International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.12 (2015), pp.285-296
http://dx.doi.org/10.14257/ijsip.2015.8.12.27

A Multiple Moving Object Segmentation Algorithm Based on Background Modeling and Adaptive Clustering

Zhengyi Hu[1,2,*], Qingchang Tan[1], Kun Zhang[3] and Xin Wang[3]

[1]*College of Mechanical Science and Engineering, Jilin University, Changchun, 130000, China*
[2]*Changchun automobile industry institute, Changchun, 130000, China*
[3]*Northeastern University, 110084, Liaoning, China*
luozhangyulong@163.com

Abstract

A multiple moving object segmentation algorithm based on Background Modeling and Adaptive Clustering (named as BMAC) algorithm is proposed in this paper. For moving object segmentation, the algorithm uses Chebyshev inequality and the kernel density estimation method to do background modeling firstly. Then in order to classify image pixels as background points, foreground points and suspicious points, an adaptive threshold algorithm is proposed accordingly. After using background modeling, adaptive clustering is used for multi-object segmentation. It defines pixel space connectivity rate and designs a perpendicular split method, initial cluster adaptive splitting and merging self-organizing the iterative clustering segmentation algorithm, without pre-set number of clustering, completes multi-object segmentation for the foreground image. The segmentation results are consistent with the human visual judgment, the use of space connectivity information improve the accuracy of clustering segmentation, comparison and analysis the experimental results show that the proposed algorithm is feasible, rapid and effective.

Keywords: *Background Modeling; Chebyshev Inequality; Adaptive Clustering; Multi-object Segmentation*

1. Introduction

With the computer hardware and software technology continues to mature, a variety of multimedia technology begins to flourish, so does the network video monitoring system. In order to deal with the massive recorded video content, it requires an intelligent video surveillance system to analyze the critical target automatically. The detection and segmentation of moving objects from a video stream is a basic and fundamental problem of tracking and traffic control *etc.*

Main segmentation methods include histogram threshold [1], features clustering [2-4], area-based method [5] *etc.* Recent image segmentation approaches have provided interactive methods that implicitly define the segmentation problem relative to a particular task of content localization. This approach to image segmentation requires users (or preprocessor) guidance of the working algorithm to define the desired object to be extracted. For example, frame difference method [6], background subtraction [7] and optical flow method [8]. Elgammal [9] proposed kernel density estimation method for each pixel location in the video sequence to establish non-parametric probability model, using a number of neighboring frames sample value constructed close to the actual probability distribution. Nancy M. Salem [10] proposed a new segmentation scheme for the white blood cells from microscopic images is proposed. The method is based on the

[*] Corresponding Author

ISSN: 2005-4254 IJSIP

K-means clustering technique. Zhao Zaixin [11] proposed a clustering algorithm for image segmentation. Not only the contrasts among regions are different, but also the gradual changing grey values are varied in the same region in a wide range. These problems present segmentation with difficulties. It is needed to perform detection respectively for these different target areas.

In this paper we explain the related work of our algorithm, including Chebyshev inequality, the kernel density estimation and adaptive clustering. Section 3 describes the proposed algorithm in detail. Section 4 provides the experimental results and the analysis. Finally, there are conclusion and references.

2. Related Work

Chebyshev inequality can distinguish the background pixels and foreground pixels quickly. Let random variable X with its expectation $E(x) = \mu$ and its variance $D(x) = \sigma^2$, for any positive number $\varepsilon > 0$, there is the Chebyshev inequality

$$P\{|X - \mu| \geq \varepsilon\} \leq \frac{\sigma^2}{\varepsilon^2} \quad (\text{or } P\{|X - \mu| < \varepsilon\} \geq 1 - \frac{\sigma^2}{\varepsilon^2}) \tag{1}$$

We define a random variable X represent the value of each pixel for foreground segmentation, the probability of event $\{|X - \mu| < \varepsilon\}$ can be estimated by the Chebyshev inequality, it reflects the changing circumstances about the corresponding value of video image pixels.

