

# Multiple Properties-Based Moving Object Detection Algorithm

Changjian Zhou\*, Jinge Xing\*, and Haibo Liu\*\*

## Abstract

Object detection is a fundamental yet challenging task in computer vision that plays an important role in object recognition, tracking, scene analysis and understanding. This paper aims to propose a multiproperty fusion algorithm for moving object detection. First, we build a scale-invariant feature transform (SIFT) vector field and analyze vectors in the SIFT vector field to divide vectors in the SIFT vector field into different classes. Second, the distance of each class is calculated by dispersion analysis. Next, the target and contour can be extracted, and then we segment the different images, reversal process and carry on morphological processing, the moving objects can be detected. The experimental results have good stability, accuracy and efficiency.

## Keywords

Moving Object Detection, Multiple Properties, SIFT Vector Field

## 1. Introduction

Object detection requires the algorithm to predict a bounding box with a category label for each instance of interest in an image [1]. The moving object detection algorithm also plays an important role in computer vision [2]. This paper, based on the scale-invariant feature transform (SIFT) vector field algorithm, analyzes vectors in the field and cluster dispersion, which can more easily detect moving targets. In moving object contour extraction, we grayscale process the two image frames, and motions compensate for two original images. After that, the image difference between the two gray frame images is determined to require the images to be differenced by image. Then, we segment the different images, reverse process and carry out other morphological processing. Finally, we isolate each object and extract each object outline.

## 2. Related Work

### 2.1 Cluster Analysis

Cluster analysis is based on an objective set of several natural classes, and some properties of every

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received December 11, 2018; first revision November 21, 2019; second revision March 23, 2020; accepted April 7, 2020.

**Corresponding Author:** Jinge Xing ([xingjinge@neau.edu.cn](mailto:xingjinge@neau.edu.cn))

\* Dept. of Modern Educational Technology, Northeast Agricultural University, Harbin, China ([zhouchangjian@neau.edu.cn](mailto:zhouchangjian@neau.edu.cn), [xingjinge@neau.edu.cn](mailto:xingjinge@neau.edu.cn))

\*\* College of Computer Science and Technology, Harbin Engineering University, Harbin, China ([liuhaibo@hrbeu.edu.cn](mailto:liuhaibo@hrbeu.edu.cn))

individual in each class have strong similarity to establish a kind of data description method [3]. Therefore, from the principle perspective, cluster analysis divides some models that are given into several groups. For selected properties or features, each group has a similar model but has considerable differences from other groups. In this method, the one-class cluster analysis algorithm is very intuitive and simple. The similarity of various attributes or characteristics in models determine how to classify them; similar models are in one class, and different models are in different classes [4]. Thus, the classification model is divided into several non-overlapping subsets. In addition, the other cluster method defines the appropriate criteria and makes use of mathematical tools or relevant statistical concepts and principles in classification.

The basic elements of cluster analysis are feature extraction, pattern similarity measurement, the distance between point and class, the distance between class and class, cluster criteria, cluster algorithms and validity analysis.

## 2.2 Dispersion Analysis

Dispersion is generally used to describe a set of data distributions and magnitudes of change and shows the degree of deviation from the mean from a set of data. Commonly, measures used in dispersion are range, sum of squares of deviation from mean, variance, standard deviation and coefficient of variation. This paper uses the dispersion analysis of the specific implementation steps as follows:

- 1) Input a SiftClass from ShiftClassSet.
- 2) Determine the number of vectors ( $n$ ) in this classification set. If  $n < 10$ , remove them from the classification set. Otherwise, go to 3).
- 3) Calculate the interclass distance in the ShiftClass.
- 4) Determine whether the classifications are not reaching the end; if not, go to 1) and continue. Otherwise, go to 5).
- 5) Sort the ShiftClassSet descending according to the size.
- 6) Set the first classification in ShiftClassSet to background and others to the goal.
- 7) Dispersion analysis is completed.

## 2.3 Motion Compensation

Motion compensation is an adjacent frames method in different ways. Specifically, it describes how each piece in the former frame moves to one position in the current frame. The method is used to reduce spatial redundancy in video sequences, video compression or video coding [5].

