to I_n}$ denotes the number of overlapping pixels of I_n and I + n. M_{I-I_n} denotes the number of pixels in the overlapping pixels that satisfy the threshold of the difference between the two pixels greater than ρ .



Figure 2. The image registration process.

3.2. Global Projection Optimization

The motion compensation for inter-frame motion using the transformation matrix H generated by the preceding method frequently has substantial inaccuracies. To improve the transformation accuracy even further, the registration images are transferred to the coordinate system using the intermediate frame as a reference, and the registration data are processed together. It will solve the synchronization problem on the special Euclidean group by optimizing the mentioned inter-frame relative transformation matrix with a back-end optimization algorithm. Given a set of noisy paired transformation matrices, the values of an unknown pose (position and orientation in Euclidean space) are estimated through optimization, as shown in Equation (5). By utilizing such optimization method, the noisy inter-frame relative transformation matrix \tilde{T}_{ij} is globally optimized to obtain the absolute transformation matrix T_i with the reference frame being the intermediate frame.

$$(T_j)_j^N = \arg\min\sum_{(i,j)\in\varepsilon} a_{ij} \|R_i - R_j \tilde{R}_{ij}\|_F^2 + b_{ij} \|t_i - t_j - R_j \tilde{t}_{ij}\|_{l^2}^2$$
(5)

Here, the inter-frame information is denoted as a non-linear weighted undirected graph $G = (v, \varepsilon, \omega)$, and v denotes the finite set of optimized frames, $(i, j) \in \varepsilon$ denotes the matching relationship between the *i*-th frame and the *j*-th frame, and the weights *w* store information about the inter-frame relative transformation matrix \tilde{T}_{ij} . *N* is the *n*-th frame of the image. $\|*\|_F^2$ and $\|*\|_{l_2}^2$ are Frobenius norm and L2 norm, respectively. a_{ij} and b_{ij} are optimized weights, while \tilde{t}_{ij} and \tilde{R}_{ij} (rotation matrix) belong to the planar rotation group SO(2). $\tilde{T}_{ij} = \begin{bmatrix} \tilde{R}_{ij} & \tilde{t}_{ij} \\ 0_{1\times 2} & 1 \end{bmatrix}$ and $T_j = \begin{bmatrix} R_j & t_j \\ 0_{1\times 2} & 1 \end{bmatrix}$.

The STN is utilized to optimize the residual transform and minimize the loss in Equation (6). It has the ability to automatically perform spatial mapping transformations and learn translations, scaling, rotations, and more general transformations. Its input consists of a series of transformation parameters T_j and images, and its output is the transformed image. For more information about the benefits of using STN to learn projection transformations, please refer to [61].

$$\Gamma = \arg\min\sum_{j=1}^{N}\sum_{i=1}^{D}\frac{\rho_{i}^{j}f\left(x_{i}^{j}-u_{i},\varphi\right)}{\sum_{a\in D}\rho_{a}^{j}} + \lambda\sum_{j=1}^{N}d(T_{j},\theta)$$
(6)

$$u_{i} = \frac{\sum_{j=1}^{N} \rho_{j}^{j} x_{i}^{j}}{\sum_{j=1}^{N} \rho_{i}^{j}}$$
(7)

Here, *N* is the *n*-th frame image, and *D* denotes the panorama resulting from absolutely transforming all image frames after back-end optimization. The value of ρ_i^j indicates the positional relationship of an image within a panoramic image. It takes a value of 1 when there is overlap between the image transformed by the absolute transformation of the *j*-th frame and the panoramic image, and 0 otherwise. $f(., \varphi)$ is a differentiable robust error function. x_i^j is the pixel value at position *i* of the *j*-th frame of the transformed image, and u_i

is the average pixel value of all transformed images at point *i* of the panorama. $\lambda > 0$; the parameter d(*) is used to control a regularization penalty term, which projects the absolute transformation matrix T_i computed from STN to a regular matrix θ .

Figure 3 shows the panoramic background image obtained from the above global motion compensation process. By comparison, it is evident that the proposed method achieves more refined inter-frame motion compensation results and a more complete panoramic background image.



Figure 3. Panoramic background image: (**a**) shows the panoramic image obtained by inter-frame alignment only, (**b**) shows the panoramic image after global optimization, and (**c**) shows the panoramic image after global optimization and residual transformation.

3.3. Local Spatial Detection

The foreground–background detection problem in the context of a moving camera can be simplified to the standard static camera foreground–background detection problem through motion compensation. The RPCA is commonly used for foreground object detection. Its core idea is to decompose the noise-containing matrix (video sequence) into a low-rank matrix (background of the video sequence) and a sparse matrix (foreground of the video sequence). Thus, the projected transformed image data can be decomposed as follows:

$$\min_{L,S} \|L\|_* + \zeta \|S\|_1 \quad \text{s.t.} \quad M = L + S$$
(8)

where $\|.\|_*$ represents the nuclear norm of the matrix, $\|.\|_1$ represents the *L*1 norm of the matrix, and ζ is the parameter that controls sparsity. The matrix solution process can be optimized using the Alternating Direction Method of Multipliers (ADMM) [62].

When the RPCA is used directly for a panoramic image, it frequently results in excessive computational complexity and insufficient memory. Therefore, the method divides the panorama size into several smaller sliding windows. Each window is shifted by *z* pixels horizontally or vertically, as shown in Figure 4.