The kernel density estimation technique is a particular nonparametric technique. In this technique, the underlying probability density function is estimated as

$$\hat{f}(x) = \sum_i \alpha_i K(x - x_i) \tag{2}$$

Let $x_1, x_2, \cdots, x_N$ be a sample of intensity values for a pixel. Given the observed intensity x_t at time t, we can estimate the probability of this observation as

$$P_r(x_t) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{(x_t - x_i)^2}{\sigma^2}} \tag{3}$$

Here we choose the kernel function K to be Gaussian. If we assume that this local-in-time distribution would be Gaussian $N(\mu, \sigma^2)$, then the distribution for the deviation $(x_i - x_{i+1})$ would be Gaussian $N(0, 2\sigma^2)$. The kernel function bandwidth σ can be estimated as

$$\hat{\sigma} = m \big/ (0.68\sqrt{2}),\ m = median(|x_i - x_{i+1}|),\ \sigma = \max(1, \hat{\sigma}) \tag{4}$$

For image segmentation K-means clustering method is the use of feature vectors in feature space pixel to each pixel is divided into different clustering to achieve image segmentation. Given a set of observations $x_1, x_2, \cdots, x_N$, where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into $k(\leq n)$ sets $S = \{S_1, S_2, \cdots S_k\}$ so as to minimize the within-cluster sum of squares (WCSS). In other words, its objective is to find:

$$\arg\min_S \sum_{i=1}^{k} \sum_{x \in S_{ii}} \|x - \mu_i\|^2 \tag{5}$$

Where μ_i is the mean of points in S_i. K-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

Segmentation of images with uneven regions has long been a hard nut in image processing. In view of the above methods, in this paper we proposed a multi-object segmentation algorithm based on Background Modeling and Adaptive Clustering algorithm which is named as BMAC algorithm.

3. Background Modeling and Adaptive Clustering Algorithm

In this section, we describe the proposed BMAC algorithm in detail. The algorithm uses pixel intensity (gray scale or color) as the basic feature for the background modelling and region growing. Let $X_1, X_2, \cdots, X_N$ be a sample of adjacent N-frame video images, the location of the pixel in the image is marked as (i, j), $X_1(i, j), X_2(i, j), \cdots, X_N(i, j)$ represent the pixel gray values in the given position (i, j) for these images. The BMAC algorithm is as follows:

(1) Calculate the sample mean $\bar{X}(i, j)$ and sample second-order centre distance $S(i, j)$ for each pixel.

$$\bar{X}(i, j) = \frac{1}{N}\sum_{k=1}^{N} X_k(i, j), \quad S(i, j) = \frac{1}{N}\sum_{k=1}^{N} (X_k(i, j) - \bar{X}(i, j))^2 \tag{6}$$

Here, $\bar{X}(i, j)$ and $S(i, j)$ are the maximum likelihood estimators of the expectation $E(X(i, j)) = \mu(i, j)$ and variance $D(X(i, j)) = \sigma^2(i, j)$. That is

$$\hat{\mu}(i, j) = \bar{X}(i, j), \hat{\sigma}^2(i, j) = S(i, j) \tag{7}$$

(2) Calculate $\sigma_k^2(i, j) = (X_k(i, j) - \bar{X}(i, j))^2$ for the frame $X_k(k = 1, 2, \cdots, N)$. The calculation results of $\sigma_k^2(i, j)/\varepsilon^2$ and $1 - \sigma_k^2(i, j)/\varepsilon^2$ for different ε values show in Table 1.

Table 1. Different Calculation Results According to Different ε

ε	$\hat{\sigma}$	$1.225\hat{\sigma}$	$1.414\hat{\sigma}$	$1.581\hat{\sigma}$	$1.732\hat{\sigma}$	$1.871\hat{\sigma}$	$2\hat{\sigma}$
$\hat{\sigma}^2/\varepsilon^2$	1	0.667	0.5	0.4	0.333	0.286	0.25
$1 - \hat{\sigma}^2/\varepsilon^2$	0	0.333	0.5	0.6	0.667	0.714	0.75

If $\sigma_k^2(i, j) < \hat{\sigma}^2(i, j)$ or $1 - \sigma_k^2(i, j)/\varepsilon^2 > 1 - \hat{\sigma}^2(i, j)/\varepsilon^2$, the gray value change of pixel $X_k(i, j)$ is smaller than the variance , this pixel is more likely to be a background point. If $\sigma_k^2(i, j) > \hat{\sigma}^2(i, j)$ or $1 - \sigma_k^2(i, j)/\varepsilon^2 < 1 - \hat{\sigma}^2(i, j)/\varepsilon^2$, this pixel is more likely to be a foreground point.

