# SDIES: A Background subtraction method with sample dynamic indicator and edge similarity

**Lian Huang**
*Key Laboratory of Intelligent Information Processing and Control*
*Chongqing Three Gorges University*
*Chongqing*
*China*
*hlcqsxxy@163.com*

**Yuming Feng**
*Chongqing Engineering Research Center of Internet of Things and Intelligent Control Technology*
*Chongqing Three Gorges University*
*Chongqing*
*China*
*yumingfeng25928@163.com,ymfeng@sanxiau.edu.cn*

**Li Cai**
*Key Laboratory of Intelligent Information Processing and Control*
*Chongqing Three Gorges University*
*Chongqing*
*China*
*39485885@qq.com*

**Wei Zhang***
*School of Three Gorges Artificial Intelligence*
*Chongqing Three Gorges University*
*Chongqing*
*China*
*weizhang@sanxiau.edu.cn, cqec126@126.com*

**Babatunde Oluwaseun Onasanya**
*Key Laboratory of Intelligent Information Processing and Control*
*Chongqing Three Gorges University*
*Wanzhou, Chongqing, 404100*
*P. R. China*
*Department of Mathematics*
*University of Ibadan*
*Oyo State 200284*
*Nigeria*
*babtu2001@yahoo.com*

∗. Corresponding author.

**Abstract.** Traditional sample-based methods are inefficient for detecting dynamic background and intermittent object motion. To solve the problem, a background subtraction method based on sample dynamic indicator and edge similarity is proposed. This method utilizes the standard deviation of recently observed decision distances as a background dynamic indicator and also uses the improved blink pixel estimator to control adaptive threshold feedback. Unlike the traditional sample-based methods, this method estimates the reference background at the initialization stage and distinguishes static objects from ghosts by calculating the edge similarity of foreground objects in reference background and current frame. So the existence property of foreground objects can suppress ghosts quickly. The evaluation results on the ChangeDetection 2012 dataset indicate that this method can be well adapted to the dynamic background and intermittent object motion. Besides, without paying too much attention to other scenario categories, the overall evaluation performance is comparable to most state-of-the-art methods.

**Keywords:** background subtraction, without neighborhood diffusion, edge similarity, sample dynamic.

## 1. Introduction

Motion foreground detection is the bottom task of intelligent video surveillance [1], visual object tracking [2] and optical motion capture [3]. Background subtraction (BS) is widely used to separate the motion objects called foreground and the static objects called background from video sequences. The main steps consist of background model initialization, foreground segmentation and background model update. Firstly, BS estimates a reference background by some historical frames. Then, a binary classification is generated by comparing the difference between the reference background and the current frame. Background model update makes the model to adapt well to the changes of video scenes. In the last two decades, a lot of BS methods have been proposed. Due to the existing typical challenges such as dynamic background, illumination change, camera jitter, shadows, etc., BS is still an active field of research. Significant progress has been made in BS methods and the following are some recent surveys [4, 5].

One of the most popular strategies for BS is probability-based method. Wren et al. [6] were the first to propose a single Gaussian distribution to present the color distribution of background pixels. Stauffer and Grimson [7, 8] proposed the Mixture of Gaussian Model (MoG), which was one of the most famous multimodal BS methods. MoG assumed that pixels obeying the normal distribution in time domain were background and could be used for updating background model and that other pixels that did not obey the normal distribution were foreground. While MoG could adapt to slightly dynamic background, as well as light change, it was unable to deal with highly dynamic background , as detailed in [11]. Varadarajan et al. [9, 10] proposed a Region-based Mixture of Gaussian Model (RMoG) framework, which took the neighboring pixels into account when constructing the model for the observed scenes. Besides, they theoretically derived the update equations and extended the standard MoG more generally. However, the RMoG faces high computational costs.

Different from the idea of MoG, Barnich et al. [11] proposed a simpler and more efficient sample-based method named Visual Background Extractor (ViBe). Instead of estimating a reference background, this method kept several feature values (called samples or background samples) for the position of one pixel so that foreground segmentation and update tasks could make full use of the samples' information. ViBe was inspired by Sample Consensus (SACON) method [12, 13], which assumed that the pixels whose matching counts with samples exceed the threshold are background and other pixels are foreground. Moreover, ViBe creatively adopted neighborhood diffusion to suppress ghosts. Note that ViBe is easily implemented at high speed and is suitable for many scenarios. Researchers inspired by ViBe have proposed many improved methods [14, 15].

However, the fixed segmentation threshold of ViBe reduces the ability to adapt to the dynamic background. Droogenbroeck and Paquot [16] suggested an improved method named ViBe+. They followed the suggestion of [17] that ViBe should use a threshold associated with samples in the model for a better handling of camouflaged foreground. The half of standard deviation of samples was defined as the adaptive threshold. Instead of choosing the same threshold control strategy, Hofmann et al. [18] nominated the Pixel-Based Adaptive Segmenter (PBAS) method, which used average of currently observed minimal distance to estimate the dynamic degree. Threshold can achieve adaptive feedback in a liner way for static or dynamic region. However, the response time was relatively long. To counter this issue, Stcharles et al. [19] proposed Self-Balanced Sensitivity Segmenter (SuBSENSE) method, which introduced the blink pixel, a strictly-positive factor, to control feedback of segmentation threshold. Different from the above methods, Varghese and Sreelekha [20] pointed out that the adaptive minimal cardinality min can get better detection results. min is adapted for each pixel independently with background dynamic estimated. Although their methods can suppress dynamic noises, there is an over-segmentation problem when foreground objects pass through the dynamic region.

