Active Attentional Sampling for Speed-up of Background Subtraction

Hyung Jin Chang, Hawook Jeong and Jin Young Choi
Perception and Intelligence Lab., School of EECS, ASRI,
Seoul National University, Seoul, Korea
{changhj,hwjeong,jychoi}@snu.ac.kr

Abstract

In this paper, we present an active sampling method to speed up conventional pixel-wise background subtraction algorithms. The proposed active sampling strategy is designed to focus on attentional region such as foreground regions. The attentional region is estimated by detection results of previous frame in a recursive probabilistic way. For the estimation of the attentional region, we propose a foreground probability map based on temporal, spatial, and frequency properties of foregrounds. By using this foreground probability map, active attentional sampling scheme is developed to make a minimal sampling mask covering almost foregrounds. The effectiveness of the proposed active sampling method is shown through various experiments. The proposed masking method successfully speeds up pixel-wise background subtraction methods approximately 6.6 times without deteriorating detection performance. Also real-time detection with Full HD video is successfully achieved through various conventional background subtraction algorithms.

1. Introduction

Background subtraction is a process which aims to segment moving foreground objects from a relatively stationary background[16]. Recently pixel-based probabilistic model methods [2, 8, 19, 21, 6, 10] gained lots of interests and have shown good detection results. There have been many improvements in detection performance for these methods under various situations, but the computational time still takes too much time. Computation time reduction issue is getting more important in a systematic view, because the background subtraction is generally considered as a low level image processing task, which needs to be done with little computation, and video sizes are getting bigger.

To reduce computation time of background subtraction methods, several approaches have been studied. The first type of approach is based on optimizing algorithms. Although the Gaussian mixture model (GMM) scheme proposed by Stauffer and Grimson[19] works well for various environments, it suffers from slow learning rates and heavy computational load for each frame[10]. Lee [14] makes the convergence fast by using a modified schedule that gradually switches between two stage learning schemes. Zivkovic[21] achieved a significant speed-up by formulating a Bayesian approach to select the required number of Gaussian modes for each pixel in the scene. Gorur[10] modified Zivkovic's method[21] by windowed weight update that minimizes floating point computations.

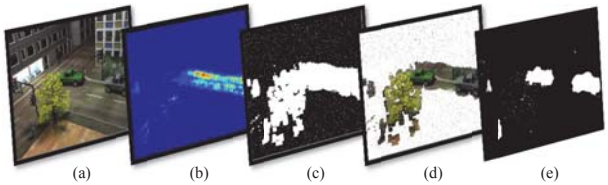


Figure 1. Background subtraction by active attentional sampling mask. (a) Input video image (b) Foreground probability map (c) Active attentional sampling mask (d) Sampled pixels (e) Foreground detection result

The second type of approach is using parallel computation. Multi-core processors in a parallel form, using the OpenMP system are applied for speed-up[20]. Also Graphical Processing Units (GPUs) are used to achieve real-time performance[5] with computationally heavy algorithms. Pham *et al.*[18] perform real time detection even in full HD video using GPU. They could successfully achieve speed-up, but special hardware resources are required.

A selective sampling based speed-up method is the third type of approach. Park *et al.*[17] proposed a hierarchical quad-tree structure to decompose an input image. Using the image decomposition, they could achieve the computational complexity reduction. However, their algorithm may miss small objects because they randomly sample from a relatively large region. Kim *et al.*[13] presented a sampling mask designing method which can be readily applied to many existing object detection algorithms. Lee *et al.*[15] proposed a two-level pixel sampling method. Their algorithm provides accurate segmentation results without flickering artifacts. Kim *et al.*[13] and Lee *et al.*[15] use com-

978-1-4673-1228-8/12/$31.00 ©2012 IEEE

pactly designed grid pattern masks to detect small objects, but these grid patterns still cause redundant operations.

In this paper, we propose a new method of the third type of approach (sampling mask approach) which can be utilized together with the other two approaches. We aim to find an active attentional sampling solution which can be generally applied to most conventional background subtraction methods. We design a foreground probability map based on temporal, spatial and frequency properties of the foreground region. Using previous foreground detection result, the foreground probability map is updated. A sequential coarse-to-fine approach, which involves sparse random sampling and filling in a space in attentional region according to the probability map, achieves a very significant reduction in computation time without degrading the detection performance. Figure 1 illustrates the process of the proposed algorithm. By combining with conventional background subtraction methods, our method makes these methods even be able to handle full HD videos in real-time.

2. Overview

2.1. Motivation

We imitate the selective attention mechanism of human[11], where previously recognized results are reflected in the focusing position of current frame. When a guard monitors a CCTV camera, he/she does not concentrate on whole of the image since he/she has empirically learned that the video image can be categorized into background region, unimportant dynamic scene region and important moving object appearing region. Then he/she takes his/her attention to the regions which have moving object appearing intentionally and does a sparse scanning to the other regions such as background or dynamic region. The key idea of proposed approach is to simulate this selective attention scheme.