(3) For each $X_k(i, j)(k = 1, 2, \cdots, N)$ the adaptive classification threshold setting formula is as follows:

$$\begin{cases} T_1 = \dfrac{(1 - \hat{\sigma}^2(i, j)/\varepsilon_1^2) + 0.5}{2}, & \varepsilon_1 = 1.414\hat{\sigma}(i, j) + \theta_1\hat{\sigma}(i, j) \\ T_2 = \dfrac{(1 - \hat{\sigma}^2(i, j)/\varepsilon_2^2) + 0.5}{2} & \varepsilon_2 = 1.414\hat{\sigma}(i, j) - \theta_2\hat{\sigma}(i, j) \end{cases} \quad (0 < \theta_1, \theta_2 < 1) \tag{8}$$

Adaptive threshold T_1 and T_2 can changed with the change of $\hat{\sigma}$ for a given θ_1 and θ_2. Different video frames conditions, the above-mentioned formula can calculate the foreground and background distinguish threshold T_1 and T_2 effectively. Adjust the coefficients θ_1 and θ_2 make $0 < |\varepsilon_1 - \varepsilon_2| < 2\hat{\sigma}$, the greater θ_1 the larger T_1, but the greater θ_2 the smaller T_2.

(4) Use the following formula for $X_k(i,j)(k = 1, 2, \cdots, N)$ to classify background points, foreground points and suspicious points.

$$C_k^1(i,j) = \begin{cases} 1 & (1 - \dfrac{\sigma_k^{\,2}(i,j)}{\varepsilon^2}) \ge T_1 \\ 0 & (1 - \dfrac{\sigma_k^{\,2}(i,j)}{\varepsilon^2}) \le T_2 \end{cases} \qquad \varepsilon = 1.414\hat{\sigma} \tag{9}$$

When $C_k^1(i,j) = 1$, the pixel is classified as background point. When $C_k^1(i,j) = 0$, the pixel is classified as foreground point. The pixel is classified as suspicious point When $T_2 < (1 - \sigma_k^2(i,j)/\varepsilon^2) < T_1$, for suspicious point we use density estimation method for further discrimination. If $T_1 = T_2 = 0.5$, the suspicious point set is empty, at this time the BMAC algorithm uses only Chebyshev inequality for background subtraction, and if $T_1 = 1,\ T_2 = 0$, all pixels are suspicious points, the BMAC algorithm uses only kernel density estimation for background subtraction.

(5) Calculate $P_r(X_k) = \dfrac{1}{N}\sum_{i=1}^{N}\dfrac{1}{\sqrt{2\pi\hat{\sigma}^2}}e^{-\frac{1}{2}\frac{(X_k - X_i)^2}{\hat{\sigma}^2}}$ for suspicious point, set threshold T_3 and classification function:

$$C_k^2(i,j) = \begin{cases} 1 & P_r(x_k) \ge T_3 \\ 0 & P_r(x_k) < T_3 \end{cases} \tag{10}$$

When $C_k^2(i,j) = 1$, the suspicious point is further identified as a background point. When $C_k^2(i,j) = 0$, the suspicious point is further identified as foreground point.

(6) Update the background pixel value using the following formula:

$$\begin{cases} H_N^1(i,j) = \sum_{k=1}^{N} X_k(i,j) \cdot C_k^1(i,j),\ Q_N^1 = \sum_{k=1}^{N} C_k^1(i,j), & (i,j) \notin K \\ H_N^2(i,j) = \sum_{k=1}^{N} X_k(i,j) \cdot C_k^2(i,j),\ Q_N^2 = \sum_{k=1}^{N} C_k^2(i,j), & (i,j) \in K \\ B_N(i,j) = \dfrac{H_N^1(i,j) + H_N^2(i,j)}{Q_N^1(i,j) + Q_N^2(i,j)}, & (i,j) \in K \cup \overline{K} \end{cases} \tag{11}$$

Here, $B_N(i,j)$ means the pixel value in the position (i,j) of the background image (marked as B_N) for the sample $X_1, X_2, \cdots, X_N$, K is a collection of suspicious point location. It can be seen that the background image updating is based on the real-time sample $X_1, X_2, \cdots, X_N$. Therefore, real-time background modeling can be achieved simply by adding new real-time adjacent N-frame video images and ignoring previous samples.