The neighborhood diffusion mechanism of ViBe can gradually eliminate ghosts, but the foreground object can be easily eaten-up slowly from the surrounding region since the low-speed motion and static object have similar characteristics with ghosts. ViBe+ calculated the gradient of the inner boundary of the background spot and that of the foreground spot. The neighborhood diffusion was restrained when the gradient value exceeded threshold. PBAS and SuBSENSE still utilized average of currently observed minimal distance to control update rate, which made sure foreground object shape is more complete. Note that the essence of these methods is to minimize the negative effects of the neighborhood diffusion mechanism. In general, neighborhood diffusion mechanism is not positive for intermittent motion objects.

Recently, some other complex BS methods are proposed, such as fuzzy background models [21, 22, 23], discriminative subspace learning models [24, 25], robust principle component analysis models [26, 27], sparse models [28, 29] and transform domain models [30, 31, 32]. Following the simple and practical principles, we still choose the framework of simple-based methods. In addition, selecting two or more features can obtain better detection results, though it also can increase computing costs. Thus, only a single color feature is selected. In addition, it is difficult for the sample-based methods to obtain the reference background directly. In our initialization phase, not only the regular background samples are built, the reference background is also accurately estimated. Our method can benefit from this reference background. Neighborhood update is the particular way of sample-based methods.

However, this mechanism has inherent defect in detecting intermittent motion objects. Our method does not select the neighborhood update. The edge similarity is utilized to determine whether the object is a foreground or a ghost, and then different updates are adopted to overcome the above defect. Some sample-based methods use adaptive feedback mechanism to control dynamic background noise which can easily cause over-segmentation. Our method utilizes the standard deviation of recently observed decision distances as the background dynamic indicator and improves blink pixel estimator. These guarantee the accurate shape of foreground objects while suppressing dynamic background.

This paper is organized as follow: Section 2 describes the initialization, reference background estimation, foreground segmentation and background update of the proposed method; Section 3 firstly sets universal parameters and then evaluates the detection quality of the proposed method from qualitative and quantitative aspects. Finally, section 4 summarizes the method.

## 2. Method description

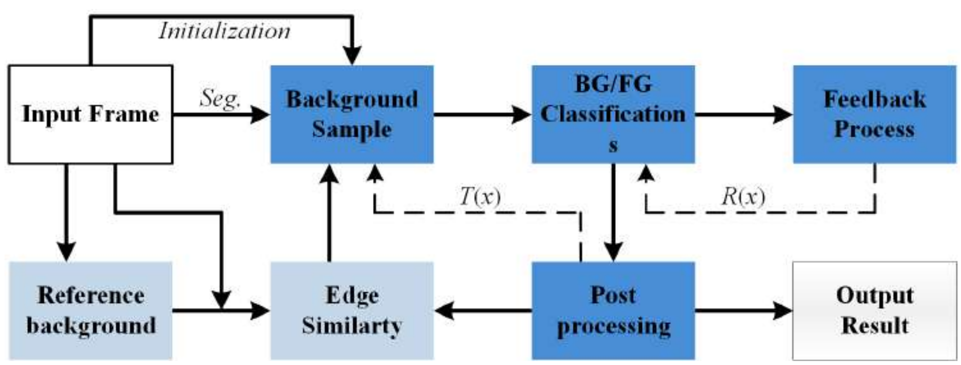


Figure 1: Block diagram of proposed SDIES method.

In general, there are two initialization operations for sample-based methods, such as selecting recently observed consecutive frames or repeatedly sampling from neighborhood regions of pixels in the single frame. These operations can all give pixels multi-modal distribution. However, using single frame for initialization may cause the situation that foreground pixels exist in background samples. Moreover, variations among consecutive frames are often subtle. Therefore, in order to estimate the reference background, the traditional initialization method needs changes. In our method, we select discontinuous frames with fixed time intervals and increasing the sample intervals between each frame makes the difference between the current frame and the previous frame more obvious. Figure 1 illustrates the overall diagram of our method and Algorithm 1 describes our pseudocode in detail. The background model B is defined as

Algorithm 1. Pseudocode for SDIES

*a.* Use $(N+1)(P+1)$ frames construct background model
*b.* Estimate reference background by background model
*c.* Foreground segmentation
    for each$I(t)$ that $t > (N+1)(P+1)$
        $\mathrm{F}(x)$ = BG/FG classifications
    update background model with post-processing version of $\mathrm{F}(x)$
    calculate edge similarity of foreground objects between reference background and $I(t)$
    if foreground objects are ghosts
        $com_b(x) = com_b(x) + 1$
        if $com_b(x)$ exceeds threshold
            update foreground objects into background model
*d.* Use 9*9 median filter of $\mathrm{F}(x)$ as final output

$$(1)\ B(x) = \{B_1(x), \cdots, B_N(x)\} \quad B_i(x) = I_{p(i-1)+1}(x) \ \ i = 1, 2, \cdots, N.$$

where $p$ is the fixed time interval, and $N$ is background sample number.

$I_{p(i-1)+1}(x)$ is the current frame at $p(i-1)+1$ moment. After sampling period $p$, the current frame $I_{p(i-1)+1}(x)$ is added to the background model until the $N$ frames are processed. In short, the time interval is extended from 1 to $p$. As mentioned, only color feature is used.