All images are transformed by absolute projection to their corresponding positions on the panoramic image. The RPCA is applied to compute the local window background image for the area where the image coverage of the sliding window exceeds κ . During background subtraction, the local windows of the transformed image area are stitched together to obtain a local background image. The subtraction is performed between the transformed image area and the local background image within a range of *k* pixels around it. The minimum difference is computed as the final position, as shown in Equation (9):

$$\arg\min\sum_{i=1}^{D'} |x_{i-c} - p_i|$$
(9)

where D' represents the area obtained after performing absolute projection of the image, i denotes the pixel position at a certain point within the area, and c represents the offset distance from the pixel point i. The valid range for c is [0, k]. x is the pixel value of the pixel point of the image at position i - c, p_i is the local background pixel value corresponding to the x position. By varying the value of c, the minimum value of Equation (9) is obtained as the precise adjustment position.

Due to the lack of illumination in the shadow area of moving objects, the pixel values exhibit local inconsistency. We define T_h as the consistency of changes between the current

frame and the local background, which is used to eliminate shadow areas during the background subtraction process.

$$T_h = \frac{1}{N+M} \sum_{i=x-N}^{i=x+N} \sum_{i=y-M}^{i=y+M} (I(i,j) - B(i,j) - \delta(x,y))^2$$
(10)

Here, I(i, j) represents the pixel values at the position of the moving object in the panoramic image, while B(i, j) represents the pixel values of the local background that are consistent with the position of the moving object. The local difference between two frames is computed by averaging the pixel differences within a rectangular area ($N \times M$) around the corresponding pixels (x, y). This average is denoted as $\delta(x, y)$. If T_h exceeds a certain threshold, it is considered that the point on the motion object corresponds to a shadow.



Figure 4. The green bounding box indicates the position of the image frame in the panorama. The red bounding box represents a sliding window, which only computes the overlap between the sliding window and the transformed image (with an overlap area larger than κ). The non-overlapping areas (black areas) within the red bounding box and the areas outside the red bounding box are excluded from the computation. The results are then saved for each window.

3.4. Spatial Information Fusion

During local spatial subtraction, there is a possibility of low recall due to the similarity between the object pixel color and the background. By incorporating optical flow segmentation techniques, it is possible to accurately extract pixels of moving objects with high recall, even when their intensities are close to the background. The proposed method effectively integrates optical flow and background spatial information to optimize situations involving foreground false alarms, detection omissions, and unclear boundaries. The proposed method combines optical flow and background spatial information to optimize foreground false alarms, detection misses, and unclear boundaries. It aims to achieve a better balance between sensitivity and specificity in object detection.

Edge segmentation is performed on the coarse foreground and visualized optical flow field maps resulting from the above process. The degree of segmentation is determined based on the changes in the areas (adjacent areas of edge pixels) in the binary image, and the number of areas increases as the segmentation degree deepens. Then, at a different degree of segmentation, the intersection over union (IoU) between the two image areas is calculated based on whether they are adjacent in position. The final segmentation degree is determined by selecting the segmentation degree with the highest IoU. In other words, the binary segmentation degree that achieves the most stable changes in area blocks is selected for refining the foreground with greater precision. Thus, the area block division problem is solved by using the method of element grouping, and each area block is enclosed within a minimum enclosing circle. The binary processing and area segmentation results are shown in Figure 5.



Figure 5. Spatial classification results. Different segmentation areas are indicated by different colours.

After selecting the binary areas of the visualized optical flow field and the coarse foreground image, the corresponding areas between the two images are evaluated. This evaluation is based on the distance between their centroids and the radius relationship. The areas that satisfy Equation (11) are retained, while the areas that do not satisfy it are discarded.

$$\alpha |d| + \beta |r_1 - r_2| < \sigma \tag{11}$$

Here, *d* is the centroid distance between the corresponding areas of the two images, *r* is the respective radius, and α and β are the weights of the corresponding centroid distance and radius. Since the focus here is on the accuracy of the position, a higher weight is assigned to the centroid distance *d* compared with the weight assigned to the radius. The threshold value is denoted as σ .

Once the correct areas are identified, an iterative segmentation is performed using interactive GraphCut algorithms, such as GraphCut [63], OneCut [64], and GrabCut [65], on the minimum bounding rectangles of the selected areas. This process results in a refined foreground image, as shown in Figure 6.



(a) Image segmentation

(b) Foreground object

Figure 6. Interactive segmentation result: (**a**) shows the location situation after area selection, and (**b**) applies the GraphCut algorithm to the selected regions.

4. Experiments

In this section, we conducted extensive experimental evaluations on three datasets featuring fixed, jittery, and moving cameras, i.e., CDNET2014 [66], FBMS-59 [40], and CBD [67]. These datasets consist of various video sequences capturing both rigid and non-rigid moving objects from PTZ cameras, handheld cameras, and unmanned aerial vehicles (UAVs). Our method was also compared with state-of-the-art methods from recent years through qualitative and quantitative evaluations. Typical images from the experimental dataset are shown in Figure 7. Our experiments are run on a Windows 10 PC with AMD Ryzen 7 4800U with Radeon Graphics 1.8 GHz. The processing time for our computation is approximately several seconds per frame.

(a)CDNET2014Image: Simple simple

Figure 7. Typical frames of selected datasets.

4.1. Dataset and Metrics

The CDNET2014 [66] dataset is a dedicated video dataset for evaluating MOD methods with relatively fixed video capture devices. The dataset consists of 53 video sequences, categorized into 11 different complex background scenarios encountered in indoor and outdoor surveillance video environments, including Bad Weather (BW), Baseline (BL), Camera Jitter (CJ), Dynamic Background (DB), Intermittent Object Motion (IOM), Low Framerate (LF), Night Videos (NV), Pan–Tilt–Zoom (PTZ), Shadow (SH), Thermal (TH), and Turbulence (TU).