(7) Complete the foreground image segmentation through background subtraction method. These foreground images are marked as $F_k(k = 1, 2, \cdots, N)$. T_3 and T_4 are empirical values, generally based on the specific test data to select.

$$F_k(i,j) = \begin{cases} X_k(i,j) & X_k(i,j) - B_N(i,j) \ge T_4 \\ 255 & X_k(i,j) - B_N(i,j) < T_4 \end{cases} \quad (k = 1, 2, \cdots, N) \tag{12}$$

(8) Read foreground image pixel and the new sample: $X_1, X_2, \cdots X_N$ whose gray-scale pixel values are not 255 (referred N as the total number). Set c_j as the initial cluster centers, N_c as the initial number of clusters, T as the space communication rate threshold; T_{Split} as the cluster splitting threshold; T_{Merge} as the cluster merge threshold.

The sample point is divided according to the closest distance to the nearest cluster S_j, set $D_j = \min\{\|X - c_j\|, j = 1, 2, \cdots N_c\}$, if $\|X - c_j\|$ is the minimum distance, then $X \in S_j$. Amendment to the cluster center as:

$$z_j = \frac{1}{N_j} \sum_{X \in S_j} X, \ j = 1, 2, \cdots, N_c \tag{13}$$

(9) Calculate the clustering connectivity rate l_{ij}^T of each sample point t:

$$l_{ij}^T = \|x_i - c_j\| / n_{ij}^T, \ x_i \in S_j, i = 1, 2, \cdots, N_j, \ j = 1, 2, \cdots, N_c \tag{14}$$

Define pixel space connectivity rate of pixel X_i and X_j as $l_{ij}^T = d_{ij} / n_{ij}^T$, here $d_{i,j} = \|X_i - X_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, the location of the pixel X_i in the image is marked as (x_i, y_i). Connect the pixel X_j and X_i obtain the segment $\overline{X_j X_i}$, located a distance T. We get a rectangle (shown in Figure 1), wide $2T$, long $\overline{X_j X_i}$, statistics the number of pixels in the rectangular marked as n_{ij}^T.

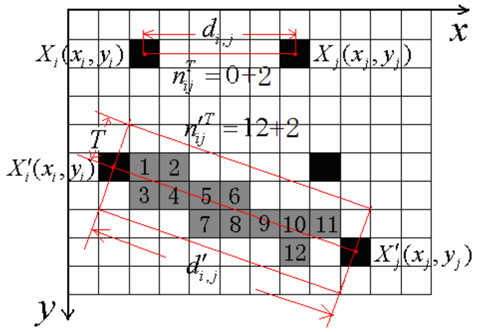


Figure 1. Pixel Space Connectivity Rate $l_{ij}^T = d_{ij} / n_{ij}^T$

(10) According to the cluster splitting threshold T_{Split}, using the vertical clustering split (shown in Figure 2), clustering division, and fix the cluster center, updating a sample number of clusters.

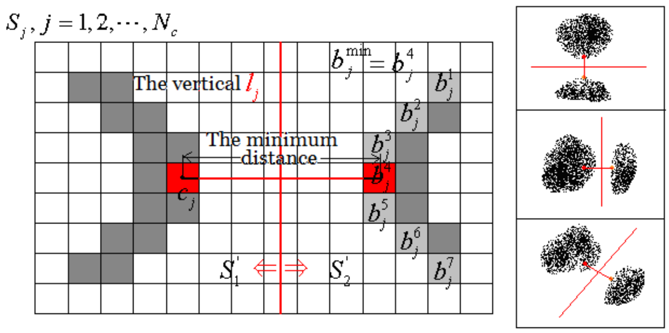


Figure 2. The Vertical Clustering Split

If $l_{ij}^{T} > T_{Split}$, marked the sample point as b_j, named as suspicious boundary point, all suspicious boundary point of cluster S_j consist a collection B_j. Calculate the boundary distance between b_j and c_j, then find the minimum distance and mark the corresponding boundary point as $b_j^{\min}$. Obtain the vertical dividing line of the segment which two end points are $b_j^{\min}$ and c_j .According to herein vertical as the split-line (shown in Figure 1), the cluster $S_j, j = 1, 2, \cdots, N_c$ split into two new clusters $S_1^{'}, S_2^{'}$.

According to the formula (14) update new cluster center coordinates c_1', c_2', and update the number of clusters $N_c = N_c + 1$. The number of clusters before cluster split marked as N_{div}, the number of clusters after clustering split marked as N'_{div}. After the above steps, adaptive clustering division process completed.

