Optimization of Three - Frame Difference Method and Improvement of Pedestrian Detection Code Book

Lei Shang¹, Fucheng You¹, Chao Han¹, Xuewei Wang¹, and Shuai Zhao¹

¹Beijing Institute of Graphic Communication. No. 1, Xinghua Street (two sections), Daxing District, Beijing
Email: 820596518@qq.com

Abstract. Pedestrian detection is one of the most important applications of computer vision. In this paper, aiming at the problem of poor anti-interference ability of traditional codebook algorithm in the complex background and the cavity-prone problem of traditional frame difference method, a pedestrian detection algorithm integrating the optimized frame difference method and improved codebook is proposed. In this algorithm, a double code word structure is designed. An update mechanism is added to the code words in the codebook background modeling process and detection process, and improved double code word book algorithm based on YUV color space is designed to realize the real-time background update function. After that, bilateral filtering is added to the traditional three-frame difference method to further remove the noise. At the same time, the extracted rectangular region is used for filling operation and morphological processing to obtain the corresponding foreground image. Finally, the foreground image of codebook model and the foreground image of frame difference method are calculated with "and", "different or" and "or" of pixels to obtain the precise region of moving target. The experimental results show that this algorithm combines the advantages of the two methods and abandons the disadvantages of each method in the detection prospect. It can accurately detect moving targets and has good robustness. The effectiveness of this algorithm is also verified by experiments.

1. Introduction
Moving target detection refers to the extraction of moving targets from complex video images, that is, the research of foreground detection and background detection. For moving targets algorithms, there are mainly several methods as follows:

1.1. Frame Difference
This method is mainly to analyze the differences between adjacent frames in image sequence, and then get difference results [1], this algorithm is relatively simple, and low complexity, related factors such as lighting will not bring impact to extract the moving object. The limitations of this algorithm are mainly reflected in the fact that it is usually impossible to obtain complete edge information when the target moves while extracting moving objects. When the background color is similar to the target color, the information in the detected moving target is often considered as the background information, which makes it difficult to accurately extract the target.

1.2. Background Differencing
This method is to build a corresponding background model for each pixel in the frame sequence of video, and then make a comparative analysis between the current pixel and the model, judge whether the pixel belongs to the background[2]. This algorithm has strong anti-interference and strong
applicability, and can be applied to both complex and simple backgrounds [3]. Real-time update of background image can improve detection accuracy.

1.3. Optical Flow
For a moving object, changes in its state will constitute a corresponding series of images, which can be simply regarded as optical flow. By analyzing these image sequences and location information, it is possible to identify whether there are moving objects in the image. In a certain period of time, moving objects in a series of images can present optical flow properties. Then, the moving target can be detected by the optical flow field estimation of vector displacement, which is the optical flow method. For the calculation of optical flow field, scholar Sidenbladh introduced support vector machine (SVM) technology to extract moving targets, and experiments showed that this method could significantly improve the detection effect of moving targets. In addition, Okada [4] et al. introduced digital signal processing technology, which improved the speed of motion detection in some extent.

1.4. A Variety of Moving Target Detection Algorithm Combined

Background subtraction method, optical flow method and background modeling method can combine multiple moving target detection algorithms to realize moving target detection in complex background [5].

2. Traditional Codebook Models and Yuv Color Spaces

2.1. Traditional Codebook Model
With codebook algorithm [6], it is necessary to construct the corresponding codebook for video sequence images, that is \( \{c_1, c_2, c_3, ..., c_L\} \), its length can be set to L. In the RGB color space, each code word: \( \{c_i; i \in \{1, 2, 3, ..., L\}\} \), it's all given by the RGB vector, which is \( V_i = (R, G, B) \) and Six parameters constitute. \( f_j \) Represents the frequency of code words being accessed, \( p_j, q_j \) is the first and last accessed time of the code word, and the code word in the training stage is the maximum time interval when it is not re-accessed, represent by \( \hat{\lambda}_i \).