In general, most pixels from surveillance video are background region, and foreground region takes very small portion in both spatially and temporally. We have measured a percentage of the foreground area of commonly used data set in background subtraction papers. The tested data sets are Wallflower[1], VSSN2006[2], PETS2006[3], AVSS2007 i-LIDS challenge[4], PETS2009[5] and SABS[3][6]. As we can see in Table1, the proportions of foreground regions are very small. Hence, if background substraction can be focused on foreground area, necessary calculation would be reduced significantly. In this paper we try to find attentional region in a current frame considering foreground region detected in a previous frame.

Data Set	# of tested frames	Mean(%)	Std.
Wallflower	7553	5.03	6.25
VSSN2006	16074	2.30	1.13
PETS2006	41194	1.04	0.26
AVSS2007	33000	3.36	1.02
PETS2009	2581	5.48	1.58
SABS	6400	2.42	1.83
	Average	**2.42**	**1.18**

Table 1. Statistical foreground region ratio of several widely used datasets. Only 2.42% of total pixels are foreground pixels.

2.2. Overall Scheme of Proposed Algorithm

Figure 2 shows the overall scheme of the proposed method. To get active sampling mask for background substraction, we use three properties of foreground; temporal, spatial, frequency properties. The temporal property is that a pixel is more likely to be a part of the foreground region if it has been a foreground pixel previously. The spatial property is that a pixel has a high probability of being a foreground pixel if its surrounding pixels are of the foreground. The probability is proportional to the number of surrounding foreground pixels. This spatial ergodic property was also used in [12][16] for background modeling. The frequency property is that if foreground/background index of a pixel is changed too frequently, then the pixel is more likely to be a noise or dynamic background region. So the probability of being a stable foreground region is low. Based on the properties, we make a foreground probability map P_{FG} (described in Section 3).

The active sampling strategy is updated in every frame according to the foreground probability map $(P_{FG})^{t-1}$. The strategy is composed of three sampling strategies such as *randomly scattered sampling*, *spatially expanding importance sampling*, and *surprise pixel sampling*, which are performed sequentially. We make the sampling mask $\mathbf{M}^t$ at every frame (described in Section 4). Using sampling mask $\mathbf{M}^t$, selective pixel-wise background subtraction is performed, only for the pixels of $\mathbf{M}^t(n) = 1$ where n indicates the pixel index. This sampling mask can be combined with any kind of pixel-wise background subtraction methods.

The background subtraction task finds a sequence of detection masks $\{D^1, ..., D^T\}$ using a sequence of video frames $\{I^1, ..., I^T\}$ and sampling mask $\{\mathbf{M}^1, ..., \mathbf{M}^T\}$. Each video image I^t, sampling mask $\mathbf{M}^t$ and detection mask D^t are composed of N pixels $\{I^t(1), ..., I^t(N)\}$,

[1] http://research.microsoft.com/~jckrumm/wallflower/testimages.htm

[2] http://mmc36.informatik.uni-augsburg.de/VSSN06$_$OSAC

[3] http://www.cvg.rdg.ac.uk/PETS2006/data.html

[4] http://www.eecs.qmul.ac.uk/~andrea/avss2007$_$ss$_$challenge.html

[5] http://www.cvg.rdg.ac.uk/PETS2009/a.html

[6] http://www.vis.uni-stuttgart.de/index.php?id=sabs

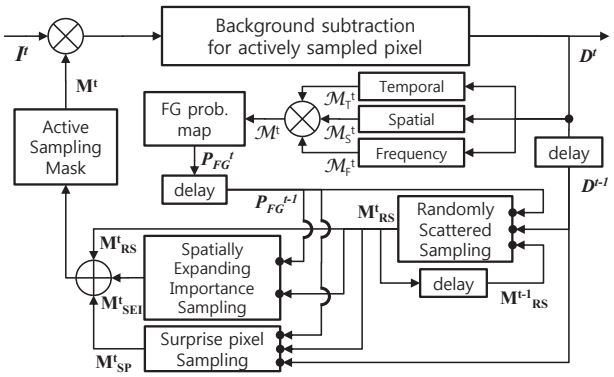


Figure 2. Overall scheme of the proposed algorithm.

$\{\mathbf{M}^t(1), ..., \mathbf{M}^t(N)\}$ and $\{D^t(1), ..., D^t(N)\}$ respectively. All the masks are binary masks. In this paper, selective pixel-wise background subtraction is performed, only for the pixels of $\mathbf{M}^t(n) = 1$. The detection mask at pixel n shall be denoted with the symbol $D(n)$: $D(n) = 0$ if pixel n belongs to the background and $D(n) = 1$ if it belongs to the foreground.