Then, the FBMS-59 [40] dataset is a collection of 59 video sequences specifically designed for motion segmentation. During the video capture process, the camera is in motion, involving translation, rotation, and scaling transformations. Similar to other methods that use partial data from the FBMS-59 dataset as a testing benchmark, the experiments were conducted only on the most commonly used video sequences (cars1-cars8, dogs01, dogs02, people1, and people2) [34,54,57]. There are two main reasons for this: (1) In some video sequences of the dataset, there are multiple moving objects present. However, only a subset of these objects is used for evaluation purposes, as including all of them would introduce additional false positives that could impact the final detection results. (2) A significant portion of the image (>70%) is occupied by foreground objects, which poses initialization challenges for most algorithms.

Then, the CBD [67] dataset is a collection of video sequences designed for MOD. It consists of five different scenes, including crowds, traffic, and other real-world moving objects. The presence of occlusions in the scenes poses significant challenges for MOD.

Similar to other papers, the R (Recall), P (Precision), and Fm (F-measure) metrics have been used for evaluation. R and P are measures of completeness and accurateness, respectively. Fm is a combination of R and P. These metrics are defined as follows:

$$R = \frac{TP}{TP + FN} \quad P = \frac{TP}{TP + FP} \quad Fm = 2 \times \frac{PR}{P + FP}$$

where TP, FP, and FN indicate true positive, false positive, and false negative, respectively. TP is the number of pixels correctly detected as foreground (moving object area). FP is the number of background pixels incorrectly detected as foreground. FN is the number of foreground pixels incorrectly detected as background.

In addition, we set the inter-frame matching number *n* to 5, the overlapping pixel threshold ρ to 15, the penalty term coefficient λ to 0.5, the sparsity control parameter ζ to 0.6, the offset range *k* to 5, α and β to 0.4 and 0.6, and *sigma* to 10. All parameter settings were fixed for all experiments to ensure the consistency and reproducibility of our results.

4.2. Experimental Analysis of MOD under Fixed and Jittered Cameras

4.2.1. Quantitative Analysis

In this paper, our method is compared with state-of-the-art methods, including incPCP [7], SWCD [9], AdMH [10], FBS-ABL [6], t-OMoGMF [5], FgSegNet v2 [18], and BSUV-Net [17] for quantitative analysis under fixed cameras on the CDNET2014 dataset. These methods represent typical, recent mainstream, and deep learning methods. The Fm values of each method in different categories of the CDNET2014 dataset are presented in Table 1. According to Table 1, it can be observed that our method achieves better performance in most video categories. The average performance of our method outperforms all comparison methods, with an Fm of 0.8027. It demonstrates overall robustness, performing well in categories such as BW, BL, DB, PTZ, and SH. Compared with the second-best method BSUV-Net, which performs remarkably well in fixed scenes, our method has increased the average Fm by 0.0159. In the BL category, the BSUV-net achieves an impressive Fm of 0.9693. However, its performance in the PTZ category is less satisfactory, with an Fm of only 0.6282. The decrease in detection accuracy in the PTZ category is due to the non-rigid transformations that foreground objects may undergo in such scenarios. Moreover, the BSUV-Net heavily relies on the quality of semantic information, which needs to be incorporated as prior knowledge during training. Therefore, its detection performance is highly dependent on the quality of semantic information. Meanwhile, the traditional background model-based methods such as SWCD, FBS-ABL, incPCP, AdMH, and t-OMoGMF also exhibit unsatisfactory detection performance in the PTZ category. In the category BW, it can be seen that our method can still accurately detect moving objects under bad weather conditions, with a detection accuracy reaching 0.8873. The proposed method in this paper fully utilizes spatial information for complementary advantages and reduces the impact of non-rigid transformations. The deep-learning-based method FgSegNet v2 shows low overall detection accuracy in unsupervised scenarios, making it difficult to detect well-defined foreground objects.

Table 1. Quantitative analysis of the Fm values for each method on the CDNET2014 dataset. To denote the best Fm for each category of videos, use bold formatting, and for the second-best, use underline.

Mathada	Videos											
Methous	BW	BL	CJ	DB	IOM	LF	NV	PTZ	SH	TH	TU	Overall
incPCP [7]	0.7324	0.8287	0.5684	0.6844	0.5691	0.5767	0.4736	<u>0.6514</u>	0.7154	0.7436	0.6247	0.6524
SWCD [9]	0.8233	<u>0.9214</u>	0.7411	0.8645	0.7092	0.7374	0.5807	0.4545	0.8779	0.8581	0.7735	0.7583
AdMH [10]	0.5600	0.7900	0.7150	0.7750	0.4900	0.4500	0.2600	0.0900	0.6900	0.6900	0.6325	0.5584
FBS-ABL [6]	0.8106	0.8649	0.5298	0.7424	0.7232	0.6328	0.5272	0.3267	0.8671	0.6619	0.5564	0.6585
t-OMoGM [5]	0.7649	0.8027	0.7060	0.7126	0.7348	<u>0.7418</u>	0.5413	0.5843	0.6143	0.7346	0.5466	0.6789
FgSegNet v2 [18]	0.3277	0.6926	0.4266	0.3634	0.2002	0.2482	0.2800	0.3503	0.5295	0.6038	0.0643	0.3715
BSUV-Net [17]	0.8713	0.9693	<u>0.7743</u>	0.7967	0.7499	0.6797	0.6987	0.6282	<u>0.9233</u>	0.8581	<u>0.7051</u>	0.7868
OURS	0.8873	0.8724	0.7962	0.8759	0.6751	0.7618	0.6584	0.8032	0.9376	0.7791	0.7825	0.8027

From Table 1, it can be observed that most methods perform poorly in the NV and IOM categories. This is because the NV category contains strong glare halos due to low visibility. In the IOM category, methods may mistakenly identify long-term stationary foreground objects as background, which significantly affects the detection performance.