The inchoate motion compensation design simply subtracts the reference frame from the current frame and generally gains less energy (or information) of the “residual” to encode by the lower bit rate. The decoder can be fully restored by the simple addition coded frame.

A more complex design estimates the whole moving frame scene and moving objects in the frame and moves these movements into the code stream by certain parameters. This predicted value of the pixel on the frame is generated by the reference frame that has a certain displacement of the corresponding pixel value. This method can gain less residual and better compression ratio than simple subtraction. Of course, the parameters used to describe the movement cannot occupy too many streams; otherwise, it offsets the complexity of motion estimation benefits.

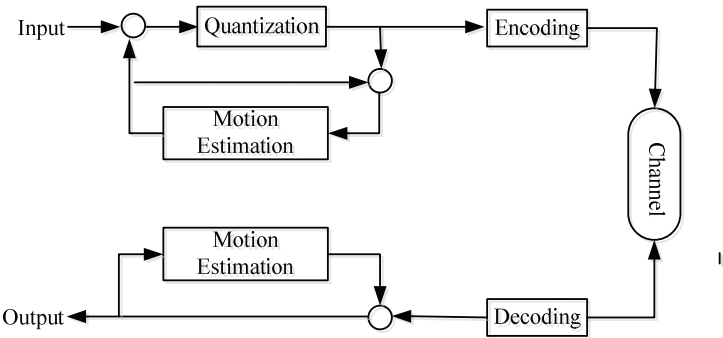
Usually, processing the image frame is group by group, and the first frame in each group does not use the motion estimation method in coding, such as the intraframe coding frame or I frame (intraframe). Other frames in this group use interframe coding frames (interframes), usually called P-frames. This encoding is often called IPPPP, which means that the first frame in encoding is the I frame and the others are P-frames.

In most projection times, not only can the past frame be used to predict the current frame but the future frame is also used to predict the current frame. Of course, when encoding, the future frame must be encoded earlier than the current frame, which means that the coding sequence is different from the playing sequence. Usually, this kind of current frame uses the past I frame or future P frame to predict at the same time, called bidirectional predicted frames, or the B frame. For instance, IBBPBBPBBPBB is an encoding sequence in this encoded mode.

The frames among moving images are not only linear correlation, such as the image prospect being changed and background being motionless but also related considerable movement in the macroscopic view, which means the past frame always comes from the former frame by image translation, scaling, rotation and other motions, such as a camera lens shake. To take full advantage of the movement of the image sequence information and eliminate redundancy, motion compensation techniques must be used to improve video compression efficiency [6]. H.26x and MPEG standards use motion compensation as the encoding method in the interframe. Motion compensation is a method that can use displacement vectors of information and pixels to encode images in dynamic sequence image real-time encoding. In addition, it is a kind of time prediction. A motion compensation block diagram is shown in Fig. 1.

Motion compensation typically includes the following processes [7]:

- 1) Segment moving targets from the image.
- 2) Estimate moving targets.
- 3) Compensate prediction by displacement estimation.
- 4) Encode the prediction information.

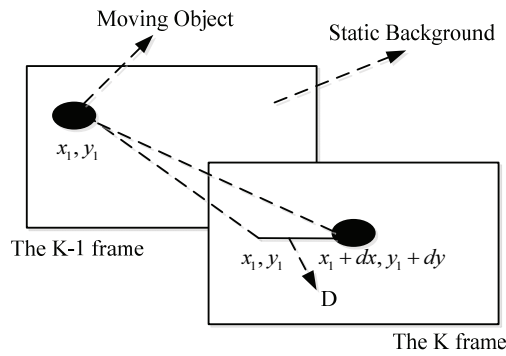


**Fig. 1.** Motion compensation block diagram.

The basic principle is as follows. Assuming that the location of the moving object in the center point of the k-1 frame is  $(x_1, y_1)$ , the location of the moving object in the center point of the K frame is  $(x_1 + dx, y_1 + dy)$ . Shown in Fig. 2, the displacement vector is  $D = (dx, dy)$ . If the difference value between two frames is computed directly, as the center point  $(x_1 + dx, y_1 + dy)$  in the K frame moving object has little relativity with the center point in the K-1 frame moving object, the amplitude of the difference

is large. Additionally, the amplitude of the difference is also very large in the K frame  $(x_1, y_1)$  point (background section) and the K-1 frame corresponding point (moving objects). However, if the displacement vector of moving objects can be motion compensated, which means retracting the moving object in the K frame  $(x_1 + dx, y_1 + dy)$  point back to the  $(x_1, y_1)$  point, and then performing the difference operation with the K-1 frame corresponding points will increase the correlation and reduce the difference signals to increase the compression ratio. Therefore, the displacement vector value of moving objects must be estimated first.

