

Research Article

Method of Multiobject Detecting and Tracking Based on DM643

Yitao Liang, Deshan Zhang, Feng Wang, Yonggang Li, and Meng Zhang

School of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China

Correspondence should be addressed to Yitao Liang; liangyt2011@126.com

Received 20 December 2013; Revised 14 March 2014; Accepted 25 March 2014; Published 23 April 2014

Academic Editor: Shuhui Bi

Copyright © 2014 Yitao Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The technology of moving objects detection has become an important research subject for its extensive application prospect. In this paper, it is presented that interframe difference algorithm and background difference algorithm are combined to update the background. The algorithm can deal with the flaw of background difference algorithm. The mathematical morphology method is employed to denoise the image, which may be helpful to improve the accuracy of the detection. The Pyramid algorithm is used to compress each frame data of video sequence. Then, the detecting and tracking of moving objects are tested on the hardware platform (DM643) and the software frame (RF5). The running speed is about 3 times faster than before. The result shows that the accuracy demanded by the detection is met. This method can provide a useful reference for similar application.

1. Introduction

The definition of moving target detection is that the valuable and mobile targets are separated from background [1, 2]. In the field of pattern recognition and computer vision, the technology of moving target detection is a very important topic. It is also widely used in other fields, such as artificial intelligence, security surveillance, human-computer interaction, and intelligent machines; it is a hot topic in recent years.

At present, three existing methods of moving target detection are optical method [3], background difference [4], and interframe difference [5]. Optical method is often used in the condition with unknown information. In this case, the independent movement of the target may be detected. However, this method is time-consuming and difficult to meet real-time detection. Therefore, background difference and interframe difference are usually used instead of this method. It is to provide a relatively complete feature data and obtain accurate target by background difference, but it is sensitive to the changes of external circumstances caused by light. Compared with background difference, the method of interframe difference is insensitive to the changes of external circumstances caused by light. But when there is large gray uniform region of the target surface, holes appeared in

the area of the target so that the target is to be divided into multiple regions.

Interframe difference is simple and it is insensitive to the light scene changes; it can adapt to the dynamic environment and have good stability. Although the algorithm is simple, each pixel of image frame has to be differenced, which may reduce the running speed. In this paper, the Pyramid algorithm is used to compress frame data of video sequence so that the number of pixels of each frame becomes a quarter of the original. Therefore, this algorithm can accelerate the running speed and improve the performance of interframe difference.

The system of moving object detection is employed by using digital signal processor—TMS320DM643 [6] as a hardware platform. The feasibility of the algorithm is tested through the effect of target tracking.

It is to provide a relatively complete feature data and obtain accurate target by background difference, but it is sensitive to the changes of external circumstances caused by light. To solve the faultiness of the algorithm, the background can be updated by combining interframe difference and background difference in this paper. Besides, a great number of false moving targets are eliminated using the method of mathematical morphology [7]. The foreground of moving objects is acquired accurately.

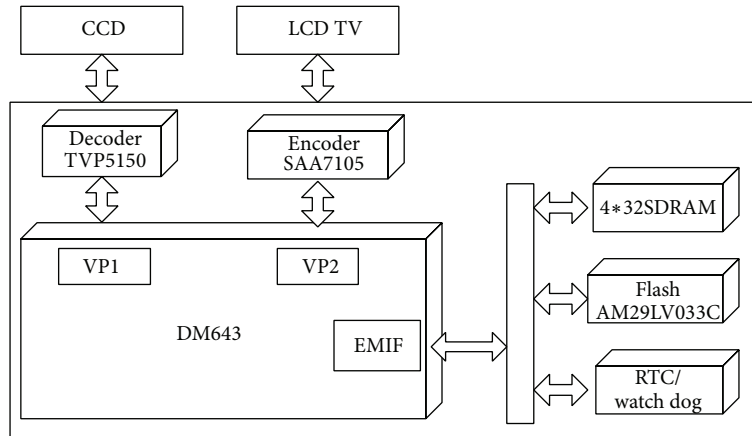


FIGURE 1: Architecture of the system.

2. The Outline of System

2.1. Hardware Platform. The platform contains CCD camera, decoder (TVP5150) [8], video processor (DM643), encoder (SAA7105) [9], LCD monitor, and other peripheral configurations. The architecture of the system is shown in Figure 1.