(11) Calculate connectivity rate $l_{c_i c_j}^{T}$ for the updated cluster centers and other cluster centers:

$$l_{c_i c_j}^{T} = \left\| c_i - c_j \right\| \Big/ n_{c_i c_j}^{T}, i = 1, 2, \cdots, N_c - 1, j = i + 1, i + 2, \cdots, N_c \tag{15}$$

If $l_{c_i c_j}^{T} < T_{Merge}$, then clustering S_i and S_j combined to a new cluster, and update the new cluster center as:

$$c'_{i+j} = \frac{1}{N_i + N_j}(N_i * c_i + N_j * c_j) \tag{16}$$

Update the number of clusters $N_c = N_c - 1$, and in accordance with the minimum distance method and the new cluster center coordinates, update the sample points clustering division. The number of clusters before cluster merged marked as N_{cn}, the number of clusters after clustering merged marked as N'_{cn}. The above steps are completed the merger process of adaptive clustering.

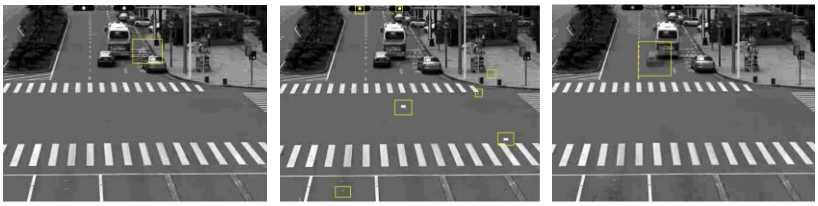
(12) If $N_{div}^{'} = N_{div}, N_{cn}^{'} = N_{cn}$ or reaches the maximum defining number of iterations, Then clustering splitting and merging process is completed, clustering iterative split end, output final image clustering segmentation results. Otherwise, go to the step (9) for iterative clustering operations.

4. Experimental Results

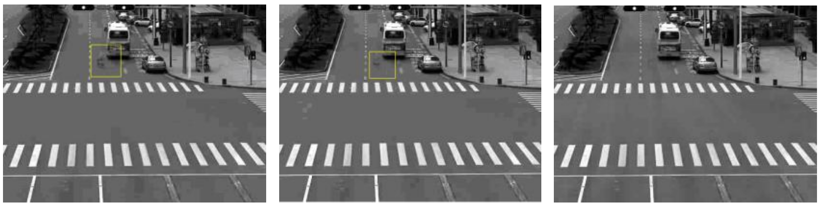
To demonstrate the feasibility and effectiveness of the BMAC algorithm, in this paper, we use the Intel (R) Core (TM)2 6300 CPUs, 1.86GHz, 1GB memory PC, use the Java language programming in the Eclipse development platform. The video images are captured from 336×448 traffic surveillance video (shown in Figure 3).

Figure 3. Surveillance Video Images

The background modeling experimental results of the BMAC algorithm about different threshold settings is shown in Figure 4.



(1) $N = 20, \theta_1 = 0.6, \theta_2 = 0.4, T_3 = 0.005$ (2) $N = 20, \theta_1 = 1, \theta_2 = 0, T_3 = 0.005$ (3) $N = 40, \theta_1 = 0.5, \theta_2 = 0.5, T_3 = 0.005$

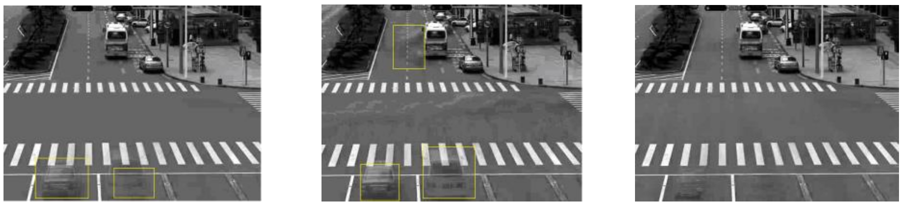


(4) $N = 40, \theta_1 = 1, \theta_2 = 0, T_3 = 0.005$ (5) $N = 80, \theta_1 = 0.5, \theta_2 = 0.5, T_3 = 0.005$ (6) $N = 80, \theta_1 = 1, \theta_2 = 0, T_3 = 0.005$

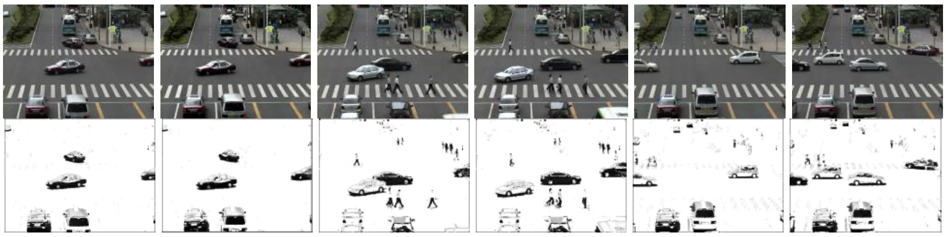
Figure 4. Background Modeling Experimental Results