Motion compensation is performed through the previous images to predict and compensate for the partial images. It is an effective method for reducing the frame sequence redundant information. It contains global motion compensation and block motion compensation. In addition, there is variable block motion compensation and block overlapped motion compensation [8].



**Fig. 2.** Interframe displacement of moving objects.

## 2.4 Morphological Processing

Mathematical morphology is based on geometry and analyzes images. The main idea is to use a structural element as a basic tool, which can detect and extract image features, expecting to determine whether this structural element can be put into images inside effectively. The basic operations of mathematical morphology are dilation, erosion, opening and closing [9-11].

### 2.4.1 Dilation and erosion

Dilation and erosion are basic operations in mathematical morphology, and other operations are all combined from these two operations.

#### 1) Dilation operation

“I” is the original image, “S” is the structural element, and the definition of the original image I and structural element S is

$$I \oplus S = \{z \mid (\hat{S})_z \cap I \neq \emptyset\} \quad (1)$$

In the equation, character  $\oplus$  denotes a dilation operation, and  $\hat{S}$  denotes the reflection set of set S.

$$\hat{S} = \{x \mid x = -s, s \in S\} \quad (2)$$

From the above equation, the essence of dilation  $I$  to  $S$  is formed by the set of balance amounts  $Z$ , and the meeting of these balance amounts is that when the reflection set  $S$  shifts  $Z$ , the intersection with set  $I$  cannot be empty. After dilation, there are more image pixels in the current images than in the original image. The dilation operation meets the exchange rate, which is:

$$I \oplus S = S \oplus I \quad (3)$$

In OpenCV, the dilation operation can be implemented by the function `cvDilate`, and the function includes:

```
void cvDilate(const CvArr* src, CvArr* dst, IplConvKernel* element=NULL, int iterations=1);
```

“src” is the input image, “dst” is the output image, “element” is the structural element used by dilation, the default value is NULL, and the size is  $3 \times 3$ . Iterations are the iteration numbers of dilation. The input image and the output image can be the same picture.

## 2) Erosion

“ $I$ ” is the original image, “ $S$ ” is the structural element, and the expression of structural element  $S$  of the original image  $I$  is as follows.

$$I \ominus S = \{z \mid (S)_z \subseteq I\} \quad (4)$$

The symbol  $\ominus$  denotes the erosion operation, and the symbol  $(S)_z$  denotes the set in which set  $S$  shifted  $z$ . The  $S$  is a set of shifted value  $z$  to the erosion  $I$ , and after the shift, these shifted values still are a part of set  $I$ . The resulting image after erosion is slightly smaller than the original image, and the resulting image is a subset of the original set.

In OpenCV, the erosion operation can be implemented by the function `cvErode`.

```
void cvErode (constCvArr* src, CvArr* dst, IplConvKernel* element=NULL, int iterations=1);
```

“src” is the input image, “dst” is the output image, “element” is the structural element used by erosion, the default value is NULL, and the size is  $3 \times 3$ . Iterations is the number of erosion iterations. The input image and the output image can be the same picture.

## 2.4.2 Opening and closing

Opening and closing are two important morphological operations, and they are combination operations of dilation and erosion [12-15]. The opening operation can usually smooth the contours of the images, remove the burr profile to highlight, and cut a narrow valley. Although the closing operation plays the role of smoothing contours, it can remove the hole areas and fill narrow fractures, slender gullies and contour gaps.