A. Reference Background Estimated

The traditional initialization ends at this stage. However, the reference background can be estimated with the existing background model. Our reference background estimation is inspired by Independent Multi-modal Background Subtraction (IMBS) [33], which holds that the background appears for a long time while the foreground is opposite. A pixel can be observed several times in the same position over a period of time, which indicates this pixel is more likely to be the background. So, we create the reference background sample $RB(x) = RB_1(x), ..., RB_N(x)$ and count $C(x) = C_1(x), ..., C_N(x)$. The first element $B_1(x)$ of the background sample is inserted into the reference background $RB_1(x)$, then count $C_1(x) = 1$. Starting with the second element, $B_2(x)$ will be compared with $RB_i(x)$ as follows:

$$\begin{cases} RB_i(x) = RB'_i(x), C_i(x) = C_i(x) + 1, & \text{if } \|RB_i(x) - B_i(x)\| \leq R_{ini}, \\ RB_{i+1}(x) = B_i(x), & \text{otherwise,} \end{cases} \tag{2}$$

$$RB_i(x) = \frac{B_i(x)C_i(x) + RB_i(x)}{C_i(x) + 1}, \tag{3}$$

where $\| \cdot \|$ is the Euclidean distance and parameter $R_{ini}$ is the distance threshold. If the distance difference between $RB_i(x)$ and $B_i(x)$ is small, then a fusion value is generated to replace the original background sample and corresponding counts adds 1. The specific fusion calculation process uses formula (3). On the contrary, $B_i(x)$ is inserted into the next position and the corresponding counts is set to 1. When $N$ background samples are processed, there are many values in each reference background position. So, unimportant reference background values need to be filtered. Traversing the index of the maximum value in $C(x)$, then the pixel value at corresponding index in $RB(x)$ is treated as the reference background. Note that reference background and background sample are independent. Now our method can suppress ghosts effectively and that will be described in detail later.

(a) highway (b) PETS2006 (c) abandonedbox (d) fountain01

Figure 2: Reference background estimate on some CDnet2012 sequences. Even if there are foreground objects in the initial video sequence, our method can estimate the background accurately.

B. Foreground Detection

Our pixel-level foreground detection mechanism is inspired by [11]. It determines whether a coming pixel can be foreground or background based on the similarity with background samples as shown in formula (4)

$$F(x) = \begin{cases} 1, & \text{if } \phi\{\|RB_i(x) - B_i(x)\| < R(x)\} < \phi_{\min}, \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

where $F(x) = 1$ implies foreground, $F(x) = 0$ implies background, $R(x)$ is the distance threshold, $\phi(\cdot)$ is the counts that distance difference between current pixel $I(x)$ and background sample $B_i(x)$ is close to $R(x)$. $\phi_{\min}$ is the minimum match counts threshold and we fixed $\phi_{\min} = 2$ like [11] for our method.

Setting a distance threshold $R$ is a challenging issue for background subtraction. Different video scenes have different characteristics, and global $R$ cannot adapt to the scenes changes [16]. An over-high or over-low threshold may lead to undesirable results. Naturally, designing the adaptive threshold adjustment strategies is a trend. As previously stated, the sample-based method is very sensitive to background samples reliability. Therefore, background dynamic degree is used as the basis for adjusting the threshold value. Similar to [18, 19], we use all the recently observed sample distance to estimate background dynamic. In order to achieve fast threshold response, the minimal distance (without considering whether the update is implemented or not) and the pixel classification state are observed. This strategy makes it possible for distance threshold $R$ to make quick feedback. It is noted that $R$ in static region, where foreground objects pass frequently, also presents a high value which makes it hard to detect the foreground object (canoe and boats sequences). To solve this problem, the blink pixels accumulator is adopted. Although the neighborhood update mechanism can suppress the ghost effect but it cannot distinguish the static foreground object from the ghost. Instead of neighborhood diffusion, our update could solve the main shortcoming of traditional sample-based methods and detect intermittent motion objects better. The detailed description is in the following subsection.

### C. Feedback Control

Foreground detection is a binary classification task. The dynamic background has the same motion attribute as the foreground object. Without any distinction, the dynamic background will be detected as the foreground object falsely. So, the main idea is to measure the motion entropy of a single pixel position in a small time window based on model fidelity. The motion entropy of dynamic background is bigger than static background. In our case, instead of [18, 19] using the moving average of the recently observed minimal distance to measure background dynamics, we select the standard deviation $S(x)$ of mean of all the observed distances between $I(x)$ and $B(x)$ as follows

$$S(x) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(D_{mean_i}(x) - \frac{1}{N}\sum_{i=1}^{N} D_{mean_i}(x)\right)^2}, \tag{5}$$

where $S(x)$ is the standard deviation and $D_{mean}(x)$ is the mean of all the observed distances between $I(x)$ and $B(x)$. The background motion entropy depends on sample differences, that is, the dispersion scale. The standard deviation is more consistent with the cognition of statistical data and more accurate representation of sample differences. In order to evaluate background dynamics, the threshold feedback is defined as follows:

$$R_t(x) = \begin{cases} R_{t-1}(x) + V(x), & \text{if } R_{t-1}(x) < 3S_t(x), \\ R_{t-1}(x) - \frac{1}{V(x)}, & \text{otherwise}, \end{cases} \tag{6}$$