Real-time video information is captured by CCD camera. The information is transmitted to the decoder (TVP5150) through the coaxial cable. And then, the processor (DM643) [10] implements the function of identifying moving targets. At last, the real-time results after encoding (SAA7105) can be displayed on the LCD monitor.

The processor (DM643) device is based on the second-generation high-performance, advanced VelociTI very-long-instruction-word (VLIW) architecture (VelociTI.2) developed by Texas Instruments (TI), making these DSPs an excellent choice for digital media applications. The DM643 DSP possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. The C64x DSP core processor has 64 general-purpose registers of 32 bit word length and eight highly independent functional units, two multipliers for a 32 bit result, and six arithmetic logic units (ALUs). These performances make DM643 suitable for digital video processing applications particularly. The model of other devices is as follows: camera model: PNT-696; the specification of image acquisition: 99×49.5 cm (39×19.5 in) and monitor model: 7 inch TFT LCD with the resolution: 480×234 ; and the output format: PAL. The system hardware platform is shown in Figure 2.

2.2. The Software Structure. The software flow chart of the system is shown in Figure 3. Firstly, the system performs initialization. It mainly includes CSL and BIOS [11] initialization, setting CACHE to 64 K, mapping CACHE to the CE0 and CE1 space, and setting the DMA priority sequence and length. Secondly, the RF-5 module initialization is to initialize SCOM module that transfers message to the internal unit [12]. Thirdly, the task module is initialized. It is employed to allocate and manage tasks storage space. After that, real-time video image is captured from the input



FIGURE 2: System hardware platform.

device through the TSK-input thread. Here, video sequence is formatted according to 4:2:2 compression interleaved mode by FVID_exchange function [13]. Next, the acquisition-ending message from the input device is sent to the output device; then, when video output task detects the acquisition-ending message, the TSK-process thread that process the video data by the detection algorithm is activated. After processing, the video data are written into the output device by the TSK-output thread, and the FVID_exchange function is started to switch on the display. Finally, the output task sends the ending message to video acquisition tasks, and the system returns a waiting state for the next acquisition-ending message.

The module of data processing task mainly includes five modules (interframe difference, background difference, image denoising, object detecting, and object tracking). The flow chart of data processing task is shown in Figure 4. The aim of interframe difference is to detect moving objects and to provide information to background updating. Background difference module is employed to extract the complete area of objects. Image denoising module is used to remove noise in the binary image by mathematical morphology. And the aim of detecting and tracking object is to mark objects timely.

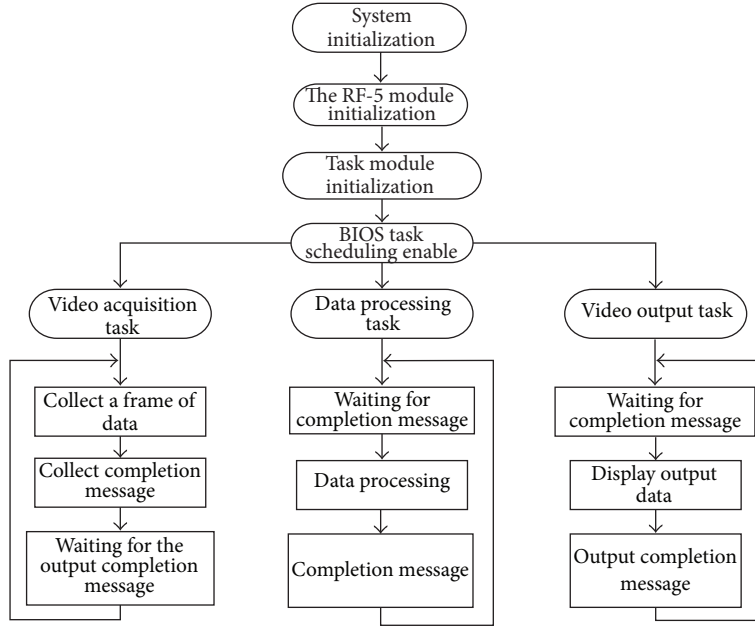


FIGURE 3: Flow chart of software.