### 1) Opening operation

“ $I$ ” is the original image, “ $S$ ” is the structural element, and the expression of structural element  $S$  open to the original image  $I$  is as follows.

$$I \circ S = (I \ominus S) \oplus S \quad (5)$$

The symbol  $\circ$  indicates the opening operation; in other words, the opening operation erodes the original image  $I$  by using structural element  $S$  and then increases the result of erosion. The result of the opening operation is a subset of the original images.

## 2) Closing operation

“I” is the original image, “S” is the structural element, and the expression of structural element S close to the original image I is as follows.

$$I \bullet S = (I \oplus S) \ominus S \quad (6)$$

The symbol  $\bullet$  means closing operation. The closing operation is opposite to the opening operation; it dilates the original images first and then erodes the dilated result. The result of the closing operation is also a subset of the original images.

## 3. Moving Object Contour Extraction Algorithm

After the first few parts of the work, the target contour extraction algorithm is as follows:

- 1) Input the first frame and extract the SIFT feature.
- 2) Input the second frame and extract the SIFT feature.
- 3) Match the SIFT feature between the two frames, output matching pairs.
- 4) Recognition of SIFT feature matching pairs using the SIFT vector field construction algorithm to build.
- 5) Recognition built SIFT vector field, eliminate error classification.
- 6) Recognition of background and target objects by cluster analysis and dispersion analysis algorithms.
- 7) Graying the current frame and previous frame.
- 8) The grayscales of the current frame and previous frame are used for motion compensation.
- 9) After the motion compensation of two grayscales, calculate the difference between the two frames using differential operators.
- 10) Deal the differential image by thresholding, inverted, and morphological processing.
- 11) Extract target contour.
- 12) Determine whether reaching the end of the video sequence. If YES, then end; otherwise, go to 2).

This algorithm is shown in Fig. 3. There is a skill in choosing video frames; if every frame is calculated, there will be a considerable time and space complexity. We attempt to choose video frames every second, every 2 seconds and every 5 seconds. The experiment shows that every 2 seconds of video frame extraction not only has good performance but also has an ideal time and space complexity.

## 4. Experimental Results and Analysis

### 4.1 Target Contour Extraction in Static Background

According to the vector displacement value in SIFT vector field classification, the two original images need motion compensation. The comparison between the original image and motion compensated image is shown in Fig. 4. The rights of the 1st and 2nd frames are eliminated slightly. The locations of the car in the two frames are completely the same.

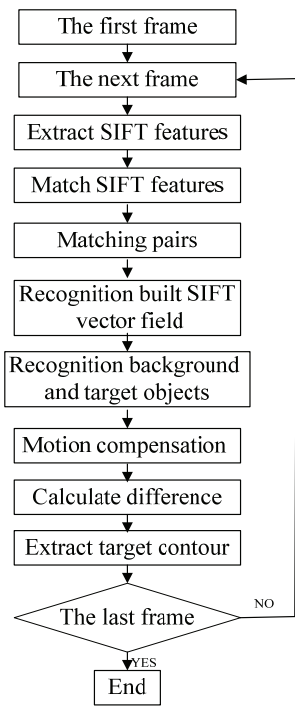


Fig. 3. Moving object extraction algorithm flowchart.