Here we chose gray scale images for example to carry out the experiments. On the basis of a large number of experimental data, we compare and analysis the experimental results of different threshold settings. From the Figure 4 it can be seen that for the same parameter θ_1, θ_2, T_3 settings, the greater the number of samples N the experimental effect is more ideal (shown in Figure 4-,(3),(5)), for the same N, T_3 settings, the larger θ_1 and the smaller θ_2 the experimental effect is more ideal (shown in Figure 4-(3),(4)). However, when N is small, there will be some empty area of the background image (shown in Figure 4-(2)). Therefore, according to different background modeling requirements, adjust the parameters properly, the BMAC algorithm can achieve an ideal background modeling results (shown in Figure.4-(6)).

In order to test the experimental results of the BMAC algorithm, we chose the frame-difference algorithm [6] and the original kernel density estimation algorithm [10] for comparison. For the same video frame images, experimental results of background modeling for three different algorithms are shown in Figure 5-(1-3).



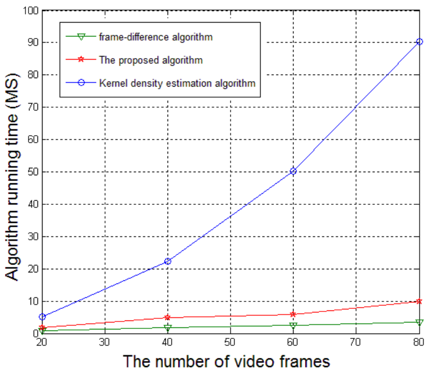
(1) Frame-difference algorithm (2) Kernel density estimation (3) The BMAC algorithm

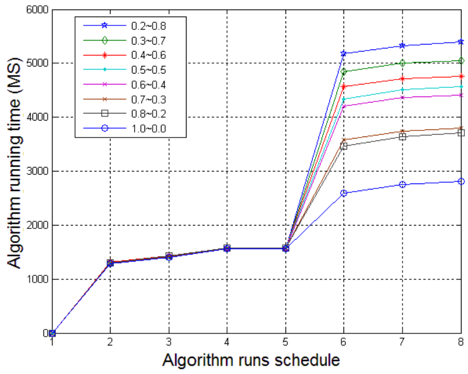


(4) Foreground image

Figure 5. Background Modeling and Foreground Image Extraction Experimental Results

From Figure 5, we can see that the BMAC algorithm has the best experimental result. It can get good foreground images (shown in Figure 5-(4)), but also can save computation cost, this is another superiority of our algorithm. The comparison chart of algorithm running time is shown in Figure 6.

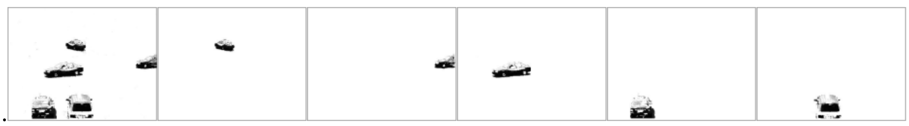




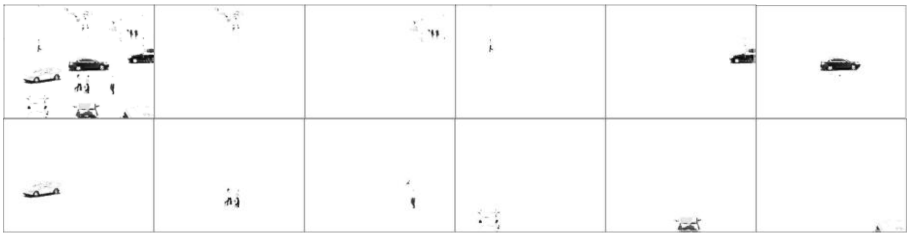
(1) Three Algorithms Run-Time Comparison Chart (2) Different θ_1,θ_2 Algorithm Run-Time Chart