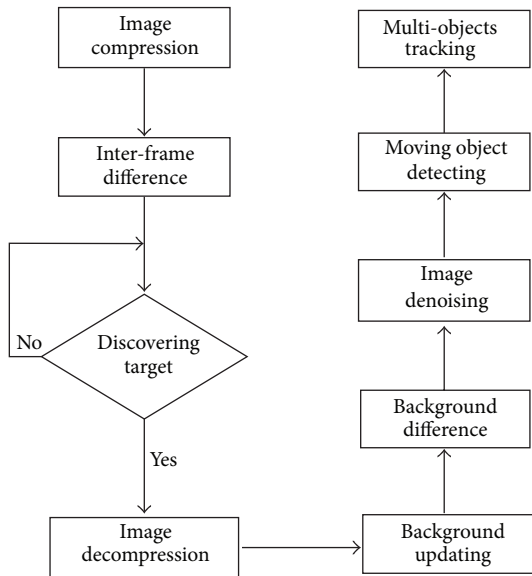


FIGURE 4: Flow chart of data processing task.

3. Improved Interframe Difference

Although the existing interframe difference is simple, each pixel of image frame has to be differenced, which may reduce the running speed. In this paper, the Pyramid algorithm is used to compress each frame data of video sequence so that the number of pixels of each frame becomes a quarter of the original. Therefore, this algorithm can accelerate the running speed and improve the performance of interframe difference.

3.1. The Existing Interframe Difference. The method of interframe difference is mainly to detect and extract target by the difference between two successive frames in the video sequence [14, 15]. The basic process of interframe difference is shown in Figure 5.

The algorithm is explained as follows:

$$D_k(x, y) = |f_k(x, y) - f_{k-1}(x, y)|, \quad (1)$$

$$R_k = \begin{cases} 0 & \text{foreground } D_k(x, y) \geq T \\ 255 & \text{background } D_k(x, y) < T, \end{cases} \quad (2)$$

where $f_{k-1}(x, y)$ and $f_k(x, y)$ indicate the two successive image sequences, x and y are coordinate, k is frame number, $D_k(x, y)$ is the difference image, R_k is the binary image, and T means threshold.

First, (1) is employed to calculate the difference image $D_k(x, y)$ between the k th frame and the $(k-1)$ th frame. And then, according to (2), if a pixel grayscale value in $D_k(x, y)$ is greater than the given threshold T , the pixel belongs to foreground and is set as 0 in the binary image R_k . Otherwise, the pixel is considered as pixel of background pixel and is set as 255 in R_k .

3.2. Interframe Difference Based on Pyramid Algorithm. If all pixels of each frame are differenced, the computational complexity of the algorithm may be high. As it is known, the image always includes a lot of redundant information. So, if a reasonable compressed algorithm is employed and the compressed image instead of the original image is differenced, the computational complexity could be reduced highly.

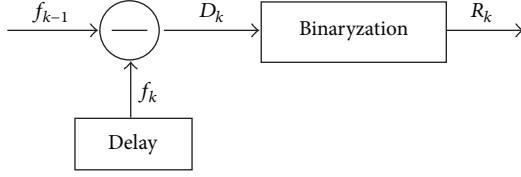


FIGURE 5: Principle diagram of interframe difference.

Here, Pyramid algorithm is used to compress image data. The algorithm is explained as follows:

$$\begin{aligned}
 F_k(x', y') &= \frac{1}{4} \{f_k(x, y) + f_k(x, y+1) \\
 &\quad + f_k(x+1, y) + f_k(x+1, y+1)\}, \\
 D'_K(x', y') &= |F_k(x', y') - F_{k-1}(x', y')|,
 \end{aligned} \tag{3}$$

where pixels in the original image are described by $f_k(x, y)$, $f_k(x, y+1)$, $f_k(x+1, y)$, $f_k(x+1, y+1)$; $F_k(x', y')$ indicates the pixel after compression; and $D'_K(x', y')$ is the difference image. According to the algorithm, an image with the size of $4 * 4$ can be compressed to the size of $2 * 2$.

3.3. Comparison and Analysis. The number of clock cycles required by the above two algorithms during processing a frame image is calculated. In the same condition, it runs about 31198334 cycles after the application of Pyramid algorithm, while the existing method consumes about 100642094 cycles. Obviously, the running speed of interframe difference with Pyramid algorithm is 3 times faster than the existing method.

3.4. Dynamic Threshold Calculation. If the gray threshold in the algorithm of interframe difference and background difference is fixed, the threshold is adjusted instantly by manual intervention; otherwise, the changes of environment or scene could influence the result of moving objects detection. This problem could be solved by OTSU algorithm [16]; it is adopted to determine adaptively the threshold T . The algorithm is based on the idea of the maximum variance between the two clusters. The process of the algorithm is as follows.