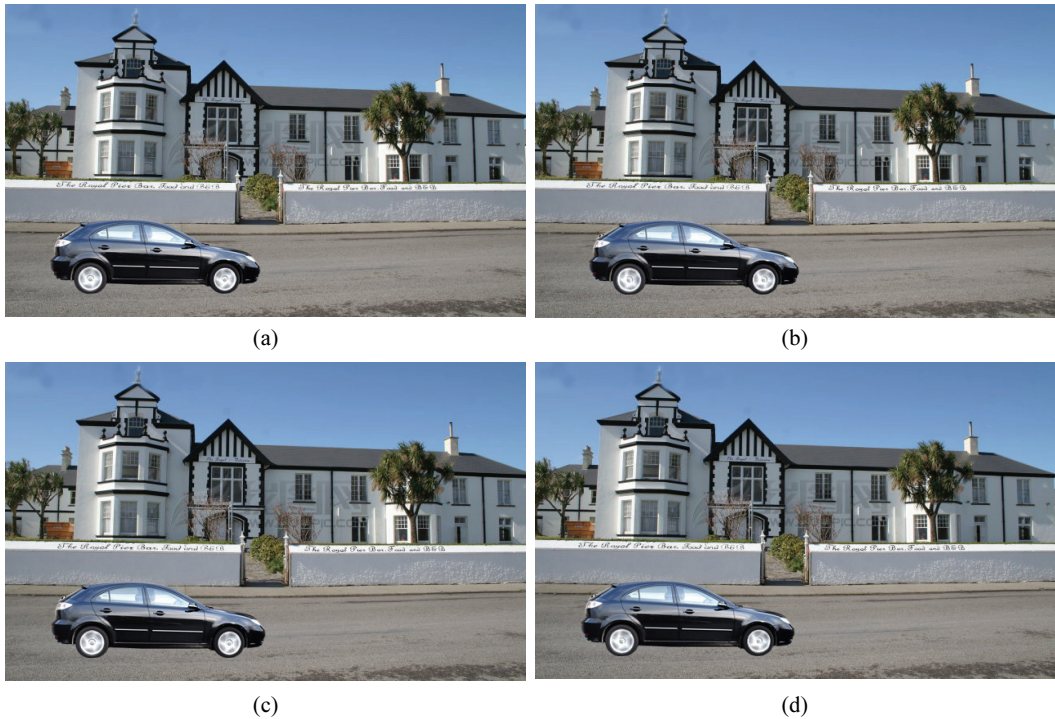


Fig. 4. The original images and compensated images: (a) the 1st frame, (b) 2nd frame, (c) comprehension of the 1st frame, and (d) comprehension of the 2nd frame.

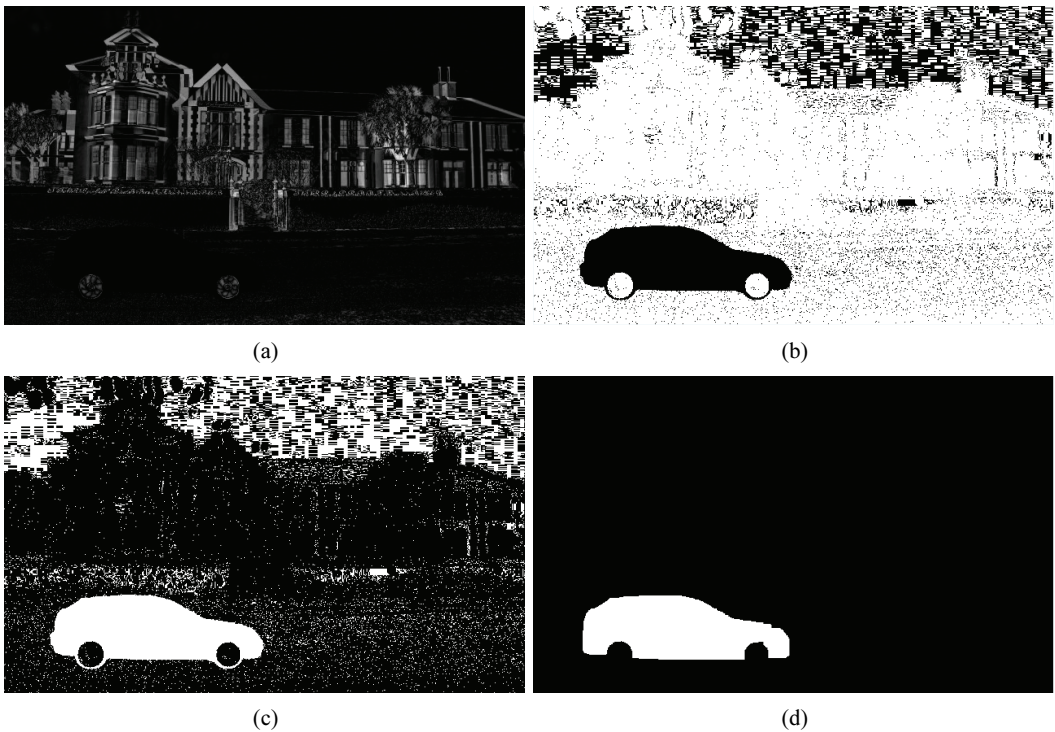


**Fig. 5.** The results of grayscale: (a) the 1st frame and (b) 2nd frame.

The result of the grayscale compensation is shown in Fig. 5.