There are several empirical and theoretical results suggesting that use of data collected in early stages can be significantly more efficient to guide the selection of new samples [7, 9]. Conventional background subtraction algorithms are based on passive sampling. The collection of sample points is chosen independent to the labels, and a prior probability distribution of foreground is assumed uniform. So, in order to detect unexpected foreground, the sampling becomes a full search regardless of previous observations.

On the other hand, the active sampling scenario allows the sample location to be chosen using the information collected up to that point[4]. So the sampling becomes adaptive and flexible[7]. However, a prior information about the dependency between samples and labels are necessary to design the sampling strategy. In the following sections, we describe a way how to design the sampling strategy using the properties of attentional foreground region.

3. Foreground Probability Map Generation

3.1. Estimation of Foreground Properties

Estimation models are proposed to measure the temporal, spatial, and frequency properties of each pixel. The three property measures are referred to as $\{\mathcal{M}_T, \mathcal{M}_S$, and $\mathcal{M}_F\}$. The temporal property measure $\mathcal{M}_T$ is estimated by the recent history of detection results. The spatial property $\mathcal{M}_S$ is estimated by the number of foreground pixels around each pixel. The frequency property $\mathcal{M}_F$ is estimated by the ratio of detection result flipping over a period of time. All estimation models are updated by a running average method, with learning rates α_T, α_F and α_S (all learning rates are between 0 and 1). The estimation models for the measures of the properties are given in the following.

- **Temporal property** $\mathcal{M}_T$: At each location n, a recent history of detection mask results at that location are averaged to estimate the property.

$$\mathcal{M}_T^t(n) = (1 - \alpha_T)\mathcal{M}_T^{t-1}(n) + \alpha_T D^t(n). \quad (1)$$

  As the value of $\mathcal{M}_T^t(n)$ comes close to 1, the possibility of foreground appearance at the pixel is high.

- **Spatial property** $\mathcal{M}_S$: Detection results of nearby pixels are used to measure the spatial coherency of each pixel n.

$$\mathcal{M}_S^t(n) = (1 - \alpha_S)\mathcal{M}_S^{t-1}(n) + \alpha_S s^t(n), \quad (2)$$
$$(s^t(n) = \frac{1}{w^2} \sum_{i \in \mathcal{N}(n)} D^t(i)),$$

  where $\mathcal{N}(n)$ denotes a spatial neighborhood around pixel n ($w \times w$ square region centered at n). $\mathcal{M}_S^t(n)$ closer to 1 means high probability of being a part of the foreground.

- **Frequency property** $\mathcal{M}_F$: If detection results have been changed twice during previous three frames, we consider it as a clue of dynamic scene.

$$\mathcal{M}_F^t(n) = (1 - \alpha_F)\mathcal{M}_F^{t-1}(n) + \alpha_F f^t(n), \quad (3)$$
$$f^t(n) = \begin{cases} 1 & (D^{t-2}(n) \neq D^{t-1}(n)) \\ & \&(D^{t-1}(n) \neq D^t(n)) \\ 0 & \text{otherwise}. \end{cases}$$

  where $f^t(n)$ denotes a frequently changing property at n. Unlike the other measures, the pixel n has a high probability of being a foreground, as the value $\mathcal{M}_F^t(n)$ is close to 0.

3.2. Foreground Probability Map: P_{FG}

By estimating the three foreground properties, we get the three measurements, $\mathcal{M}_T, \mathcal{M}_S$, and $\mathcal{M}_F$. Every measurement has a value between 0 and 1. So we define the foreground probability for a pixel n at frame t as

$$P_{FG}^t(n) = \mathcal{M}_T^t(n) \times \mathcal{M}_S^t(n) \times (1 - \mathcal{M}_F^t(n)). \quad (4)$$

The foreground probability map P_{FG}^t is a composition of $\{P_{FG}^t(n)\}_{n=1}^N$.

4. Active Sampling Mask Generation

The sampling mask $\mathbf{M}^t$ is obtained by a combination of three masks by a pixel-wise 'OR' operation ($\oplus$) as

$$\mathbf{M}^t = \mathbf{M}_{RS}^t \oplus \mathbf{M}_{SEI}^t \oplus \mathbf{M}_{SP}^t, \quad (5)$$