First, set an initial threshold t to separate the foreground image from the background image. w_0 means the proportion that pixels of foreground account for the total number of pixels, u_0 is the average gray value of the foreground image, w_1 is the proportion that pixels of background account for the total number of pixels, u_1 is the average gray value of the background image, u is the total average gray value of

the image, and V_t is the variance between the two clusters. Consider

$$\begin{aligned}
 w_0 &= \frac{1}{MN} \sum_{f(i,j) < t} 1, \\
 u_0 &= \left(\sum_{f(i,j)=1} 1 + 2 \times \sum_{f(i,j)=2} 1 + \dots + (t-1) \right. \\
 &\quad \times \left. \sum_{f(i,j)=t-1} 1 \right) \left(\sum_{f(i,j) < t} 1 \right)^{-1}, \\
 w_1 &= 1 - w_0, \\
 u_1 &= \left(t \times \sum_{f(i,j)=t} 1 + (t+1) \times \sum_{f(i,j)=t+1} 1 + \dots + m \right. \\
 &\quad \times \left. \sum_{f(i,j)=m} 1 \right) \left(\sum_{f(i,j) \geq t} 1 \right)^{-1}, \\
 u &= w_0 \times u_0 + w_1 \times u_1, \\
 V_t &= w_0(u_0 - u)^2 + w_1(u_1 - u)^2, \\
 T &= \arg \max_{0 \leq t \leq m} V_t,
 \end{aligned} \tag{4}$$

where $f(i, j)$ is image sequences having the size $(M * N)$, i and j are coordinate, t is a threshold, $(t = 0, 1, \dots, m)$, m is the maximum value of grayscale, and T is the obtained dynamic threshold.

4. Improved Background Difference

In order to meet the demand of real-time processing of video monitoring system, in the paper, it is presented that interframe difference algorithm and background difference algorithm are combined to update the background, and an improved background difference is used to extract motion region, which could deal with the problem that background cannot be automatically updated with dynamic scene. The algorithm mainly includes three steps: background extraction, motion region extraction, and background updating which are as follows.

- (1) *Background Extraction.* The initial background image is extracted by calculating the pixel average of the first 100 frames from video sequences. Let the initial background be $f_b(x, y, t)$, where x and y are coordinate, and t is frame number as shown in

$$f_b(x, y, t) = \frac{1}{100} \sum_{i=1}^{100} f(x, y, i), \tag{5}$$

where $f(x, y, i)$ is the video sequences from the first 100 frames and i is the frame number, $(i = 1, 2, \dots, 100)$.

- (2) *Motion Region Extraction.* The difference between every frame and background is conducted to obtain motion region in current frame after background extraction. Motion region extraction includes two steps: making difference and difference image binarization. The output is a binarization image whose grayscale value is 255 (indicating background area) or 0 (indicating motion region). Let $f_b(x, y, t)$ be the background image; the change measure can be written as

$$R(x, y, t) = \begin{cases} 0 & \text{if } |f(x, y, t+1) - f_b(x, y, t)| > T \\ 255 & \text{otherwise,} \end{cases} \quad (6)$$

where $f(x, y, t+1)$ is frame $t+1$ and OTSU algorithm is adopted to determine adaptively the threshold T .

- (3) *Background Updating.* First, the binary image of foreground image $ff(x, y, t+1)$ is acquired by subtracting the current frame $f(x, y, t)$ from the previous frame $f(x, y, t+1)$ based on interframe difference; secondly, let $k(x, y)$ be the pixel of $ff(x, y, t+1)$. If the value of $k(x, y)$ is 0, background should be updated; otherwise, background keeps the original value. Variable a is the updating speed. Background updating could be expressed as

$$\begin{aligned} f_b(x, y, t+1) &= \begin{cases} f_b(x, y, t) & R_i = 255 \\ af_b(x, y, t) + (1-a)f(x, y, t+1) & R_i = 0, \end{cases} \\ a &= \frac{|f(x, y, t+1) - f_b(x, y, t)|}{\text{Max}(f(x, y, t+1), f_b(x, y, t))}. \end{aligned} \quad (7)$$

- (4) If the value of all pixels in the binary image of foreground image is 0, the program ends; otherwise, the updated background is assigned to $f_b(x, y, t)$. Go back to step (2).