Next, the results are processed by difference, thresholding, reverse and morphological processing, and the results are shown in Fig. 6.

Finally, the object contour is extracted and displayed on the original image. The results as shown are in Fig. 7. The object in the image is detected, and the contour extraction is also very accurate. However, because of the wheels and body have different moving style, the wheels cannot be detected, and it is also the disadvantage that the algorithm needs to be improved.



**Fig. 6.** Image processing: (a) difference image, (b) thresholding segmentation, (c) opposition, and (d) morphological processing.

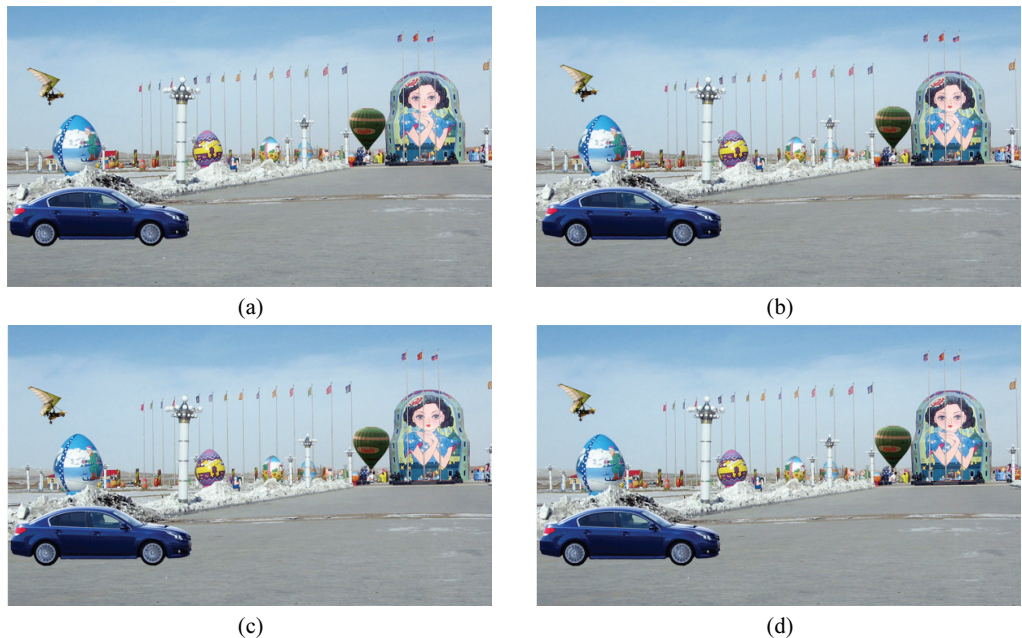


**Fig. 7.** Object extraction result.

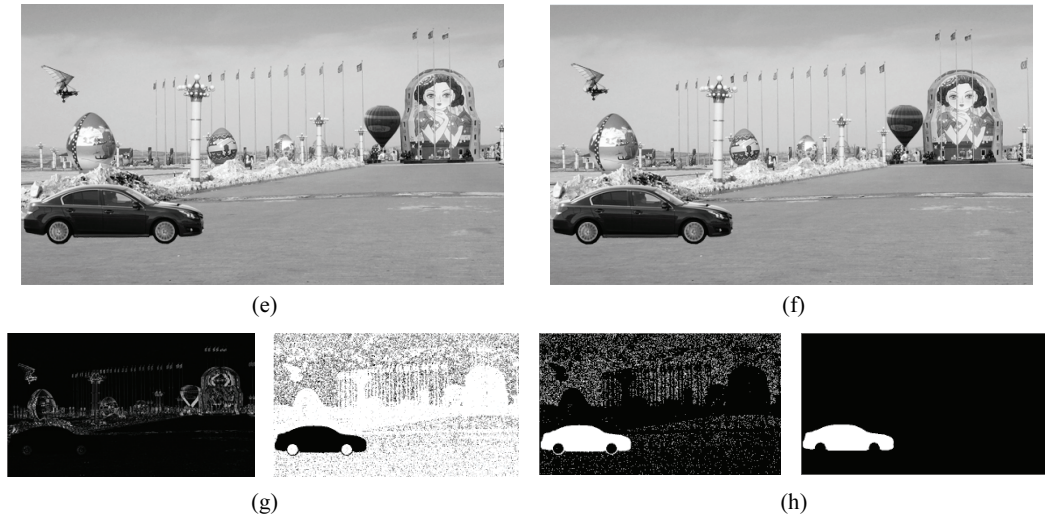
4.2 Target Contour Extraction in Dynamic Background