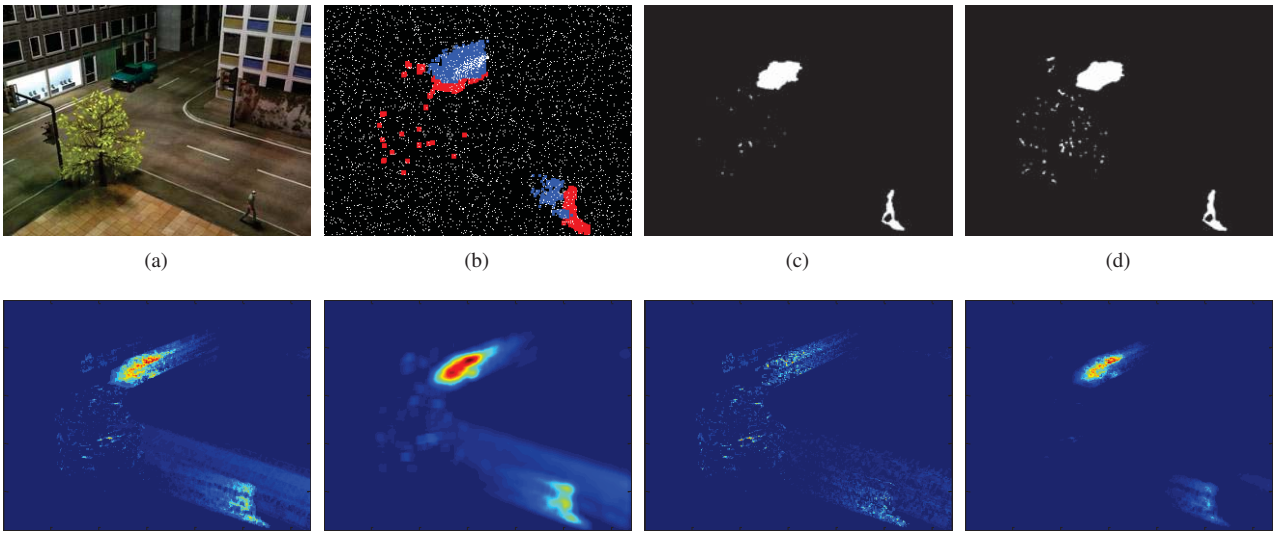


Figure 3. Active attentional mask generation by foreground probability map. (a) is a current input video image. (b) shows the active attentional mask used for background subtraction. The white points are randomly scattered sampling mask $\mathbf{M}^t_{RS}$. The blue pixels represent $\mathbf{M}^t_{SEI}$ and the red regions are $\mathbf{M}^t_{SP}$. As we can see in (b), most of mask $\mathbf{M}^t$ become zeros. The mask, whose redundancy is removed, optimizes the necessary computational load. (c) Foreground detection result by GMM method [19] with the active mask. (d) Foreground detection result by GMM method [19] without the mask. (e) Temporal property $\mathcal{M}^t_T$. (f) Spatial property $\mathcal{M}^t_S$. (g) Frequency property $\mathcal{M}^t_F$. (h) Foreground probability map P^t_{FG}

where $\mathbf{M}^t_{RS}$, $\mathbf{M}^t_{SEI}$ and $\mathbf{M}^t_{SP}$ are sampling masks of *randomly scattered sampling* ($\mathbf{S}_{RS}$), *spatially expanding importance sampling* ($\mathbf{S}_{SEI}$) and *surprise pixel sampling* ($\mathbf{S}_{SP}$) respectively.

At each sampling stage, the sampling masks are generated based on the foreground probability map P_{FG} and foreground detection result D. We design the active sampling strategies as

$$\mathbf{M}^t_{RS} = \mathbf{S}^t_{RS}(\mathbf{M}^{t-1}_{RS}, D^{t-1}, P^{t-1}_{FG}), \quad (6)$$

$$\mathbf{M}^t_{SEI} = \mathbf{S}^t_{SEI}(\mathbf{M}^t_{RS}, P^{t-1}_{FG}), \quad (7)$$

$$\mathbf{M}^t_{SP} = \mathbf{S}^t_{SP}(\mathbf{M}^t_{RS}, D^{t-1}, P^{t-1}_{FG}). \quad (8)$$

Figure 3 shows the foregorund property measurements, corresponding sampling mask $\mathbf{M}^t$ and foreground detection results with and without $\mathbf{M}^t$. In the following, we describe the details on the sampling strategies in (6), (7), and (8).

4.1. Randomly Scattered Sampling

First, $100 \times \rho\%$ (usually ρ 0.05 to 0.1) pixels of the entire pixels are selected through randomly scattered sampling. Uniform random sampling approximates that every pixel is checked probabilistically on average once among $1/\rho$ frames. The number of random samples N_s is ρN. This number is constant for all frames. However, some of the random points generated in the previous frames are worth to be preserved. The determination of these points are based on the amount of information measured by the foreground probability P^{t-1}_{FG}. A sample point n which was $\mathbf{M}^{t-1}_{RS}(n) = D^{t-1}(n) = 1$ is used again in current frame($\mathbf{M}^t_{RS}(n) = 1$). Therefore, the number of reused samples N_{reuse} changes adaptively. Then, $N_s - N_{reuse}$ samples are resampled randomly across the entire image.