Additionally, we specifically analyze the impact of the camera jitter and perform performance analysis using four camera jitter video sequences from the CDNET2014 dataset: badminton, boulevard, sidewalk, and traffic. The quantitative comparative analysis results are shown in Figure 8. According to Figure 8, our proposed method demonstrates favorable performance on the jitter video sequences. Compared with other methods, our approach achieves Fm values above 0.7 for each video sequence class, with an average value of 0.7962, showing a better detection performance. In contrast, the values obtained from AdMH indicate the instability of its detection performance. When there is significant camera jitter and image frame blur in a video, as seen in the "sidewalk" example, the detection performance of most algorithms suffers. In contrast, our proposed method maintains a relatively stable detection level overall.



Figure 8. Performance of our method compared with other methods on jittery video sequences.

4.2.2. Qualitative Analysis

Figure 9 shows the foreground results obtained by our method on each category of the CDNET2014 dataset. Our method shows superior performance in detecting foreground object contours for the video categories BL, PTZ, BW, SH, CJ, and DB, independent of external environments such as bad weather. Notably, it exhibits the ability to accurately detect smaller foreground objects, such as LF. In the categories of IOM, TH, and TU, a few instances of false positives were detected. However, overall, our method demonstrates improved robustness in background restoration, as well as reduced noise and shadow effects in video frames. It exhibits strong adaptability to various conditions. The foreground objects detected in the PTZ and NV categories are slightly larger than the ground truth. This is due to the inherent perspective changes that occur as the objects move, causing shape deformations. Additionally, the pixel similarity between the shadow areas of these objects and the background limits the effectiveness of local inconsistency detection, as discrepancies of less than 5 pixels are insufficient to trigger such detections.



Figure 9. The foreground detection results of our method on the CDNET2014 dataset.

The experiment also included testing on the office video sequences from CDNET2014 under different illumination conditions, and the results are shown in Figure 10. It can be observed that the larger the illumination scaling factor |Alpha| value, the clearer the contours shown in the residual images. This indicates a greater disparity in illumination conditions between the illumination images and the adaptive background image. By adaptively processing multiple image frames, our method gradually obtains the most suitable background condition, demonstrating a certain level of resistance to variations in lighting.



Figure 10. Performance tests with different illumination conditions: (**a**) is the original image sequence. (**b**) is the collection of images processed with different illumination scaling factors (Alpha). (**c**) is the background image obtained by our method under different illumination. (**d**) is the corresponding residual images.

4.3. Experimental Analysis of MOD under Freely Moving Cameras

Tables 2 and 3 show a performance comparison of our method under moving cameras with MLBS [57], IFB [58], JA-POLS [37], Sugimura [31], CAG-DDE [51], DS-Net [48], LOCATE [45], and LTA [52] on the FBMS-59 and CBD datasets. CAG-DDE and DS-Net are deep learning methods that use optical flow features as inputs. The average value of multiple measurements is taken as the final Fm. In the IFB algorithm, specifically, the average of the SIFT, SURF, and KAZE feature matching algorithms is used as the final result. Figure 11 qualitatively illustrates the comparative results of various methods on typical frames from the FBMS-59 dataset.

Table 2. Quantitative Analysis of the Fm values for each method on the FBMS-59 dataset. To denote the best Fm for each category of videos, use bold formatting, and for the second-best, use underline.

Mathada	Videos												
Wiethous	cars1	cars2	cars3	cars4	cars5	cars6	cars7	cars8	dogs01	dogs02	people1	people2	Overall
MLBS [57]	0.9204	0.9016	<u>0.9316</u>	<u>0.9155</u>	0.8662	<u>0.9223</u>	0.9117	0.8593	<u>0.8200</u>	0.8200	0.8138	0.9434	<u>0.8855</u>
IFB [58]	0.6700	0.8167	0.6900	0.8267	0.7333	0.6267	0.7482	0.7103	0.7837	0.9200	0.5626	0.7315	0.7350
JA-POLS [37]	0.5325	0.6587	0.6851	0.7189	0.5126	0.5178	0.4548	0.3671	0.6376	0.3149	0.6012	0.6493	0.5542
Sugimura [31]	0.9010	0.9020	0.9550	0.8740	0.9090	0.8790	0.9150	0.9210	0.8143	0.8285	0.8020	0.8820	0.8819
CAG-DDE [51]	0.9312	<u>0.9186</u>	0.6605	0.9110	0.4358	0.8634	0.8946	0.9371	0.8154	0.8146	0.8738	0.8454	0.8251
DS-Net [48]	0.8922	0.7268	0.4965	0.9089	0.9154	0.8639	0.9146	0.8814	0.7548	0.8459	0.9074	0.8154	0.8269
LOCATE [45]	0.8742	0.7589	0.8473	0.6146	0.9018	0.8356	0.8842	0.8174	0.7732	0.7958	0.8092	0.8467	0.8132
LTS [52]	0.7598	0.5178	0.6734	0.5489	0.8067	0.7256	0.5874	0.6793	0.7421	0.7972	0.8126	0.8298	0.7067
OURS	0.9359	0.9225	0.8846	0.9231	0.9168	0.9384	0.9389	<u>0.9293</u>	0.8530	<u>0.8661</u>	0.8358	<u>0.8965</u>	0.9034

Table 3. Quantitative analysis of the Fm values for each method on the CBD dataset. To denote the best Fm for each category of videos, use bold formatting, and for the second-best, use underline.