Object detection in the dynamic background has the same processing order as that detection in the static background. The different point is that in the static background, there is only one motion target, but in the dynamic background, there are three motion targets in each frame. For every target, motion compensation, grayscale, difference, thresholding segmentation, reverse processing, morphological processing and contour extraction are needed. Finally, the whole target outline can be obtained by drawing on the original image using the contour extracted each time.

The whole procedure for the first target extraction between the 1st and 2nd frames is shown in Fig. 8. The extraction method can also be used in the 1st and 2nd frames and the 2nd and 3rd frames. Extraction results are shown in Fig. 9.



**Fig. 8.** The extraction procedure of the first target. (a) The 1st frame and (b) 2nd frame of the original image. (c) The 1st target and (d) 2nd target comprehensive images in the 1st frame. (e) The 1st target and (f) 2nd target grayscale images in the 2nd frame. (g) The difference and thresholding results of the first target in the first frame the second frame. (h) The reverse and morphology processing results of the first target in the first frame the second frame.

**Fig. 8.** (Continued)**Fig. 9.** The extraction procedure between the 2nd frame and the 3rd frame. (a) The contour extraction results of the 2nd target (the balloon) between the 1st frame and the 2nd frame. (b) The contour extraction results of the 3rd target (the glider) between the 1st frame and the 2nd frame.

### 4.3 Experimental Results Analysis

Thus far, the moving objects can be detected frame by frame. The experimental results have good accuracy and efficiency. This algorithm can also be applicable for moving objects in shading and floating. Target object extraction in shading situations is the same as in moving backgrounds. For each target frame, motion compensation, grayscale, difference, thresholding segmentation, reverse processing, morphological processing and contour extraction are needed. Finally, the whole target outline can be obtained by drawing on the original image by using the contour each time extracted.

## 5. Conclusion

The moving object detection algorithm is an important field in computer vision. Although there are many classic algorithms in moving target detection, there are some limitations in these algorithms and many problems in practice. This paper combines the SIFT vector field, clustering and dispersion analysis

and block motion compensation as well as other features, and the effectiveness, stability, accuracy and timeliness of the moving object detection algorithm has been improved. Build the SIFT vector field on the basis of the SIFT feature matching algorithm, which is based on relatively stable local features, rather than simply using the image pixels. Then, cluster and dispersion analyses of the SIFT vector field are performed to separate the background and objectives, providing a general understanding of the situation before image extraction of the target contour, which is better than other algorithms. Moreover, we use the target-based block motion compensation method, which is more flexible than global-based motion compensation and can even decide which one of the target contours needs to be extracted and which one has to give up. Experiments show that this algorithm improves stability and accuracy.

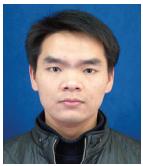
## Acknowledgement

This work was supported by grants from the National Natural Science Foundation of China (No. 41306086 and No. 31700067), the Fundamental Research Funds for the Central Universities (No. HEUCF100606), and the Special Topics on Education Informatization of China Higher Education Federation (No. 2016XXYB06).