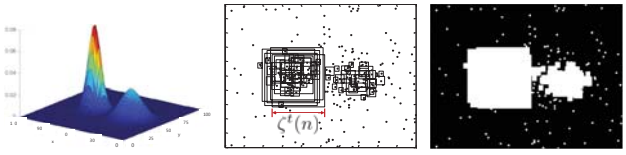
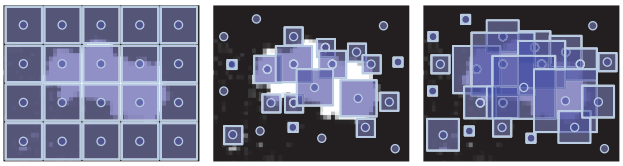


Figure 4. Spatially expanding importance sampling mask $\mathbf{M}_{SEI}$ generation by foreground probability map P_{FG}. (a) is P_{FG}. (b) For each point of M_{RS}, the spatially expanding region width ζ_s is calculated. (c) The mask $\mathbf{M}_{SEI}$ is generated by setting all the inside points of the square to 1.

4.2. Spatially Expanding Importance Sampling

The randomly sampled mask $\mathbf{M}^t_{RS}$ is too sparse to construct a complete foreground region and might miss small objects. It is therefore necessary to fill the space between sparse points in the foreground region. In order to fill the space, we develop an appropriate importance sampling solution focusing only on necessary region compactly.

Conventional importance sampling[1] draws samples densely where the importance weight is high. In our case,



(a) (b) (c)

Figure 5. (a) $r^t = 1, k = 1$. (b) $r^t = P^t_{FG}, k = 1$. (c) $r^t = P^t_{FG}, k = \sqrt{3}$.

the sampling mask should cover all of the foreground pixels and so the dense sampling is not enough in the foreground region. To solve this full coverage sampling problem, we propose a *spatially expanding importance sampling* method which expands the sampling area proportional to the importance weight at every point of $\mathbf{M}^t_{RS} = 1$ as shown in Figure 4. The shape of the expanded region is a square with width of ζ^t which depends on the importance weight at the i^{th} randomly scattered sample. Even though the square regions are overlapped, they are depicted by one region with $\mathbf{M}^t_{SEI} = 1$ as shown in Figure 4.

If the proposal distribution is assumed as an uniform distribution, importance weight of each randomly scattered sample i (where $\mathbf{M}^t_{RS}(i) = 1$) becomes $r^t(i) = P^t_{FG}(i)$. Proportional to $r^t(i)$, we expand the sampling region $\mathcal{N}(i)$ with size of $\zeta^t(i) \times \zeta^t(i)$ centered at pixel i, i.e.

$$\mathbf{M}^t_{SEI}(\mathcal{N}(i)) = 1. \tag{9}$$

The spatially expanding width $\zeta^t(i)$ is determined as

$$\zeta^t(i) = round(r^t(i) \times \omega_s), \tag{10}$$

$$\omega_s = k\sqrt{N/N_s}. \tag{11}$$

ω_s is an expanding constant with parameter k (usually k is $\sqrt{3}$ or $\sqrt{5}$). Figure 5 shows how ω_s is designed and the effect of the parameter k. As shown in Figure 5(a), the ω_s with $k = 1$ and $r^t = 1$ implies a width of one square under an assumption that the image is equally decomposed into N_s squares centered at regularly distributed N_s samples. However, in actual situation, the N_s samples are not distributed regularly and most of r^t are less than 1. So the sampling mask $\mathbf{M}^t_{SEI}$ can not cover the estimated foreground region compactly as shown in Figure 5(b). The parameter k (larger than 1) expands the sampling masks so that the masks cover the foreground region compactly (Figure 5(c)). As we can see in Figure 3(b), high foreground probability regions are widely sampled and most of $\zeta^t(n)$ are 0 in low probability region.

4.3. Surprise Pixel Sampling Mask

Even if we estimate the foreground probability correctly, the foreground detection still has unpredictability intrinsically. Abnormal foreground is caused by spontaneousness. For example, a person or a car suddenly appears from a new direction, or a thief enters into a restricted area. These surprisingly appearing moving objects should be detected successfully. In addition, rarely appearing very fast moving objects could be lost, because the spatially expanded region may not be wide enough.