where $t$ is the index of the image frame and $V(x)$ is the blink pixel accumulator used to adjust the distance threshold, which is inspired by [19]. Since $S(x)$ in the dynamic regions become higher, the corresponding distance threshold $R(x)$ will gradually increase by Formula (6). In the static regions, the opposite result appears. The feedback mechanism suppressed the dynamic noise well, but there is an exception. Following the logic above, $S(x)$ in these static regions, where foreground objects pass frequently, also presents higher values. Due to the higher $S(x)$, foreground detection result may be incomplete or void. Therefore, PBAS avoids continuously adjusting threshold to evade this phenomenon. Feedback is only carried out when background sample is updated. As a result, feedback slows down in dynamic region so that many dynamic noises cannot be suppressed in time. To achieve fast response, our threshold adjustment mechanism is continuous. In other words, update and pixel classification will not be considered. To minimize the negative effect, we use the blink pixels proposed by [19] that current detection result $F_t(x)$ is different from the previous result $F_{t-1}(x)$. Meanwhile, a pixel-level 2D map accumulator is defined, note V, which can record blink pixels counts that captured by performing the XOR operation between $F_t(x)$ and previous frame $F_{t-1}(x)$. Owing to blink pixels, threshold feedback in this region, where foreground objects pass frequently, is weakened. SuBSENCE updates $V$ using

$$v(x) = \begin{cases} V(x) + 1, & \text{if } XOR(t) = 1, \\ V(x) - 0.1, & \text{otherwise}. \end{cases} \tag{7}$$

Through experiments, we found there are two drawbacks in blink pixels: (i) the XOR operation is equivalent to frame difference, so borders of foreground objects will be falsely saved. SuBSENSE canceled these parts in XOR that intersect with the post-processed result of current frame. However, borders of the previous frame also remain in the XOR. (ii) Their blink pixels can fail in some sequences such as foreground objects that pass through the dynamic background (canoe and boats sequences), because blinking pixels accumulate all the time in the dynamic background region due to the largeness of $R(x)$. To make blink pixels more perfect, two improvement can be easily implemented: firstly, $F_t(x)$ and $F_{t-1}(x)$ with $5 \times 5$ median filters are used to capture blink pixels; secondly, these regions in XOR that intersect with the post-processed result of current frame and previous frame are canceled.

D. Ghost Suppression

(a) #963 (b) own (c) PBAS (d) SUNSENCE

Figure 3: Dynamic background detection on canoe sequence.

Sample-based methods usually adopt conservative update and neighborhood update with the $1/T$ probability. Conservative policy means that only background pixels are updated, however, the false foreground classification (ghost effect) will always exist. To address this problem, the sample-based methods utilize neighborhood update mechanism in which current pixels are diffused to neighborhood background samples. This mechanism cannot distinguish static foreground from ghost. As a price, static and low-speed foreground objects will be eaten up from the outside. Some methods try to control the different update rate to suppress the neighborhood diffusion, but the effect is not significant. Instead of suppressing neighborhood diffuse in a complex way, abandoning neighborhood updates might be a better solution. Without neighborhood update, other ghost suppressing method is considered.

Literature [34, 35] pointed out a fact that ghosts exist in background and does not exist in current frame, but the foreground is the opposite. Based on this fact, we can study the similarity among the foreground edges, the current frame edges and the background edges. Since the reference background is estimated at the initialization stage, properties of detected foreground objects can be categorized. If the foreground edges are more similar to the current frame edges, it is highly likely that the object exists in current frame, and vice versa. $Exist(k)$ is defined as follows:

$$\begin{cases} 1, & \text{if } SE_C(k) - SE_B(k) > E_{th}, \\ 0, & \text{if } SE_B(k) - SE_C(k) > E_{th}, \end{cases} \tag{8}$$

where $k$ is the $k-th$ connected component, $SE_C(k)$ are pixels sum of foreground edges intersected with current frame edges, $SE_B(k)$ are pixels sum of foreground edges intersected with reference background edges and $E_{th}$ is fixed threshold. $Exist(k) = 1$ indicates that the object exists in current frame and $exist(k) = 0$ indicates that the object exists in reference background. Moreover, three counters $Comf$, $ComC$ and $ComB$ are constructed:

$$Com_{f_t}(x) = \begin{cases} Com_{f_{t-i}}(x) + 1, & \text{if } F_p(x) = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

$$Com_{C_t}(x) = \begin{cases} Com_{C_{t-1}}(x) + 1, & \text{if } Exist(x) = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{10}$$

$$Com_{B_t}(x) = \begin{cases} Com_{B_{t-1}}(x) + 1, & \text{if } Exist(x) = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where $F_p(x)$ is a post-processing version of $F(x)$, $Com_f(x)$ is number of pixels continuously classified as foreground, $Com_B$ is number of objects continuously existing in reference background and $Com_C$ is number of objects continuously existing in current frame. Our update mechanism proposed based on above description is

$$\begin{cases} B_i(x) = I(x), & \text{if } F_p(x) = 0, \\ B_i(x) = I(x), & \text{if } Com_f(x) \geq T_f \wedge Com_B(x) \geq T_B, \end{cases} \tag{12}$$

where $T_f$ and $T_B$ are fixed parameter and $i$ is a random number. Conservative update is still implemented with the $1/T$ probability is still selected and the post-processing version is used as the update mask. Background noises are not added to background samples, which is positive for suppressing dynamic background. If a pixel is continuously detected as foreground and continuously exists in the background, it is immediately updated to the background samples. Note that the update rate $T$ is a fixed parameter.