The specific algorithm flow of building this model is given below:

- Build the codebook, where is an empty set, where the code word L value is zero.
- Set the time \( t = 1 \) to \( N \), where:

\[
X_t = (R, G, B)
\]

Retrieve whether \( C_m \) exists in codebook \( \phi = \{c_i | 1 \leq i \leq L\} \) and meet the following conditions:

\[
\begin{align*}
\text{Colordist}(X_t, V_m) & \leq \varepsilon, \\
\text{Brightness}(1, \langle t^\text{min}_m, t^\text{max}_m \rangle) & = \text{true}
\end{align*}
\]

Assuming that \( \phi \) is a null value, or no matching value is found, and the number of code words needs to be increased by 1, then initialize it according to the following rules, and obtain the corresponding new code word, namely \( c_L \), so:

\[
V_L = (R, G, B)
\]

\[
nX_L = < I, I, 1, t - 1, t, t >
\]

Otherwise, the current matching code word, namely \( C_m \), will be updated as follows:
\[
V_m = \left( \frac{f_m R_m + R}{f_m + 1}, \frac{f_m G_m + G}{f_m + 1}, \frac{f_m B_m + B}{f_m + 1} \right)
\]

\[
uX_m = \min \left\{ I, \frac{I_m}{m}, \frac{I}{m} \right\}, \max \left\{ I, \frac{I_m}{m}, \frac{I}{m} \right\}, m + 1, \max \left\{ \lambda_t, t - q \right\}, p, t \right) \quad (5)
\]

- For each code word in it, which is \( C_i, i = 1, \ldots, L \), loop to make sure that
  \[
  \lambda = \max \left\{ \lambda_t, N - q_i + p_i - 1 \right\}
  \]
  holds.

From the training link to the completion of codebook model construction, a time threshold, namely \( T_u \), can be set as 50.0% of the training time. Then, combined with formula (2-5), the model \( U \) after filtering the redundant codebook can be obtained, and the specific formula is as follows:

\[
U = \left\{ c_m \in \phi \wedge \lambda_c \leq T_m \right\} \quad (6)
\]

### 2.2. Defects in the Traditional Codebook Model
- The code word composed of 9 elements is tedious and the detection speed is slow.
- In the process of learning background of the first few decades \([7]\), parameters need to be set manually, and the setting of these parameters in practical application needs many attempts and experience accumulation.
- The most important point is that RGB color space cannot completely separate brightness and chromaticity, and the change of brightness will cause the change of chromaticity. Under the condition of equal chromaticity error, the pixel with lower brightness value is easier to judge as background than the pixel with higher brightness value. In order to solve the problems existing in the traditional codebook model, this paper introduces the codebook model into the YUV space with better discrimination.

### 2.3. YUV Color Space
RGB color space is composed of red, green and blue primary colors, which can be superimposed into any color. It is a common color space. YUV color space is characterized by the separation of brightness and chromaticity information, where the \( Y \) component represents the brightness of color, and \( U \) and \( V \) determine the chromaticity of color. In YUV color space, the value range of \( Y \) is \([0,1]\), and the range of \( U \) and \( V \) components is \([-0.5, 0.5]\).

The conversion formula between RGB space and YUV space is as follows:

\[
\begin{bmatrix}
Y \\
U \\
V
\end{bmatrix} =
\begin{bmatrix}
0.298 & 0.589 & 0.116 \\
-0.14 & -0.28 & 0.445 \\
0.527 & -0.52 & -0.085
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
\]

(7)

### 2.4. C-YUV Algorithm and Variable Definition
The codebook model in this paper is both an improvement on the traditional codebook model and the literature. In this paper, the codebook model is established in YUV space by the following methods. Firstly, the code word of the 9 attribute elements in the traditional code book model is reduced to only 6 attribute elements \( < Y_i, U_i, V_i, f_i, \lambda_i, q_i > \), where \( Y_i \) represents the brightness value of the code word corresponding to the pixel, \( U_i \) and \( V_i \) represent the chromaticity value of the code word corresponding to the pixel, and the meaning of other elements is the same as that in the traditional code book model \([8]\). This streamlining reduces storage and computation.
Color distortion $\text{colordist}(x_t, c_i)$ is calculated by formula (8) through $(\overline{U}_i, \overline{V}_i)$ in the current pixel $x_t = (Y, U, V)$ and code word $c_i$:

$$
\text{colordist}(x_t, c_i) = \sqrt{(U - \overline{U}_i)^2 + (V - \overline{V}_i)^2}
$$

Brightness distortion $\text{brightness}(x_t, c_i)$ is still represented by $\text{Brightness}(I, < I_{m}^{\min}, I_{m}^{\max} >= \text{true})$, but the upper and lower limits of its variation range are redefined as: $I_{m}^{\min} = (1 - \varepsilon_y)Y_i, I_{m}^{\max} = (1 + \varepsilon_y)Y_i$, where $\varepsilon_y$ is the threshold to control the brightness range. The calculation formula of the new code book model is simple and the process is simplified.

### 2.5. C-YUV Model Training

First read the video stream, to convert each frame image from RGB space to the YUV space, began training first n frames, and then determine whether the pixel code book is empty, if is to create a new code word $L = L + 1$, if not the brightness and colorist calculation, and determine whether individually and existing code word matching, last updated code word, the next frame repeat this process.