The randomly scattered samples become important when capturing these unpredictable cases. A pixel is defined as a *surprise pixel* if it was foreground in the previous frame even though its foreground probability is small. Because the foreground object is not expected to exist there, the observation of foreground pixel is very surprising. So by widening the sampling area around the pixel in a current frame can find new foreground pixels. For pixel i (where $\mathbf{M}^t_{RS}(i) = 1$), the *surprise pixel* index $\xi^t(i)$ is given by

$$\xi^t(i) = \begin{cases} 1 & (P^{t-1}_{FG}(i) < \theta^{t-1}_{th}) \& (D^{t-1}(i) = 1) \\ 0 & \text{otherwise}. \end{cases} \tag{12}$$

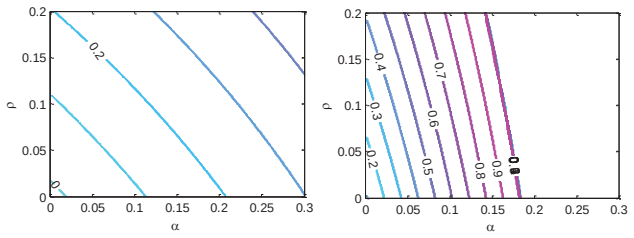
where $\theta^{t-1}_{th} = max(P^{t-1}_{FG}/\omega_s)$. Surprise pixel sampling mask is generated as $\mathbf{M}^t_{SP}(\mathcal{N}(i)) = 1$ for $\mathcal{N}(i)$ region ($\omega_s \times \omega_s$ region centered at i if $\xi^t(i) = 1$).

5. Computational Efficiency Boundary

We calculate a computational efficiency of the proposed method (C_P) comparing to the conventional full search method (C_F). α and α_{std} imply an average ratio (from 0 to 1) of foreground pixels and its standard deviation in a video, respectively. β is a computational complexity ratio of each computation block (such as $P_{FG}, \mathbf{M}^t_{RS}, \mathbf{M}^t_{SEI}, \mathbf{M}^t_{SP}$ generation) of proposed method comparing to original detection method. β_{max} and β_{min} are the largest and smallest value, respectively. Other parameters (ρ and k) are described above. Due to space limitation, we shall omit deriving the efficiency boundary. We will release a technical note with this paper online for completeness. The derived efficiency boundary is

$$\begin{aligned} &(\alpha - \alpha_{std})\{\beta_{min} + (1-\rho)(1+\beta_{min})\} + \rho(1+2\beta_{min}) \\ &< \frac{C_P}{C_F} < (\alpha + \alpha_{std})k^2\{\beta_{max} + (1-\rho)(1+\beta_{max})\} \\ &+ \rho(1+2\beta_{max}). \end{aligned} \tag{13}$$

Figure 6 is a simulated result of efficiency boundary. We have validated the analysis result (13) through actual experimental values. In our implementation GMM[19] method and SABS dataset [3] is used with $\beta_{min} = 0.03, \beta_{max} = 0.33, k = \sqrt{3}$ and $\rho = 0.05$. In this case, actual C_P/C_F is 0.25 which is between lower bound (0.06) and upper bound (0.29) of analysis (13).



(a) (b)

Figure 6. Derived efficienty bound of C_P/C_F. (a) is a lower bound and (b) is a upper bound.

6. Experimental Results

We evaluated the performance of the proposed method on several video sequences of various resolutions and situations to prove its practical applicability. The results are compared to the existing background subtraction methods such as GMM[19], KDE[8], efficient GMM[14], shadow GMM[6], Zivkovic[21][7], and Gorur[10].

We implemented our algorithm in C++ for simulation with Intel Core i7 2.67GHz processor and 2.97GB RAM. Throughout the whole experiments, we do not use any kind of parallel processing methods, such as GPUs, OpenMP, pthread, and SIMD(single instruction multiple data). We have implemented the algorithm to be computed in a sequential way in a single core, to show its efficiency. The parameters of background subtraction methods are optimized one by one for various videos as was in [3], but the parameters of the proposed method are the same regardless of combining detection methods and testing videos. The used parameters are $\alpha_T = 0.1, \alpha_F = 0.01, \alpha_S = 0.05, \rho = 0.05$ and $k = \sqrt{3}$.