Figure 6. Algorithm Running Time Comparison Charts

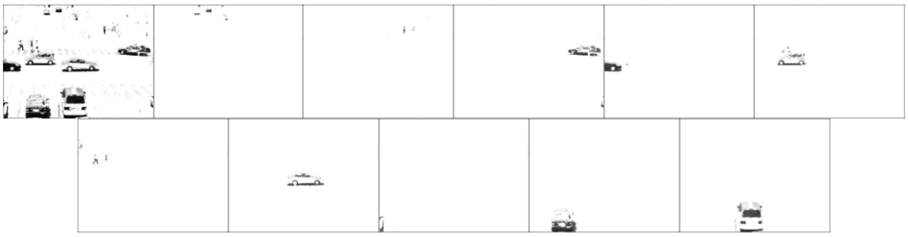
In the same number of frame images, compared with frame difference method and kernel density estimation algorithm, the background modelling time of our algorithm is far less than the kernel density estimation method, only slightly higher than the frame-difference algorithm. Consider both real-time and effectiveness, the advantages of our algorithm is obviously. Set $T=1$, $T_{Split}=60$ and $T_{Merge}=60$ based on statistical regularities, the multi-object segmentation experimental results are shown in Figure 7.



(1) Multi-object segmentation experimental result-1



(2) Multi-object segmentation experimental result-2

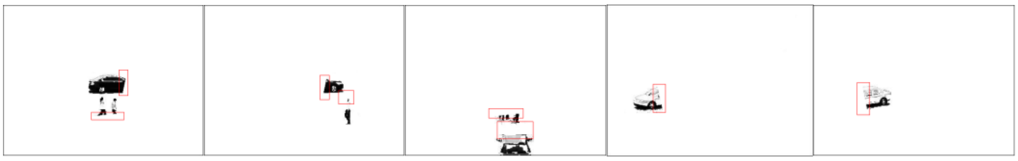


(3) Multi-object segmentation experimental result-3

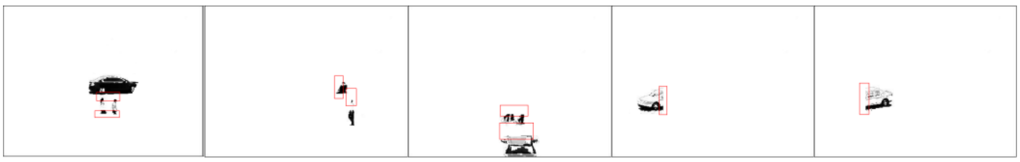
Figure 7. Multi-Object Segmentation Experimental Result

The segmentation results are consistent with the human visual judgment, the use of space connectivity information improve the accuracy of clustering segmentation, comparison and analysis the experimental results show that the proposed algorithm is feasible, rapid and effective.

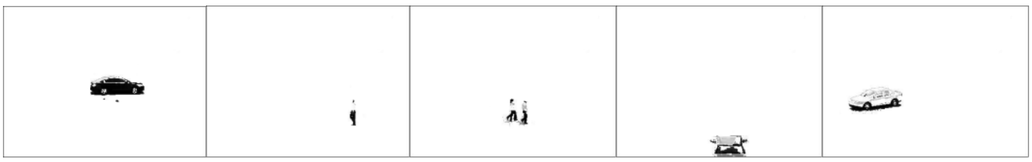
Select the K-means algorithm and ISODATA (Iterative Self-Organizing Data Analysis Techniques Algorithm) [11] as the comparison algorithm, for the same foreground image(shown in Figure 7-(2)), the multi-object segmentation experimental results of these three algorithm are shown in Figure 8.



(1) Experimental result of K-means



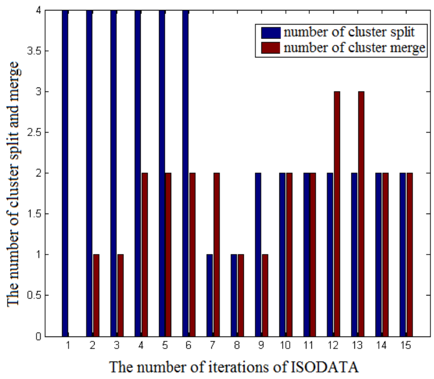
(2) Experimental result of ISODATA



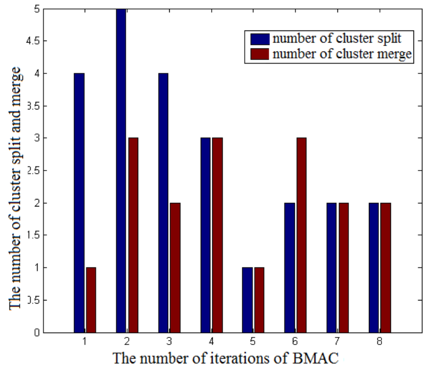
(3) Experimental result of BMAC

Figure 8. Multi-Object Segmentation Experimental Result

When the targets have short distance the segmentation result of K-means algorithm shows that one target has been split into two (shown in Figure 8-(1)), ISODATA algorithm improved, but there still has division error (shown in Figure 8-(2)), while the proposed BMAC algorithm (shown in Figure 8-(3)) has the best target segmentation result.