Mathada			Vic	leos		
Methods	Drive	Forest	Parking	Store	Traffic	Overall
MLBS [57]	0.6595	0.7220	0.8366	0.8628	0.4819	0.7126
JA-POLŠ [37]	0.2101	0.1584	0.1456	0.2876	0.3487	0.2301
Šugimura [31]	0.8880	0.8300	0.8110	0.7590	0.5580	0.7690
CAĞ-DDE [51]	0.0765	0.8507	0.4123	0.2458	0.5714	0.4313
DS-Net [48]	0.2654	0.8546	0.1769	0.1137	0.0624	0.2946
LOCATE [45]	0.6813	0.7139	0.4661	0.6764	0.5312	0.6138
LTS [52]	0.3488	0.6795	0.2746	0.5163	0.3867	0.4224
OURS	0.7968	0.8159	0.8566	0.8743	0.6381	0.7963

and the second se				-					
	-	-			~	-		~	•
	-		•			-		· · · 2°,	-
								C	
		-	-				4		•
	<u>}</u>	•			ł		ţ.	\$	*
	M	•	4		•	N.	Č	ţ	*
Original	Ground Truth	MLBS	IFB	JA-POLS	CAG-DDE	DS-Net	LOCATE	LTS	OURS

Figure 11. Qualitative comparison of the proposed method with other MOD methods on FBMS-59 dataset. From left to right: Original, Ground Truth, MLBS [57], IFB [58], JA-POLS [37], Sugimura [31], CAG-DDE [51], DS-Net [48], LOCATE [45], LTS [52], and our method.

As can be seen from Table 2, our method outperforms the compared methods in most of the video sequences, achieving an overall detection accuracy 0.0179 higher than that of the second-best method. Particularly, it demonstrates strong robustness in the case of rigid motion video sequences, such as cars1–cars8. For non-rigid motion (e.g., people and dogs), the IFB and JA-POLS methods may produce partial false negatives in the foreground due to static areas within the moving objects. Consequently, the Fm values for these methods are only around 0.6. MLBS and Sugimura use hybrid methods to handle moving objects, no longer relying solely on a single technique for detection, which allows them to adapt better to complex environments. Even so, these two methods are not as effective as our method in detecting both rigid and non-rigid motions. Our method effectively combines the advantages of background and optical flow segmentation, mitigating the impact of non-rigid motion to a certain extent. Due to the limited number of video frames, the IFB and JA-POLS methods lack sufficient images to generate statistical models, which affects the extraction of foreground objects. JA-POLS is also prone to the influence of the surrounding environment and background color, leading to the occurrence of holes in the foreground objects. Further, LOCATE relies on optical flow information for detection, thus the obtained object information is coarse. On the other hand, the method LTS primarily relies on pixel changes to detect foreground objects. Therefore, its detection performance may be affected in dynamic background situations, resulting in an overall accuracy of only 0.7067. From Figure 11, it is evident that the proposed method combines the advantages of low-rank matrix decomposition and optical flow to effectively eliminate foreground false alarms and obtain accurate foreground contours. Particularly, for rigid motion (such as cars), it exhibits a better ability to restore its original shape. The methods CAG-DDE and DS-Net, similar in nature, achieve an Fm of over 0.9 for larger objects. However, the detection performance significantly deteriorates when multiple moving objects are present in the scene or when the size of the moving objects is small (such as cars3 and cars5). Furthermore, the performance was evaluated on the occlusion dataset CBD, as shown in Table 3. The proposed algorithm maintains steady performance even in complex background circumstances on mobile cameras. It faces challenges in detecting moving objects with significant occlusions, which proves to be difficult for methods like JA-POLS, CAG-DDE, DS-Net, LOCATE, and LTS.

4.4. Analysis of Spatial Fusion Threshold and Ablation Study

4.4.1. Analysis of Threshold σ for Spatial Information Fusion

The spatial relationship in spatial information fusion is determined by the threshold parameter σ in Equation (10). Figure 12 shows the variation in precision and recall for different σ . From the graph, it can be observed that the accuracy and recall remain relatively stable in the range of 2–18 for σ . This allows Fm to maintain a high level of measurement accuracy. However, once the threshold value exceeds 18, the accuracy starts to decrease significantly. This is because increasing the threshold expands the spatial matching range, causing more background pixels to be classified as foreground objects, leading to an increase in false positives.

4.4.2. Ablation Study

We set up Test1–Test5 to verify the effectiveness of the global motion compensation optimization and the interactive graph cut algorithms (GraphCut, OneCut, and GrabCut) in our method. All three algorithms possess good interactivity and controllability, enabling easy integration with our algorithm to guide them in image segmentation. They all adopt a global optimization strategy, allowing for the consideration of relationships between pixels at a global scale, thereby obtaining accurate segmentation results. We conducted experiments using 12 representative video sequences (cars1–cars8, dogs01, dogs02, people1, and people2) from the FBMS-59 dataset. The qualitative results of the experimental part are illustrated in Figure 13. The quantitative evaluation of Fm values is presented in Figure 14. In addition, the experiments also evaluated the average time performance of the three



graph cut methods we used and other graph cut algorithms for single segmentation in the video sequence "cars1" with a resolution of 640×480 . The results are presented in Table 4.

Figure 12. Precision and recall curves for different values of the threshold parameter σ .

From Figure 13, it can be observed that, compared with using only projective transformation, our proposed motion compensation optimization method improves the accuracy of foreground object detection and reduces some false positives. Furthermore, the combination with graph cut clearly demonstrates a significant improvement in detection performance. Moreover, as shown in the quantitative results of the algorithm modules in Figure 14, it can be observed that, in terms of segmentation accuracy, the OneCut and GrabCut segmentation algorithms outperform the GraphCut algorithm overall. The performances of OneCut and GrabCut segmentation are comparable, with both achieving similar segmentation results. Table 4 reveals that, under the same experimental conditions, the OneCut algorithm exhibits slightly faster time performance compared with GrabCut. However, considering that OneCut and GraphCut perform single segmentation, while GrabCut allows for multiple iterative segmentations, GrabCut demonstrates better control over details and can handle more complex segmentation scenarios. Therefore, when conducting experiments, a reasonable choice should be made based on the specific circumstances. Moreover, compared with other methods using graph cuts, our approach maintains a considerable detection speed even with limited computational resources (Windows 10 PC with AMD Ryzen 7 4800U with Radeon Graphics 1.8 GHz). Optimization measures such as object pooling and caching were implemented in the code to improve memory usage efficiency (peak memory usage is 341.2 MiB, and incremental memory usage is 227.9 MiB). We believe that these results highlight the robustness and applicability of our approach, even under challenging computational conditions. In the future, running our experiments on a more powerful server will significantly reduce processing times and improve overall performance.