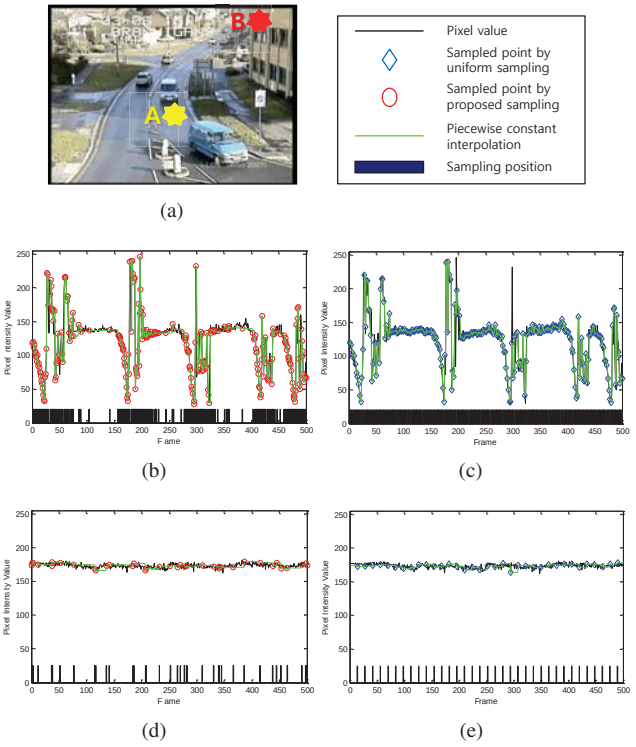
6.1. Efficiency of Active Attentional Sampling

We have monitored sequential intensity changes of two pixels (**A** and **B**) in Figure 7(a) (AVSS i-LIDS dataset is used). **A** is from a road and **B** is a pixel of a building wall. Active attentional sampling resulted in different number of samples. As we have expected, the road pixels are more frequently sampled. Also the effectiveness of active attentional sampling is compared with uniform sampling. As shown in Figure 7, the proposed sampling does not miss critical points (such as radically changing values). We have measured the RMSE (root mean squared error) of two different sampling methods in Table 2. The results show that the proposed sampling catches pixel value changing moment adaptively and accurately with much less samples.

6.2. Detection Performance Comparison

The SABS dataset[3] is used to test detection performance of the proposed method over various situations. The

[7]implementation from author: www.zoranz.net



(a) (b) (c) (d) (e)

Figure 7. The intensity of pixel **A** changes frequently because of the crossing cars. The value of **B** remains almost unchanged. The graphs show the intensity values and bars under the graphs indicate the sampled positions. For pixel **A**, the active attentional sampling samples 256 times and 25 times for pixel **B** during 500 frames. The same number of samples are generated uniformly for each sequence, and the piecewise constant interpolation is performed to reconstruct the sequence. (b) and (d) show estimated intensity graphs by proposed sampling method for A and B, respectively. (c) and (e) are reconstructed graphs by uniform sampling. We can see that the proposed one concentrate the sampling on the foreground pixels in frames with moving objects.

Sampling Method	A	B
Uniform Sampling	20.09	3.04
Proposed Sampling	9.64	3.79

Table 2. Estimation accuracy comparison in RMSE.

SABS dataset is an artificial dataset for pixel-wise evaluation of background subtraction method. For every frame of each test sequence, ground-truth annotation is provided as foreground masks. Even though it is generated artificially, there are realistic scenarios such as light reflection, shadows, traffic lights and waving trees. When considering the fact that the best F_1-*Measure* in [3] is just 0.8, SABS datasets are difficult enough to evaluate the performance of algorithm. The correctness of foreground detection is expressed by F_1-*Measure* as in [3] which is a harmonic mean of *recall* and *precision*. Detection results are

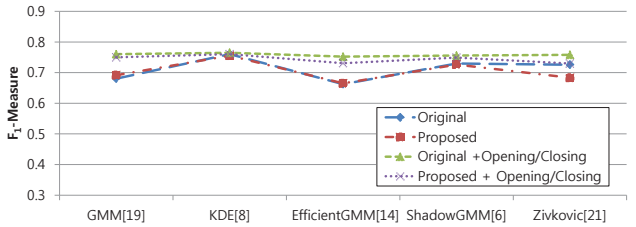


Figure 8. Best F_1-*Measure* for various background subtraction methods. Post image processing methods, such as opening/closing, also can be used.

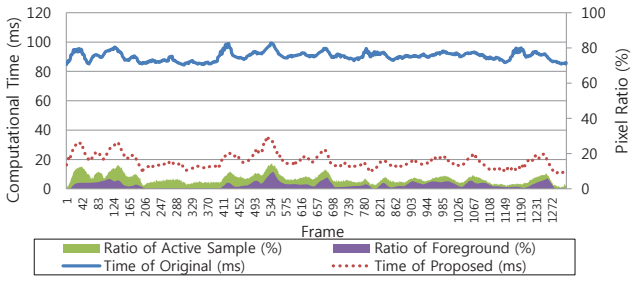


Figure 10. Computational time changes over foreground region ratio. The foreground region varies from 0 % to 10%. Not only the proposed method but also the original detection [19] takes more time as the ratio of foreground region increases.

optimally tuned and the value of Figure 8 is an average of each frames's F_1-*Measure* over whole sequences. The proposed method can be successfully combined with various background subtraction methods and post image processing methods without performance degradation.