(a) background (b) current (c) 2700 (d) 2736

Figure 4: Intermittent object detection on abandonedbox sequence. Background object leaves the background region and ghost appears in the original position. When counters reach the own critical value, pixels in the ghost position are immediately updated to the background samples.

E. Post processing

Even if the adaptive feedback mechanism is adopted, the detection results may inevitably involve unwanted noises that reduce the accuracy of the results. So, the 99 median filter is used as the post-processing tool.

## 3. Experiments and results

To properly evaluate the performance of our background subtraction method, a benchmark dataset that contains as many video surveillance scenarios as possible is needed. Thanks to contribution of [36, 37], the ChangeDetection.net (CDnet) dataset offers diverse video scenes in real conditions with corresponding ground-truth masks. CDnet consists of 2012 version and 2014 version, including 11 categories: baseline, camera jitter, dynamic background, intermittent object motion, shadow, thermal, bad weather, low framerate, night videos, point-tilt-zoom (PTZ) and turbulence. Besides, 7 official metrics for evaluating methods are Recall (Re), Specificity (Sp), False Positive Rate (FPR), False Negative Rate (FNR), Percentage of Wrong Classifications (PWC), F-Measure (FM), and Precision (Pr). All the metrics are calculated by true positive (TP), false positive (FP), true negative (TN) and false negative (FN). For specific definitions, please refer to [36, 37]. In our evaluation process, FM is selected as the main evaluation indicator since it is generally accepted and widely used to evaluate the detection quality [19, 38, 39]. Our approach is primarily concerned with Dynamic background and intermittent object motion. So, only CDnet2012 dataset is selected.

A. Parameter Settings

As described above, the method needs to determine some parameters for adapting universal video scenes better. The number of samples $N$, distance thresholds $R_{ini}$ (in reference background estimation phase), distance thresholds $R$ (in foreground segmentation phase), update rate factor $T$, fixed time interval $P$, two critical values of counters $T_f$ and $T_B$, and edge threshold $E_{th}$ are all the parameters that should be discussed. It is noted that $R$ can self-regulate by the adaptive feedback mechanism. So it is only necessary to determine lower bound and upper bound of $R$.

(a) $R_{ini} = 5$: This parameter is only used to estimate reference background. In the update phase, accurate reference background is helpful to obtain the correct background edges, otherwise, it will affect the edge similarity of foreground object. Considering that estimate reference background in CDnet dataset is not difficult, so we roughly determine that $R_{ini} = 5$. For more difficult dataset, $R_{ini} = 5$ can be modified by widely used background initialization performance indicators. Reader can refer to [19, 38, 39].

(b) $N = 30$: The number of background samples. In theory, more background samples are positive for samples multi-modal distribution. This positive effect can approximatively reach saturation when $N$ increases. Note that much more samples only lead slight improvement for detection results.

(c) $P = 4$: The main purpose of increasing sampling interval is to increase the difference between background samples.

(d) $R_{lower} = 20$: The lower bound of decision threshold for foreground segmentation.

(e) $R_{upper} = 40$: The upper bound of decision threshold for foreground segmentation. Threshold feedback is determined by blinking pixels accumulation. Without upper bound, Re and FM are low.

(f) $T = 6$: Background update rate is fixed. Our method cancels neighborhood diffusion; static or low-speed foreground objects are not affected by high update probability.

(g) $T_f = 30$: Counter critical value.

(h) $T_b = 30$: Counter critical value.

(i) $E_{th} = 10$: Edge threshold determines edge similarity.

B. Experimental Results on CDnet2012

In this section, the method performance is evaluated in detail. Dynamic background and intermittent object motion have been targeted for particular attention. The complete evaluation results on dynamic background and intermittent object motion are respectively shown in Table 1 and Table 2.

Table 1. Evaluation results in dynamic background

| | ***Re*** | ***Sp*** | ***FPR*** | ***FNR*** | ***PWC*** | ***Pr*** | ***FM*** |
|---|---|---|---|---|---|---|---|
| overpass | 0.9381 | 0.9997 | 0.0003 | 0.0619 | 0.1116 | 0.9777 | 0.9575 |
| fountain01 | 0.8640 | 0.9993 | 0.0007 | 0.1360 | 0.0778 | 0.5188 | 0.6483 |
| fountain02 | 0.9537 | 0.9993 | 0.0007 | 0.0463 | 0.0758 | 0.7572 | 0.8441 |
| fall | 0.9435 | 0.9941 | 0.0059 | 0.0565 | 0.6759 | 0.7438 | 0.8318 |
| boats | 0.6167 | 0.9999 | 0.0001 | 0.3833 | 0.2507 | 0.9741 | 0.7553 |
| canoe | 0.9446 | 0.9991 | 0.0009 | 0.0554 | 0.2858 | 0.9739 | 0.9590 |
| **overall** | 0.8768 | 0.9986 | 0.0014 | 0.1232 | 0.2463 | 0.8243 | 0.8327 |