(1) The Number of Cluster Split and Merge of ISODATA



(2) The Number of Cluster Split and Merge of BMAC

Figure 9. Compare the Total Number of Iterations

We analysis the number of cluster split and merge generated during each iteration for BMAC algorithm and ISODATA algorithm, and compare the total number of iterations (shown in Figure 9). As can be seen the number of iterations of the algorithm BMAC is 8(shown in Figure 9-(2)), significantly less than ISODATA algorithm. BMAC iterative algorithm converges faster is better than ISODATA algorithm.

5. Conclusion

In this paper, relating to the multiple moving object segmentation we proposed a combination of background modeling and adaptive clustering algorithm for multi-object segmentation. The application of Chebyshev inequality, setting of the adaptive threshold algorithm, initial cluster adaptive splitting and merging self-organizing the iterative clustering segmentation algorithm, the proposed BMAC algorithm achieve multiple moving targets intelligent segmentation. The experimental results show that the background modeling results are significantly better than the K-mean and ISODATA algorithm, and it can save more running time than the kernel density estimation method. Multi-object segmentation results by adaptive clustering algorithm are very satisfactory. Therefore, the BMAC algorithm is feasible, rapid and effective and is suitable for a intelligent monitoring system.

References

[1] G. Koç and B. Sarıogu, “Statistical analysis of threshold algorithms in image processing based cancer cell detection”, Signal Processing and Communications Applications Conference (SIU), **(2014)**.

[2] H. Al-Jubouri, Hongbo Du and H. Sellahewa, “Adaptive clustering based segmentation for image classification”, Computer Science and Electronic Engineering Conference (CEEC), **(2013)**.

[3] M. Mignotte, “A de-texturing and spatially constrained K-means approach for image segmentation”, Pattern Recognition Letters, vol. 32, no. 2, **(2011)**.

[4] Z. Ji, Y. Xia and Q. Chen, “Fuzzy c-means clustering with weighted image patch for image segmentation”, Applied Soft Computing, vol. 12, no. 6, **(2012)**.

[5] O. P. Verma1, M. Hanmandlu and S. Susan, “A Simple Single Seeded Region Growing Algorithm for Color Image Segmentation using Adaptive Thresholding”, International Conference on Communication Systems and Network Technologies, **(2011)**.

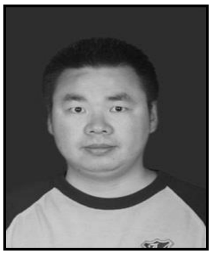
[6] W. Ru, Z. Chunjiang and X. Guo, “Improved cam-shift algorithm based on frame-difference method for video's auto tracking”, International Journal of Digital Content Technology and its Applications, vol. 6, no. 19, **(2012)**.

[7] H. Liu and X. Hou, “Moving detection research of background frame difference based on Gaussian model”, 2012 International Conference on Computer Science and Service System (CSSS), Nanjing, **(2012)**.

[8] K. Chen and L. DA, “Image sequence interpolation based on optical flow”, segmentation, and optimal control, IEEE Transactions on Image Processing, vol. 21, no. 3, **(2012)**.

[9] A. Elgammal, R. Duraiswami and D. Harwood, "Back ground and foreground modeling using nonparametric kernel density estimation for visual surveillance", Proceedings of the IEEE, vol. 90, **(2002)**.
[10] N. M. Salem, "Segmentation of white blood cells from microscopic images using K-means clustering", Radio Science Conference (NRSC), **(2014)**.
[11] Z. Zaixin, C. Lizhi and C. Guangquan, "Neighbourhood weighted fuzzy c-means clustering algorithm for image segmentation", Image Processing, IET, vol. 8, no. 3, **(2014)**.

Authors



ZhengYi Hu, he received his master in Mechanical Engineering (2010) from Changchun University of Science and Technology. Now he is a PhD candidate in College of Mechanical Science Engineering of Jilin University. He is mainly engaged in the different aspects of Machine Vision about Automotive industry.