5. Image Denoising

For the influence of noise, some pixels which belong to background could be clustered for pixels of the foreground. Contrarily, some pixels which belong to foreground are mistaken for pixels of the background image. At the same time, because of the slight flutter of objects, some pixels of background image could be falsely identified as pixels of moving target. In order to fight these harmful effects, binary image should be disposed so that motion target area is detected. The mathematical morphology method [17] is selected to denoise in the binary image in this paper.

Mathematical morphology analyses image based on the morphology of image region. The main purpose of using the method is to obtain the object topology and structure information by interacting objects and the structural elements.

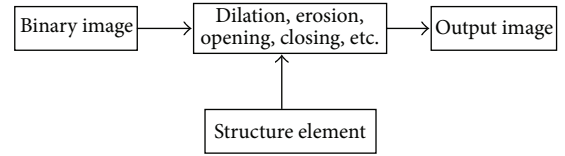


FIGURE 6: Base idea of mathematical morphology.

The basic operations of mathematical morphology contain dilation, erosion, opening, and closing. Its basic idea is shown in Figure 6.

In this paper, the denoising of binary image is reached using the erosion operation, where the structure element is 6×6 . The effects of the method are shown in Figure 7.

6. Multiobject Detecting and Tracking

After denoising, targets should be tracked. There are many kinds of methods to detect and track targets. And then the proposed method and the other existing methods are introduced.

6.1. The Existing Methods. The existing method of detecting and tracking targets is as follows.

- (1) Scan the binary image line by line. If pixels that belong to the same connected region are found by the connectivity analysis algorithm, they are marked as the same number, while in contrast, they are marked as different number.
- (2) Find the maximum and minimum value of x -coordinate and y -coordinate in each target region marked as the same number.
- (3) According to the maximum and minimum value of x -coordinate, the top and bottom positions of a frame are determined. According to the maximum and minimum value of y -coordinate, the left and right positions of the frame are determined. Targets are marked by this frame.

6.2. The Proposed Method. Pixels of the binary image are scanned line by line for targets. If one pixel of targets is discovered, the target is marked with a fixed frame. The concrete realization method is as follows.

- (1) Scan the binary image line by line. If the value of a pixel is 1, this pixel belongs to target.
- (2) This pixel is as a reference point. A fixed frame whose size is $100(\text{pixel}) \times 50(\text{pixel})$ will mark the pixel, and then the pixel is as a starting point, beating 40 pixels as the next scanning point.
- (3) Repeat steps (1) and (2) until the binary image is scanned.

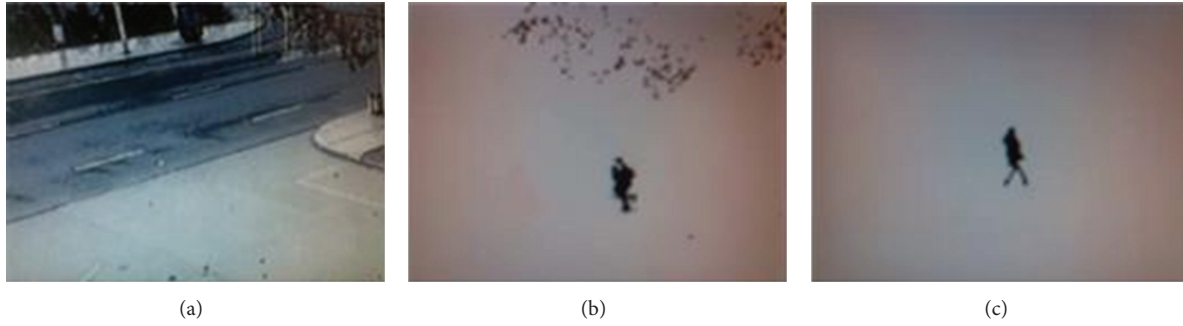


FIGURE 7: Effect of before and after denoising, where (a) is the real background image, (b) is the binary image before denoising, and (c) is the binary image after denoising.



FIGURE 8: Original image sequence of moving target.

6.3. Comparison and Analysis. The proposed method has three advantages compared with the existing methods as follows.

- (1) *Denoising.* This method only uses the erosion operation in mathematical morphology method to denoising, which is easy to realize.
- (2) *Marking Targets.* The connectivity analysis algorithm is left out. The target is marked based on individual pixels instead of the whole region of some target.
- (3) *Real-Time Performance.* Because these algorithms in this method are easier and less than ones in the existing methods, the system with this method has a good real-time performance.