Table 2. Evaluation results in dynamic background

| | ***Re*** | ***Sp*** | ***FPR*** | ***FNR*** | ***PWC*** | ***Pr*** | ***FM*** |
|---|---|---|---|---|---|---|---|
| aband.Box | 0.9705 | 0.9934 | 0.0066 | 0.0295 | 0.7658 | 0.8821 | 0.9242 |
| parking | 0.4883 | 0.9842 | 0.0158 | 0.5117 | 5.4150 | 0.7214 | 0.5824 |
| sofa | 0.7031 | 0.9966 | 0.0034 | 0.2969 | 1.6257 | 0.9032 | 0.7907 |
| streetLight | 0.9157 | 0.9999 | 0.0001 | 0.0843 | 0.4158 | 0.9985 | 0.9553 |
| tramstop | 0.9762 | 0.9981 | 0.0019 | 0.0238 | 0.5789 | 0.9914 | 0.9837 |
| wint.Dri.way | 0.7448 | 0.9879 | 0.0121 | 0.2552 | 1.3922 | 0.3170 | 0.4447 |
| **overall** | 0.7998 | 0.9934 | 0.0067 | 0.2002 | 1.6989 | 0.8023 | 0.7802 |

Table 3. Complete results for proposed method on 2012 CDnet dataset

| Category | ***Re*** | ***Sp*** | ***FPR*** | ***FNR*** | ***PWC*** | ***Pr*** | ***FM*** |
|---|---|---|---|---|---|---|---|
| baseline | 0.9325 | 0.9962 | 0.0038 | 0.0675 | 0.6671 | 0.886 | 0.9077 |
| camera jitter | 0.7958 | 0.9757 | 0.0243 | 0.2043 | 2.9965 | 0.7165 | 0.7156 |
| dynamic back. | 0.8768 | 0.9986 | 0.0014 | 0.1232 | 0.2463 | 0.8243 | 0.8327 |
| interm. obj. mot. | 0.7998 | 0.9934 | 0.0067 | 0.2002 | 1.6989 | 0.8023 | 0.7802 |
| shadow | 0.9009 | 0.9891 | 0.0109 | 0.0991 | 1.4924 | 0.8090 | 0.8500 |
| thermal | 0.7228 | 0.9962 | 0.0038 | 0.2772 | 1.3445 | 0.9023 | 0.7814 |
| **overall** | 0.8381 | 0.9915 | 0.0085 | 0.1619 | 1.4076 | 0.8234 | 0.8113 |

Table 4. Compared with some state-of-the-art sample-based methods

| | $\mathbf{FM}_{Overall}$ | $\mathbf{FM}_{baseline}$ | $\mathbf{FM}_{cam.jit.}$ | $\mathbf{FM}_{dyn.bg}$ | $\mathbf{FM}_{int.obj.mot}$ | $\mathbf{FM}_{shadow}$ | $\mathbf{FM}_{thermal}$ |
|---|---|---|---|---|---|---|---|
| proposed | 0.811 | 0.908 | 0.716 | *0.833* | **0.780** | 0.850 | 0.781 |
| PAWCS [38] | **0.858** | 0.940 | *0.814* | **0.894** | *0.776* | 0.891 | **0.832** |
| SuBSENSE[19] | *0.826* | **0.950** | **0.815** | 0.818 | 0.657 | *0.899* | *0.817* |
| WeSamBE[39] | 0.820 | *0.941* | 0.798 | 0.744 | 0.739 | **0.900** | 0.796 |
| PBAS [18] | 0.753 | 0.924 | 0.722 | 0.683 | 0.575 | 0.860 | 0.756 |
| ViBe+ [16] | 0.722 | 0.871 | 0.754 | 0.720 | 0.509 | 0.815 | 0.665 |

Note that bold entries indicate the best results and italics entries indicate the second-best results.

The SDIES adopts modified blink pixel and background dynamic indicator and the ability that suppresses dynamic background is significantly improved. Besides, F-measure increases about 13 over other methods (without PAWCS [38]). The post-processing version of F is used to capture blink pixels, which effectively solves the over-segmentation for boats and canoe sequences. In addition, the post-processing version of F is used as the update mask, which is positive for suppressing dynamic noise. This is the main reason that F-measure increases for dynamic background category.

The SDIES cancels neighborhood diffusion and uses three counters to realize conservative background update. That makes it possible to detect intermittent moving objects well and then F-measure increases about 20 over other methods. Neighborhood diffusion can suppress ghosts, but the process is slow. Our judging condition is that pixel is continuously detected as a ghost for 30 times. So, ideally, it only takes one second. This can be well represented on tramstop and abandonedbox sequences. The disadvantage of Neighborhood diffusion is that it damages the completeness of static foreground objects. Fortunately, the neighborhood diffusion is canceled. There are fewer wrong pixel classifications while ensuring complete foreground objects. Rapid ghost suppression and complete foreground objects are the main reasons why F-measure increases for intermittent object motion category.

Besides, we selecte some state-of-the-art sample-based methods to compare with the proposed method. As shown in Table 4, F-measure is superior in dynamic background and intermittent object motion. In particular, FMinterm.obj.mo exceeds all other algorithms. However, the method does not only adapt these two scenarios; Table 3 indicates that the method can actually adapt all scenarios. Although the advantages in other scenarios are not so prominent, the SDIES meets the universal requirements. The subjective evaluation results are shown in Figure 5.