Figure 13. Test 1 represents the usage of only the projection transformation and local background subtraction. Test 2 involves the addition of the motion compensation optimization framework and local background subtraction. Tests 3, 4, and 5 denote the incorporation of local spatial information fusion based on Test 2, employing different interactive graph cut algorithms (GraphCut, OneCut, and GrabCut), respectively.





Table 4. Average time performance of different interactive graph cut algorithms for one-time segmentation. The best performance is in bold and the next best performance is underlined.

Method	GraphCut [63]	OneCut [64]	GrabCut [65]	MLBS [57]	JA-POLS [37]	Sugimura [31]	LOCATE [45]
Time (ms)	7101.96	<u>1369.52</u>	2847.61	6856.73	16,913.64	5178.46	411.29

5. Conclusions

In this paper, we have proposed a novel motion object detection algorithm for mobile cameras that effectively integrates local and global visual information for foreground/background segmentation. The superiority of the method is demonstrated qualitatively and quantitatively through experiments using three public datasets. The experimental results show that the model performs well in various challenging scenarios, including poor illumination conditions, abrupt changes in illumination, camera jitter, shadows, and both fixed and mobile cameras. The model exhibits good performance and scalability. As our method relies on the differentiation of pixel points to identify continuously moving objects, it is challenging to detect disguised or intermittent moving objects effectively. In future work, we plan to incorporate object feature information such as shape, texture, and motion patterns to complement pixel-level analysis. By leveraging richer feature representations, the more accurate and robust detection of moving objects could be achieved, particularly in scenarios where traditional pixel-based methods may struggle. These enhancements may not only improve the reliability of moving object detection but also extend the applicability of our method to a wider range of dynamic scenes and environmental conditions.

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

Funding: This work was financially supported by the National Natural Science Foundation of China under Grants 62272426 and 62106238; in part by the Shanxi Scholarship Council of China under Grant 2020-113; in part by the Natural Science Foundation of Shanxi Province under Grants 202203021212138, 202303021211153, and 202203021222027; in part by the Shanxi Province Science and Technology Major Project under Grant 202201150401021; in part by the Graduate Student Innovation Project of Shanxi Province under Grant NO.2023; and in part by the Shanxi Province Science and Technology Achievement Transformation Guiding Project under Grant 202104021301055.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this study are the publicly available CDNET2014 dataset, FBMS-59 dataset, and CBD dataset. They can be downloaded at the following links: (1) CDNET2014 dataset (http://www.changedetection.net/, (accessed on 15 February 2023)), (2) FBMS-59 dataset (https://lmb.informatik.uni-freiburg.de/resources/datasets/, (accessed on 15 February 2023)), (3) CBD dataset (https://vis-www.cs.umass.edu/motionSegmentation/complexBgVideos. html, (accessed on 15 February 2023)).

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

References

- 1. Chapel, M.N.; Bouwmans, T. Moving objects detection with a moving camera: A comprehensive review. *Comput. Sci. Rev.* 2020, 38, 100310. [CrossRef]
- Wu, Y.; Lim, J.; Yang, M.H. Online object tracking: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
- Gu, S.; Ma, J.; Hui, G.; Xiao, Q.; Shi, W. STMT: Spatio-temporal memory transformer for multi-object tracking. *Appl. Intell.* 2023, 53, 23426–23441. [CrossRef]
- Bideau, P.; Learned-Miller, E. It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part VIII 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 433–449.
- Yong, H.; Meng, D.; Zuo, W.; Zhang, L. Robust online matrix factorization for dynamic background subtraction. *IEEE Trans. Pattern Anal. Mach. Intell.* 2017, 40, 1726–1740. [CrossRef]
- 6. Montero, V.J.; Jung, W.Y.; Jeong, Y.J. Fast background subtraction with adaptive block learning using expectation value suitable for real-time moving object detection. *J. Real-Time Image Process.* **2021**, *18*, 967–981. [CrossRef]
- Rodriguez, P.; Wohlberg, B. Incremental principal component pursuit for video background modeling. J. Math. Imaging Vis. 2016, 55, 1–18. [CrossRef]
- 8. Wang, J.; Zhao, Y.; Zhang, K.; Wang, Q.; Li, X. Spatio-temporal online matrix factorization for multi-scale moving objects detection. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 743–757. [CrossRef]
- Işık, Ş.; Özkan, K.; Günal, S.; Gerek, Ö.N. SWCD: A sliding window and self-regulated learning-based background updating method for change detection in videos. J. Electron. Imaging 2018, 27, 023002. [CrossRef]
- 10. He, W.; Li, W.; Zhang, G.; Tu, B.; Kim, Y.K.; Wu, J.; Qi, Q. Detection of moving objects using adaptive multi-feature histograms. *J. Vis. Commun. Image Represent.* **2021**, *80*, 103278. [CrossRef]
- 11. Huang, Y.; Jiang, Q.; Qian, Y. A novel method for video moving object detection using improved independent component analysis. *IEEE Trans. Circuits Syst. Video Technol.* 2020, *31*, 2217–2230. [CrossRef]
- 12. Delibasoglu, I. Real-time motion detection with candidate masks and region growing for moving cameras. *J. Electron. Imaging* **2021**, *30*, 063027. [CrossRef]