7. Results and Analysis

The results of detecting and tracking objects are shown in Figures 8 and 9. The original image sequence included moving targets as shown in Figure 8. As shown in Figure 9, moving targets are detected and tracked in real-time way. It shows that the algorithm presented in this paper is feasible in the schedule of multiobject detecting and tracking.

While the system is operating, the thread scheduling analyzation result of CPU (DM643) task running sequences is presented in Figure 10. Each task execution process mainly refers to three running states (“not ready”, “ready”, and “running”). The different color indicates different states in

Figure 10. White color means “not ready” state, light blue means “ready” state, and blue means “running”. KNL-swi automatically created by DSP/BIOS is a kind of software interrupt. It is used to initialize the task scheduler and the program [18]. KNL-swi has the highest priority. Before the HWI event is activated, it must be executed. The program includes three threads (“tskOutput”, “tskInput”, and “tskVideoProcess”). The priority of “tskOutput” thread is the highest in the three threads. The priority of “tskInput” thread is the next. The “tskVideoProcess” has the lowest priority.

8. Conclusion

The results of the study basically have the following six aspects.

- (1) The processor (DM643) is chosen as a platform. This processor with VLIW architecture is developed by the VelociTI technology, which is suitable for digital media applications.
- (2) The video processing algorithm based on RF5 is implemented. System tasks are scheduled by RF5 framework automatically. This framework makes three threads (TSK-input, TSK-output, and TSK-process) synchronize and transfer message between each other through SCOM message queue. It makes three threads more conveniently and efficiently.
- (3) Interframe difference algorithm with Pyramid algorithm makes detecting and tracking of moving target

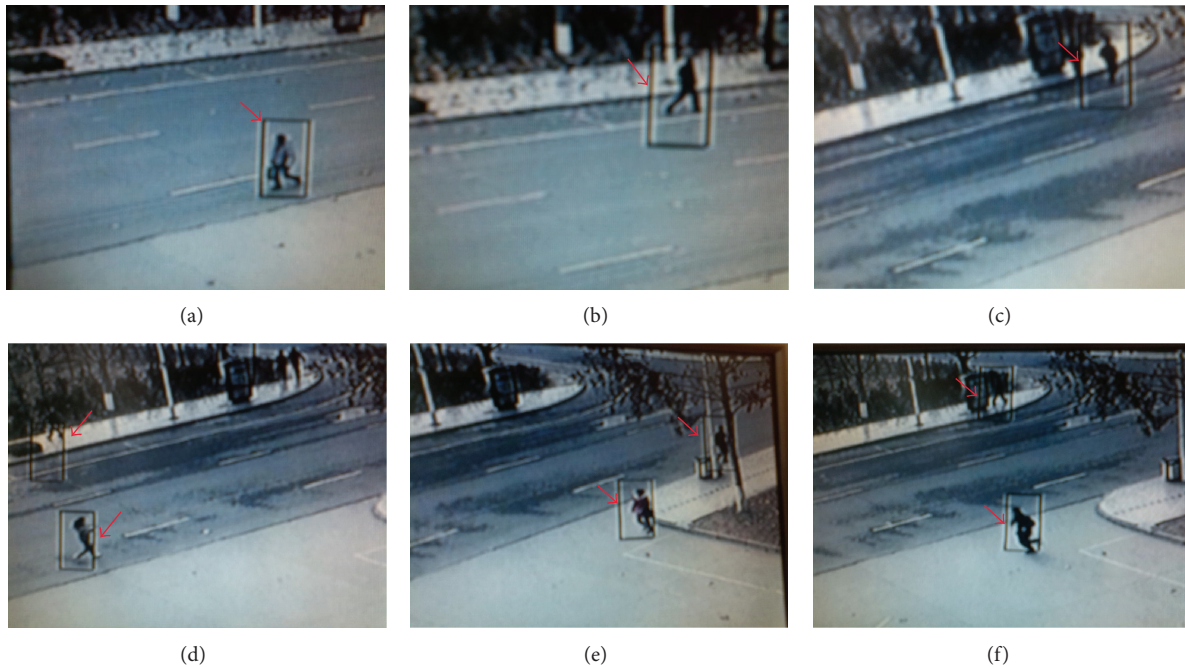


FIGURE 9: Results of detecting and tracking targets.