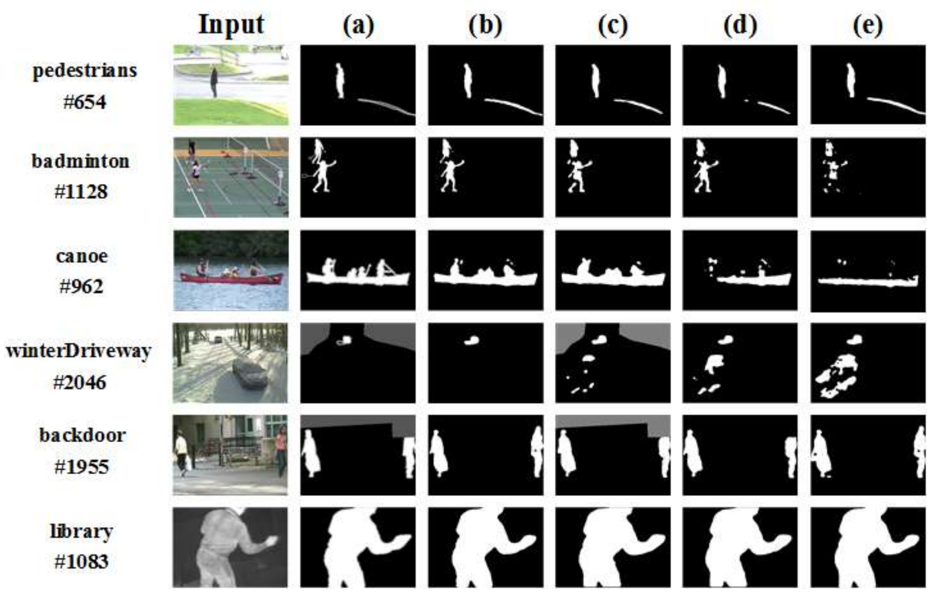


Figure 5: Foreground segmentation results on the CDnet2012 datasets. Column (a) are ground-truth, (b) are proposed results, (c) are PAWCS results [38], (d) are SuBSENSE results [35], (e) are PBASresults [18].

C. Processing Speed

The Code is written in MATLAB R2016b on PC with an Intel Core I5 CPU processor and 4GB memory. The average processing speed of SDIES is about 2 FPS. This code is the original version without any optimization, which could not meet the real-time requirements. In the future, the authors would like to try to turn it into C++ version.

## 4. Conclusion

The proposed method mainly focuses on dynamic background and intermittent object motion. Improved blinking pixels and sample dynamic indicators are used to better control adaptive threshold feedback. Instead of neighborhood update, using the estimated reference background to calculate the existence property of foreground objects could be the very way that effectively eliminates ghosts produced by intermittent object motion and ensures the complete foreground objects as well. It is noted that some scenes, such as those with hard shadow, complex background texture features and similar color features between background and foreground, have negative effect on edge similarity. In addition, adaptive threshold feedback can take high time complexity. In the future work, we devote to find a more reasonable measurement for edge similarity and try to optimize the code by C++ and parallel mode in order to meet the real-time requirements.

## Acknowledgements

This work is supported by Foundation of Chongqing Municipal Key Laboratory of Institutions of Higher Education ([2017]3), Foundation of Chongqing Development and Reform Commission (2017[1007]), and Foundation of Chongqing Three Gorges University.

## References

[1] J. S. Kim, H. Y. Dong, Y. H. Joo, *Fast and robust algorithm of tracking multiple moving objects for intelligent video surveillance systems*, IEEE Transactions on Consumer Electronics, 57 (2011), 1165-1170.

[2] J. H. Jean , F. L. Lian, *Robust visual servo control of a mobile robot for object tracking Using shape parameters*, IEEE Transactions on Control Systems Technology, 20 (2012), 1461-1472.

[3] S. Thüring, J. Herwig, A. Schmitt, *Silhouette-based motion capture for interactive VR-systems including a rear projection screen*, Computer Animation and Virtual Worlds, 16 (2010), 245-257.

[4] T. Bouwmans, *Traditional and recent approaches in background modeling for foreground detection: An overview*, Computer Science Review, 2014 (11), 31-66.

[5] Y. Benezeth, P. M. Jodoin, B. Emile, *Review and evaluation of commonly-implemented background subtraction algorithms*, IEEE International Conference on Pattern Recognition, 2012, 1-4.

[6] C. Wren, A. Azarbayejani, T. Darrell, *Pfinder: real-time tracking of the human body*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), 780-785.

[7] C. Stauffer, W.E.L. Grimson, *Adaptive background mixture models for real-time tracking*, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, 252(2).

[8] C. Stauffer, W.E.L. Grimson, *Learning patterns of activity using real-time tracking*, IEEE Trans on Pami, 22 (2000), 747-757.

[9] S. Varadarajan, H. Wang, P. Miller, *Regularised region-based mixture of Gaussians for dynamic background modelling*, IEEE International Conference on Advanced Video and Signal Based Surveillance, 2014, 19-24.

[10] S. Varadarajan, P. Miller, H. Zhou, *Region-based mixture of Gaussians modelling for foreground detection in dynamic scenes*, Pattern Recognition, 48 (2015), 3488-3503.

[11] O. Barnich, D. M. Van, *Vibe: a universal background subtraction algorithm for video sequences*, IEEE Transactions on Image Processing, 20(2011), 1709.

[12] H. Wang, D. Suter, *SACON: A consensus based model for background subtraction*, Schools and Disciplines, 2005.