### 2.6. C-YUV Foreground Detection

After the codebook model is trained, the image to be detected is compared with the codebook model trained before. When the video image is read in, if the corresponding pixel point is successfully matched, it will be the background point; if not, it will be the foreground target [9].

### 2.7. C-YUV Code Word Update

If the match is successful, the code word that matches successfully will be updated. Formula (9) is as follows:

$$
c_i = \frac{f_i \overline{Y}_i + Y}{f_i + 1}, \frac{f_i \overline{U}_i + U}{f_i + 1}, \frac{f_i \overline{V}_i + V}{f_i + 1}, f_i + 1, \max(\lambda_i, t - q_i), t >
$$

### 3. Moving Target Detection by Fusing Three Frame Difference Method

#### 3.1. Principle of Optimizing Three-Frame Difference Method

The three-frame difference method is to continuously shoot $f_{k-1}(x, y), f_k(x, y), f_{k+1}(x, y)$ three-frame images. In other words, $D1(x, y)$ can be obtained by performing difference calculation on k-1 and k frames, and $D2(x, y)$ can be obtained by performing difference calculation on k and k + 1 frames.

$$
\begin{align*}
D1(x, y) &= |f_k(x, y) - f_{k-1}(x, y)| \\
D2(x, y) &= |f_{k+1}(x, y) - f_k(x, y)|
\end{align*}
$$

Then the difference result is binarized.

$$
\begin{align*}
R1(x, y) &= \begin{cases} 1, & D1(x, y) \geq T \\
0, & D1(x, y) < T \end{cases} \\
R2(x, y) &= \begin{cases} 1, & D2(x, y) \geq T \\
0, & D2(x, y) < T \end{cases}
\end{align*}
$$
The resulting binary image is then performed with an "and" operation to remove the elongated portion of the moving object. In the following formula, \( \& \) stands for "and" operation.

\[
R(x, y) = R1(x, y) \& R2(x, y)
\]  

Then carry out "xor" operation, this step is used to get the difference part of two images of the same frame, \( \wedge \) stands for "xor" operation.

\[
L(x, y) = R1(x, y) \wedge R2(x, y)
\]

The specific implementation steps of the improved three-frame difference method are as follows:

- First, obtain three consecutive frames of Image1, Image2 and Image3.
- Bilateral filtering, median filtering and grayscale processing are respectively performed on the image to take out the moving region and use three-frame difference to carry out and operation.
- In the selected motion region, a minimum rectangular region can be found to contain the motion region \((lx, ly, width, height)\). \(lx\) represents the x-coordinate of the upper-left corner of the rectangular region, \(ly\) represents the y-coordinate of the upper-left corner of the rectangular region, \(width\) represents the width of the rectangular region, and \(height\) represents the height of the rectangular region.
- Extract the Canny edge of Image2 to get the Image2_canny image, and do a morphological closed operation. For the resulting Image2_canny image, extract a small rectangular box from the rectangular area in the previous step. After that, the closed contour composed of the inner edge of the rectangular region was extracted and filled with white.
- Median filtered image Image2_canny, get the final result image Image2_canny_end.

The processing flow is shown in figure 1.

![Figure 1. Flow chart of optimized three-frame difference method](image-url)

The optimal frame difference method can effectively acquire moving objects by difference calculation of adjacent frame images and contrast, which can effectively overcome the stretching and expansion of moving objects and reduce the void phenomenon.
3.2 Improved Codebook Model Fusion Principles

The region obtained by the optimized three-frame difference method and the improved codebook model was "or", and the foreground $C_k(x, y)$ obtained by the codebook model and the foreground $T_k(x, y)$ obtained by the three-frame difference method were calculated as follows:

$$ F_k(x, y) = C_k(x, y) \oplus T_k(x, y) $$ (14)

3.3. Background Update

In order to adapt to the change of the target motion state, the background model should also change with the scene. The change of scene mainly comes from that the moving target in the foreground stays in the background. At this time, the stationary target should be integrated into the background. In order to update the background model, the following method is applied: save the initial codebook in structure $cC$, while structure $cD$ also saves an identical codebook and corresponding parameters $cD_H$, $cD_ADD$, and $cD_DELETE$. If the sample value of a pixel matches $cC$, update $cC$; otherwise, create a new code word for $cD$ and filter according to $cD_H$. If the code word stays longer than $cD_ADD$, the code word is added to $cC$. If the code word in $cC$ has not been matched for more than $cD_DELETE$ times, the code word is deleted from $cC$. The specific algorithm is as follows:

- After the initial training, the background code is $cC$; create a new codebook $cD$.
- Corresponding to the current pixel $x=(R, G, B)$, if a matching code word is found in $cC$, the corresponding code word will be updated according to formula (20). Otherwise, search in $cD$. If found, update the corresponding code word according to formula (20); Otherwise create a new code word $h$ for $cD$.
- Filter $cD$ according to $cD_H$.
- Move the code word with a long enough time in $cD$ to $cC$, i.e.
  $$ cC = cC + \{ h_i \mid h_i \in cD, \lambda_i>cD_H \} $$ (16)
- Delete the code word that has not appeared for a long time from $cC$, i.e.
  $$ cC = cC - \{ c_i \mid c_i \in cC, \lambda_i>cD_DELETE \} $$ (18)

3.4. Fusion Codebook Implementation

The research on the optimized three-frame difference method and the improved codebook pedestrian detection proposed in this paper mainly includes five steps:

- Foreground images are obtained through the optimized three-frame difference method;
- Background modeling and foreground detection were conducted with the improved codebook model, and foreground images were obtained;
- Perform "and", "different or" or "operation on the two foreground images obtained in step (1) and step (2) to obtain the best foreground image;
- Background update is carried out according to changes in the motion state of the detected target.
In order to remove the slender part of the moving object and reduce the void effect, morphological processing is carried out to obtain more accurate moving targets.

4. Experimental Results and Analysis
In this experiment, Intel(R) Core(TM) i5-4590 CPU @3.30ghz, 8 GB of memory and Windows 7 operating system were used. This paper takes Visual Studio 2013 as the experimental basis and USES OpenCV3.4.0 library function to optimize and improve the algorithm. Two groups of experimental results were used as experimental data. The experimental materials were two different views, one was the view of pedestrians in the meeting room, and the other was the video in public places. The traditional codebook algorithm, the traditional three-frame difference algorithm, the MOG algorithm and the improved algorithm in this paper were compared.

In this group of experiments, the real-time updated codebook algorithm incorporating the three-frame difference method was compared with the algorithm without real-time update. The codebook algorithm results in less noise and better effect. At the same time, compared with other relevant algorithms, the results are the same.

The results of the second group of experiments are shown in figure 2, figure 3, figure 4, figure 5 and figure 6:

5. Conclusion
The noise of the improved codebook algorithm is obviously reduced, and real-time update brings stronger robustness. In addition, the combination of codebook and frame difference method further
strengthens the robustness of illumination and improves the overall performance, thus enhancing the practicability of the algorithm.

6. Acknowledgment
This work was partially supported by Joint Funding Project of Beijing Municipal Commission of Education and Beijing Natural Science Fund Committee (KZ201710015010), Project of National Scientific Found (No.61370188), Project of Beijing Municipal College Improvement Plan (PXM2017_014223_000063) and New Project of Green Printing and BIGC Project (Ec201803 Ed201802 Ea201806).

7. References
[1] Yin J, Liu L, Liu Q. The infrared moving object detection and security detection related algorithms based on W4 and frame difference[J]. Infrared Physics & Technology, 2016 (7): 302 - 315.
[2] Yinghui Zhang, Yangshuo Liu. Moving target detection based on frame difference method and background difference method[J]. Computer technology and development, 2017, 27(2): 25 - 28.
[3] Ting Sun, Yingchun Qi, Guohua Geng. Moving target detection algorithm based on inter-frame difference and background difference [J]. Jilin university (engineering), 2016, 46(04):1325-1329.
[4] Kuzume, K., Okada, M.. Sensor network system to promote energy conservation realization of energy smart school [P]. Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on, 2014.
[5] Shiya Liu. Research on pedestrian detection algorithm based on mixed features [D]. Dongbei University, 2013.
[6] Cheng Xu, Zheng Tian, Renfa Li. A fast motion detection algorithm based on improved codebook model [J]. Computer research and development, 2010, 47(12):2149-2156.
[7] Two Classes of Codebooks Nearly Meeting the Welch Bound. Aixian Zhang, Keqin Feng. IEEE Transactions on Information Theory. 2012
[8] Daniel Berjón, Carlos Cuevas, Francisco Morán et al.. Real-time nonparametric background subtraction with tracking-based foreground update [J]. Pattern Recognition, 2018
[9] Thangarajah Akilan, Q.M. Jonathan Wu, Yimin Yang. Fusion-based foreground enhancement for background subtraction using multivariate multi-model Gaussian distribution [J]. Information Sciences, 2018, 430-431.