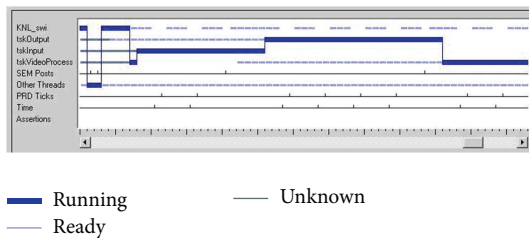


FIGURE 10: Thread scheduling analysis of the running sequence.

be lower computational complexity. It allows the system to meet the application demands of the dynamic environment and has a high stability.

- (4) Improved background difference algorithm is implemented by combining interframe difference with background difference. It could be employed to update background and overcome the shortcoming that existed background difference is sensitive to environmental change.
- (5) Noise in the binary image of foreground is removed with the algorithm of mathematical morphology. This algorithm is simple and fast. It is helpful to the real-time performance of the system.
- (6) Multiple targets are marked by fixed frame. Although the fixed frame cannot change accordingly with the change of the size of target, this method has great advantages in real-time.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is sponsored by the National Natural Science Foundation of China (31171775, 11201120, 61201389, and 61174056) and the National High Technology Research and Development Program of China (2012AA101608). Research project was supported by the Key Laboratory of Grain Information Processing and Control.

References

- [1] Z. Yujin, "Motion analysis," in *Book of Image Engineering (II) Image Analysis*, chapter 13, pp. 349–365, 2005.
- [2] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Computer Vision and Image Understanding*, vol. 80, no. 1, pp. 42–56, 2000.
- [3] A. G. Borş and I. Pitas, "Prediction and tracking of moving objects in image sequences," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1441–1445, 2000.
- [4] S. Fang, F.-Z. Xue, and X.-H. Xu, "Moving object detection based on self-adaptive background modeling," *Journal of System Simulation*, vol. 17, no. 1, pp. 159–165, 2005.
- [5] L.-L. Li, Z.-Q. Ma, and X.-Y. Zhang, "Prospects and current studies on moving object detection technology," *Journal of Nanyang Normal University*, vol. 18, no. 9, pp. 79–82, 2009.
- [6] Texas Instrument Incorporated, "The TMS320DM643 Video Port Mini-Driver," 2003.

- [7] L.-L. Chen and D.-Y. Bi, "Application of mathematical morphology in image processing," *Modern Electronic Technique*, no. 8, pp. 18–20, 2002.
- [8] Texas Instrument Incorporated, "TVP5150A/TVP5150AMI Data Manual," 2004.
- [9] Philips, "Digital Video Encoder SAA7120H-121H Data Manual," 2002.
- [10] Texas Instrument Incorporated, *DEC643 User's Guide and Data Sheet*, 2005.
- [11] H. Wei, C. Bin, and Z. Ling, "Application of DSP/BIOS in video processing based on DM642," *Information and Electronic Engineering*, vol. 4, no. 1, pp. 60–62, 2006.
- [12] S. Qun, Z. Ying-Xin, T. You-chuan, and L. Dan, "TMS320DSP algorithm standard and the reference frameworks level 5 for TMS320DSP," *Ship Electronic Engineering*, no. 2, pp. 27–31, 2006.
- [13] Z. Bo, W. Yu-Bin, and D. Hua, "Video driver of DSP based on GIO/FVID," *Microcontrollers & Embedded Systems*, no. 8, pp. 27–29, 2006.
- [14] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.
- [15] I. Haritaoglu, D. Harwood, and L. S. Davis, "W⁴: real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, 2000.
- [16] J.-M. Liang and Z.-X. Wei, "Research on improvement of the ostu algorithm," *Journal of Sichuan University of Science & Engineering*, vol. 23, no. 5, pp. 543–545, 2010.
- [17] Q.-Y. Dai and Y.-L. Yu, "The advanced of mathematical morphology in imageprocessing," *Control Theory and Application*, vol. 18, no. 4, pp. 479–481, 2001.
- [18] H.-F. Zhang, Q.-C. Pang, Y.-N. Duan, and H. Li, "Application of DSP/BIOS in dual-spectrum digital image fusion," *Control and Instruments in Chemical Industry*, vol. 33, no. 4, pp. 40–43, 2006.