- 13. Chen, C.; Song, J.; Peng, C.; Wang, G.; Fang, Y. A novel video salient object detection method via semisupervised motion quality perception. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 2732–2745. [CrossRef]
- An, Y.; Zhao, X.; Yu, T.; Guo, H.; Zhao, C.; Tang, M.; Wang, J. ZBS: Zero-shot Background Subtraction via Instance-level Background Modeling and Foreground Selection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 6355–6364.
- Mondal, A.; R, S.; Giraldo, J.H.; Bouwmans, T.; Chowdhury, A.S. Moving object detection for event-based vision using graph spectral clustering. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 876–884.
- Giraldo, J.H.; Javed, S.; Werghi, N.; Bouwmans, T. Graph CNN for moving object detection in complex environments from unseen videos. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 225–233.
- Tezcan, O.; Ishwar, P.; Konrad, J. BSUV-Net: A fully-convolutional neural network for background subtraction of unseen videos. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass, CO, USA, 1–5 March 2020; pp. 2774–2783.
- Lim, L.A.; Keles, H.Y. Learning multi-scale features for foreground segmentation. *Pattern Anal. Appl.* 2020, 23, 1369–1380. [CrossRef]
- 19. He, J.; Zhang, D.; Balzano, L.; Tao, T. Iterative Grassmannian optimization for robust image alignment. *Image Vis. Comput.* **2014**, 32, 800–813. [CrossRef]
- 20. Eltantawy, A.; Shehata, M.S. An accelerated sequential PCP-based method for ground-moving objects detection from aerial videos. *IEEE Trans. Image Process.* 2019, *28*, 5991–6006. [CrossRef]
- Chen, Y.; Zhu, X.; Li, Y.; Wei, Y.; Ye, L. Enhanced semantic feature pyramid network for small object detection. *Signal Process. Image Commun.* 2023, 113, 116919. [CrossRef]
- Shakeri, M.; Zhang, H. Moving object detection under discontinuous change in illumination using tensor low-rank and invariant sparse decomposition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7221–7230.
- 23. Sahoo, S.; Nanda, P.K. Adaptive feature fusion and spatio-temporal background modeling in KDE framework for object detection and shadow removal. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 1103–1118. [CrossRef]
- 24. Moore, B.E.; Gao, C.; Nadakuditi, R.R. Panoramic robust pca for foreground–background separation on noisy, free-motion camera video. *IEEE Trans. Comput. Imaging* **2019**, *5*, 195–211. [CrossRef]
- 25. Nakaya, Y.; Harashima, H. Motion compensation based on spatial transformations. *IEEE Trans. Circuits Syst. Video Technol.* **1994**, *4*, 339–356. [CrossRef]
- Liu, F.; Gleicher, M. Learning color and locality cues for moving object detection and segmentation. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 320–327.
- Zamalieva, D.; Yilmaz, A.; Davis, J.W. A multi-transformational model for background subtraction with moving cameras. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Proceedings, Part I 13; Springer: Berlin/Heidelberg, Germany, 2014, pp. 803–817.
- Lezki, H.; Ahu Ozturk, I.; Akif Akpinar, M.; Kerim Yucel, M.; Berker Logoglu, K.; Erdem, A.; Erdem, E. Joint exploitation of features and optical flow for real-time moving object detection on drones. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8 September 2018.
- 29. Beauchemin, S.S.; Barron, J.L. The computation of optical flow. ACM Comput. Surv. (CSUR) 1995, 27, 433-466. [CrossRef]
- 30. Shen, J.; Peng, J.; Shao, L. Submodular trajectories for better motion segmentation in videos. *IEEE Trans. Image Process.* **2018**, 27, 2688–2700. [CrossRef]
- 31. Sugimura, D.; Teshima, F.; Hamamoto, T. Online background subtraction with freely moving cameras using different motion boundaries. *Image Vis. Comput.* **2018**, *76*, 76–92. [CrossRef]
- 32. Rosen, D.M.; Carlone, L.; Bandeira, A.S.; Leonard, J.J. SE-Sync: A certifiably correct algorithm for synchronization over the special Euclidean group. *Int. J. Robot. Res.* **2019**, *38*, 95–125. [CrossRef]
- 33. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial transformer networks. *Adv. Neural Inf. Process. Syst.* 2015, 28, 1–9.
- 34. Zhang, W.; Sun, X.; Yu, Q. Moving object detection under a moving camera via background orientation reconstruction. *Sensors* **2020**, *20*, 3103. [CrossRef]
- 35. Zhou, Z.; Li, X.; Wright, J.; Candes, E.; Ma, Y. Stable principal component pursuit. In Proceedings of the 2010 IEEE International Symposium on Information Theory, Austin, TX, USA, 13–18 June 2010; IEEE: Piscataway, NJ, USA, 2010, pp. 1518–1522.
- 36. Yu, Y.; Kurnianggoro, L.; Jo, K.H. Moving object detection for a moving camera based on global motion compensation and adaptive background model. *Int. J. Control. Autom. Syst.* **2019**, *17*, 1866–1874. [CrossRef]
- Chelly, I.; Winter, V.; Litvak, D.; Rosen, D.; Freifeld, O. JA-POLS: A moving-camera background model via joint alignment and partially-overlapping local subspaces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12585–12594.
- 38. Candès, E.J.; Li, X.; Ma, Y.; Wright, J. Robust principal component analysis? J. ACM (JACM) 2011, 58, 1–37. [CrossRef]