## References

- [1] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: fully convolutional one-stage object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Seoul, South Korea, 2019, pp. 9627-9636.
- [2] H. Liu, C. Zhou, J. Shen, P. Li, and S. Zhang, "Video caption detection algorithm based on multiple instance learning," in *Proceedings of 2010 5th International Conference on Internet Computing for Science and Engineering*, Heilongjiang, China, 2010, pp. 20-24.
- [3] X. Liu and W. Wang, "Robustly extracting captions in videos based on stroke-like edges and spatio-temporal analysis," *IEEE Transactions on Multimedia*, vol. 14, no. 2, pp. 482-489, 2011.
- [4] Y. Wang, X. Yang, and C. Zhang, "Research on a kind of remote sensing registration algorithm based on improved SIFT," in *Proceedings of 2016 5th International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, Tianjin, China, 2016, pp. 1-4.
- [5] R. Ji, L. Y. Duan, J. Chen, H. Yao, J. Yuan, Y. Rui, and W. Gao, "Location discriminative vocabulary coding for mobile landmark search," *International Journal of Computer Vision*, vol. 96, no. 3, pp. 290-314, 2012.
- [6] X. Han, Y. Gao, Z. Lu, Z. Zhang, and D. Niu, "Research on moving objects detection algorithm based on improved three frame difference method and optical flow," in *Proceedings of 2015 5th International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, Qinhuaangdao, China, 2015, pp. 580-584.
- [7] X. Ye and S. Wang, "Small object detection algorithm for sonar image based on pixel hierarchy," in *Proceedings of 2015 34th Chinese Control Conference (CCC)*, Hangzhou, China, 2015, pp. 3713-3717.
- [8] J. Guo, H. Zhang, D. Chen, and N. Zhang, "Object detection algorithm based on deformable part models," in *Proceedings of 2014 4th IEEE International Conference on Network Infrastructure and Digital Content*, Beijing, China, 2014, pp. 90-94.
- [9] D. M. Rashed, M. S. Sayed, and M. I. Abdalla, "Improved moving object detection algorithm based on adaptive background subtraction," in *Proceedings of 2013 Second International Japan-Egypt Conference on Electronics, Communications and Computers (JEC-ECC)*, 6th of October City, Egypt, 2013, pp. 29-33.

- [10] C. L. Huang and H. N. Ma, "A moving object detection algorithm for vehicle localization," in *Proceedings of 2012 6th International Conference on Genetic and Evolutionary Computing*, 2012, Kitakyushu, Japan, pp. 376-379.
- [11] H. Gao, Y. Peng, Z. Dai, and F. Xie, "A new detection algorithm of moving objects based on human morphology," in *Proceedings of 2012 8th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Piraeus, Greece, 2012, pp. 411-414.
- [12] S. Chen, X. Li, and L. Zhao, "Multi-source remote sensing image registration based on sift and optimization of local self-similarity mutual information," in *Proceedings of 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Beijing, China, 2016, pp. 2548-2551.
- [13] B. T. Duy and N. Q. Trung, "Speech classification by using binary quantized SIFT features of signal spectrogram images," in *Proceedings of 2016 3rd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS)*, Danang, Vietnam, 2016, pp. 177-182.
- [14] A. Adileh, S. Eyerman, A. Jaleel, and L. Eeckhout, "Mind the power holes: sifting operating points in power-limited heterogeneous multicores," *IEEE Computer Architecture Letters*, vol. 16, no. 1, pp. 56-59, 2017.
- [15] A. Pourreza and K. Kiani, "A partial-duplicate image retrieval method using color-based SIFT," in *Proceedings of 2016 24th Iranian Conference on Electrical Engineering (ICEE)*, 2016, pp. 1410-1415.



**Changjian Zhou** <https://orcid.org/0000-0002-2094-6405>

He received M.S. degree in School of Computer Science and Technology from Harbin Engineering University in 2012. Since March 2012, he is an engineer in Department of Modern Educational Technology, Northeast Agricultural University. His current research interests include machine learning and pattern recognition.



**Jinge Xing** <https://orcid.org/0000-0001-8764-0673>

He received B.S. degree in School of Computer Science and Technology from Harbin Engineering University in 1996. Now he is a senior engineer in Department of Modern Educational Technology, Northeast Agricultural University. His current research interests include machine learning and cyberspace security.



**Haibo Liu** <https://orcid.org/0000-0003-4837-9956>

He received B.S. and Ph.D. degrees in School of Computer Science and Technology from Harbin Engineering University in 1998 and 2005. He is a professor in School of Computer Science and Technology, Harbin Engineering University. His current research interests include computer vision and pattern recognition.