[13] H. Wang, D. Suter, *A consensus-based method for tracking: modelling background scenario and foreground appearance*, Pattern Recognition, 40 (2007), 1091-1105.

[14] B. Gu, K. Song, D. Qiu, *Moving object detection based on improved ViBe Algorithm*, International Journal of Smart Home, 9 (2015), 225-232.

[15] X. Zhou, X. Liu, A. Jiang, *Improving video segmentation by fusing depth cues and the visual background extractor (ViBe) Algorithm*, Sensors, 17 (2017).

[16] M. Droogenbroeck, O. Paquot, *Background subtraction: Experiments and improvements for ViBe*. IEEE Conference on Computer Vision and Pattern Recognition, (2012), 32-37.

[17] S. Brutzer, B. Hoferlin, G. Heidemann, *Evaluation of background subtraction techniques for video surveillance*, IEEE Conference on Computer Vision and Pattern Recognition, (2011), 1937-1944.

[18] M. Hofmann, P. Tiefenbacher, G. Rigoll, *Background segmentation with feedback: The Pixel-Based Adaptive Segmenter*, IEEE Conference on Computer Vision and Pattern Recognition, (2012), 38-43.

[19] P. L. St-Charles, G. A. Bilodeau, R. Bergevin, *SuBSENSE: a universal change detection method with local adaptive sensitivity*, IEEE Transactions on Image Processing, 24 (2015), 359-373.

[20] A. Varghese, G. Sreelekha, *Sample-based integrated background subtraction and shadow detection*, IPSJ Transactions on Computer Vision and Applications, 9 (2017), 25.

[21] Y. Ding, W. Li, T. Fan, *Robust moving object detection under complex background*, Computer Science and Information Systems, 7 (2010), 201-210.

[22] Z. Zhao, T. Bouwmans, X. Zhang, *A fuzzy background modeling approach for motion detection in dynamic backgrounds*, International Conference on Multimedia and Signal Processing, (2012), 177-185.

[23] P. Chiranjeevi, S. Sengupta, *Detection of moving objects using multichannel kernel fuzzy correlogram based background subtraction*, IEEE Transactions on Cybernetics, 44(2014), 870-881.

[24] Y. Xie, W. Zhang, Y. Qu Y, *Discriminative subspace learning with sparse representation view-based model for robust visual tracking*, Pattern Recognition, 47 (2014), 1383-1394.

[25] D. Farcas, C. Marghes, T. Bouwmans, *Background subtraction via incremental maximum margin criterion: a discriminative subspace approach*, Machine Vision and Applications, 23 (2012), 1083-1101.

[26] S. Javed, A. Sobral, T. Bouwmans, *OR-PCA with dynamic feature selection for robust background subtraction*, The Acm/sigapp Symposium on Applied Computing, (2015).

[27] B. Wohlberg, R. Chartrand, J. Theiler, *Local principal component pursuit for nonlinear datasets*, International Conference on Acoustics, Speech, and Signal Processing, (2012).

[28] C. David, V. Gui, G. Alexa, *Foreground/background segmentation with learned dictionary*, International Conference on Circuits, Systems and Signals, (2009), 197-201.

[29] R. Sivalingam, A. Souza, M. Bazakos, *Dictionary learning for robust background modeling*, IEEE International Conference on Robotics and Automation, (2011), 4234-4239.

[30] C. R. Wren, F. Porikli, *Spectral similarity for object detection*, IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, (2005).

[31] W. Wang, D. Chen, W. Gao, *Modeling background from compressed video*, IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, (2005), 161-168.

[32] A. S. Jalal, V. Singh, *A robust background subtraction approach based on daubechies complex wavelet transform*, Advances in Computing and Communications, (2011), 516-524.

[33] D. Bloisi, L. Iocchi, *Independent multimodal background subtraction*, Processing of the Third International Conference on Computational Modeling of Objects Presented in Images: Fundamentals, Methods and Applications, 413 (2012), 39-44.

[34] R. H. Evangelio, T. Sikora, *Complementary background models for the detection of static and moving objects in crowded environments*, IEEE International Conference on Advanced Video and Signal-Based Surveillance, (2011), 71-76.

[35] T. Kryjak, M. Komorkiewicz, M. Gorgon, *Real-time foreground object detection combining the PBAS background modelling algorithm and feedback from scene analysis module*, International Journal of Electronics and Telecommunications, 60 (2014), 53-64.

[36] N. Goyette, P. M. Jodoin, F. Porikli, *Changedetection.net: A new change detection benchmark dataset*, IEEE International Conference on Pattern Recognition, (2012), 1-8.

[37] Y. Wang, P. M. Jodoin, F. Porikli, *CDnet 2014: An expanded change detection benchmark dataset*, IEEE International Conference on Pattern Recognition, (2014), 393-400.

[38] P. L. St-Charles, G. A. Bilodeau, R. Bergevin, *Universal background subtraction using word consensus models*, IEEE Transactions on Image Processing, 25 (2016), 4768C4781.

[39] S. Jiang, X. Lu. *WeSamBE: A weight-sample-based method for background subtraction*, IEEE Transactions on Circuits and Systems for Video Technology, 99 (2017), 1-11.

[40] T. Bouwmans, L. Maddalena, A. Petrosino, *Scene background initialization: a taxonomy*, Pattern Recognition Letters, (2017), 96.

Accepted: 27.02.2019