- Cui, X.; Huang, J.; Zhang, S.; Metaxas, D.N. Background subtraction using low rank and group sparsity constraints. In Proceedings of the Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; Proceedings, Part I 12; Springer: Berlin/Heidelberg, Germany, 2012; pp. 612–625.
- Ochs, P.; Malik, J.; Brox, T. Segmentation of moving objects by long term video analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 2013, 36, 1187–1200. [CrossRef]
- 41. Keuper, M.; Andres, B.; Brox, T. Motion trajectory segmentation via minimum cost multicuts. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3271–3279.
- Mandal, M.; Kumar, L.K.; Vipparthi, S.K. Mor-uav: A benchmark dataset and baselines for moving object recognition in uav videos. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 2626–2635.
- Elqursh, A.; Elgammal, A. Online moving camera background subtraction. In Proceedings of the Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; Proceedings, Part VI 12; Springer: Berlin/Heidelberg, Germany, 2012, pp. 228–241.
- Zhang, W.; Sun, X.; Yu, Q. Accurate moving object segmentation in unconstraint videos based on robust seed pixels selection. *Int. J. Adv. Robot. Syst.* 2020, 17, 1729881420947273. [CrossRef]
- 45. Singh, S.; Deshmukh, S.V.; Sarkar, M.; Krishnamurthy, B. LOCATE: Self-supervised Object Discovery via Flow-guided Graph-cut and Bootstrapped Self-training. In Proceedings of the 34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, 20–24 November 2023; BMVA: London, UK, 2023.
- Dave, A.; Tokmakov, P.; Ramanan, D. Towards segmenting anything that moves. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019.
- Zhang, J.; Xu, C.; Gao, Z.; Rodrigues, J.J.; de Albuquerque, V.H.C. Industrial pervasive edge computing-based intelligence IoT for surveillance saliency detection. *IEEE Trans. Ind. Inform.* 2020, 17, 5012–5020. [CrossRef]
- Liu, J.; Wang, J.; Wang, W.; Su, Y. DS-Net: Dynamic spatiotemporal network for video salient object detection. *Digit. Signal Process.* 2022, 130, 103700. [CrossRef]
- 49. Gu, Y.; Wang, L.; Wang, Z.; Liu, Y.; Cheng, M.M.; Lu, S.P. Pyramid constrained self-attention network for fast video salient object detection. *Proc. AAAI Conf. Artif. Intell.* 2020, 34, 10869–10876. [CrossRef]
- Zhang, Y.P.; Chan, K.L. Saliency detection with moving camera via background model completion. Sensors 2021, 21, 8374. [CrossRef] [PubMed]
- Chen, P.; Lai, J.; Wang, G.; Zhou, H. Confidence-guided adaptive gate and dual differential enhancement for video salient object detection. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; IEEE: Piscataway, NJ, USA, 2021, pp. 1–6.
- 52. Dong, G.; Zhao, C.; Pan, X.; Basu, A. Learning Temporal Distribution and Spatial Correlation Towards Universal Moving Object Segmentation. *IEEE Trans. Image Process.* 2024, 33, 2447–2461. [CrossRef] [PubMed]
- 53. Delibaşoğlu, İ. Moving object detection method with motion regions tracking in background subtraction. *Signal, Image Video Process.* **2023**, *17*, 2415–2423. [CrossRef]
- 54. Cui, Z.; Jiang, K.; Wang, T. Unsupervised Moving Object Segmentation from Stationary or Moving Camera Based on Multi-frame Homography Constraints. *Sensors* **2019**, *19*, 4344. [CrossRef]
- 55. Sajid, H.; Cheung, S.C.S.; Jacobs, N. Motion and appearance based background subtraction for freely moving cameras. *Signal Process. Image Commun.* **2019**, *75*, 11–21. [CrossRef]
- Makino, K.; Shibata, T.; Yachida, S.; Ogawa, T.; Takahashi, K. Moving-object detection method for moving cameras by merging background subtraction and optical flow methods. In Proceedings of the 2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Montreal, QC, Canada, 14–16 November 2017; IEEE: Piscataway, NJ, USA, 2017, pp. 383–387.
- 57. Zhu, Y.; Elgammal, A. A multilayer-based framework for online background subtraction with freely moving cameras. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5132–5141.
- 58. Zhao, C.; Sain, A.; Qu, Y.; Ge, Y.; Hu, H. Background subtraction based on integration of alternative cues in freely moving camera. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 1933–1945. [CrossRef]
- 59. Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. Lect. Notes Comput. Sci. 2006, 3951, 404–417.
- 60. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
- 61. Detlefsen, N.S.; Freifeld, O.; Hauberg, S. Deep diffeomorphic transformer networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4403–4412.
- 62. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. In *Foundations and Trends*[®] *in Machine Learning*; Stanford University: Stanford, CA, USA, 2011; Volume 3, pp. 1–122.
- 63. Peng, Z.; Qu, S.; Li, Q. Interactive image segmentation using geodesic appearance overlap graph cut. *Signal Process. Image Commun.* **2019**, *78*, 159–170. [CrossRef]
- Deshpande, A.; Dahikar, P.; Agrawal, P. An Experiment with Random Walks and GrabCut in One Cut Interactive Image Segmentation Techniques on MRI Images. In *Computational Vision and Bio-Inspired Computing: ICCVBIC 2019*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 993–1008.

- 65. Rother, C.; Kolmogorov, V.; Blake, A. "GrabCut" interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* (*TOG*) **2004**, *23*, 309–314. [CrossRef]
- 66. Goyette, N.; Jodoin, P.M.; Porikli, F.; Konrad, J.; Ishwar, P. A novel video dataset for change detection benchmarking. *IEEE Trans. Image Process.* **2014**, *23*, 4663–4679. [CrossRef] [PubMed]
- Narayana, M.; Hanson, A.; Learned-Miller, E. Coherent motion segmentation in moving camera videos using optical flow orientations. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 1577–1584.

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.