6.3. Speed-up Performance Comparison

Figure 9 shows computation time speed-up results. The proposed method significantly shortens the detection time (on average 6.6 times). Fast detection algorithms show relatively small speed-up ratio than computationally heavy algorithms. This is because the mask generation time becomes relatively large compared to the detection time.

Figure 10 shows computation time changes over frames. GMM[19] method and SABS video[3] (bootstrap video) are used for the test. The computational time of the proposed method increases as the ratio of foreground region becomes large. However, the original GMM also takes more time when the foreground region increases. So the ratio of speed-up is maintained uniformly.

Also, we have compared the computational complexity reduction performance with similar selective sampling-based methods; Park et al.[17], Kim et al.[13] and Lee et al.[15]. All speed-up performance data are based on the optimized values of the original paper. Figure 11 show the average speed-up performances. The speed-up ratio of our method outperforms the others. The other subsampling strategies are pre-designed regardless of video situation. So

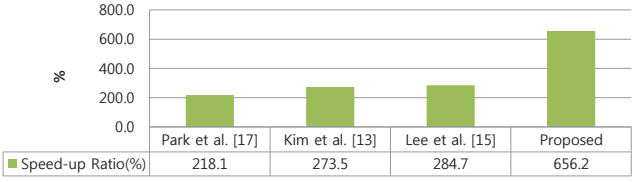


Figure 11. Comparison of selective sampling-based speed-up methods. All the methods were commonly applied to GMM [19].

Method	Original(FPS)	**Proposed(FPS)**
GPU [18]	78.9	-
GMM[19]	1.6	**18.6**
KDE[8]	3.5	**31.5**
Efficient GMM[14]	3.4	**23.5**
Shadow GMM[6]	2.2	**23.5**
Zivkovic[21]	9.7	**29.7**
Gorur[10]	11.8	**33.7**

Table 3. Comparisons of detection time in full HD videos (1920 × 1080) in terms of frame rate (FPS).

many unnecessary samplings are inevitable because of the regularly designed sampling pattern. This causes redundant calculations. The sampling strategy of our method is totally different from the grid pattern based subsampling approach. Proposed probabilistic sampling approach is more adaptive to various video situations and becomes more efficient by reducing redundant calculations.

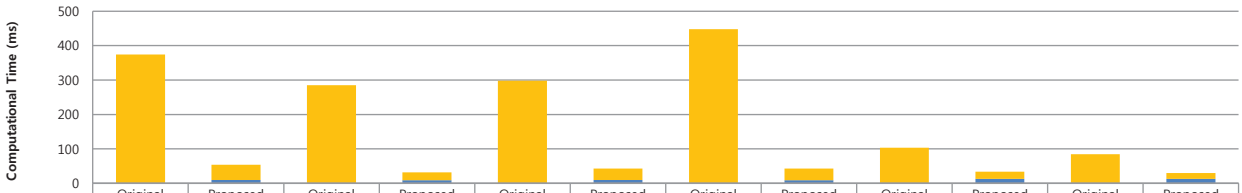
6.4. Real-time Detection in Full HD Video

Until now, allegedly, using GPU is the only solution of real time detection in full HD video[18]. However, as shown in Table 3, our method makes it possible for the conventional pixel-wise background subtraction methods to be used for high resolution videos in real-time. The experiments are performed with GeForce GTS 250 (128 CUDA cores) for GPU version [18][8] and a single core processor for the others. Every detection method is applied to a full HD video (1920 × 1080) with optimal parameters and detection time is measured with and without our method, seperately.

7. Conclusions

The computational time problem of background subtraction is very critical because it is generally considered as a lower level image processing task and the video size is getting bigger. In this paper, we proposed a speed-up method of conventional background subtraction algorithms using active attentional sampling mask generation method based on selective attention concept. The motionless background region can be skipped by attentional sampling. We designed a foreground probability map by measuring three

[8]implementation from http://www.codeproject.com/KB/GPU-Programming/cubgs.aspx



	Original	Proposed	Original	Proposed	Original	Proposed	Original	Proposed	Original	Proposed	Original	Proposed
	GMM [19]		KDE [8]		Efficient GMM [14]		Shadow GMM [6]		Zivkovic [21]		Gorur [10]	
Detection Time (ms)	374.4	44	285.2	22.6	297.9	32.7	448	33.7	103.3	20.7	84.7	17.0
Mask Gen. Time (ms)	0	9.9	0	9.1	0	9.9	0	8.9	0	13	0	12.7

Figure 9. Comparisons of the computational time speed-up. The tests were performed with full HD videos. The speed-up ratio of computationally heavy algorithms, such as GMM[19], shadow GMM[6] and KDE[8], is approximately 8.5 and the speed-up ratio of fast detection algorithms, such as Zivkovic[21] and Gorur[10], is approximately 3.

foreground region properties, and active attentional sampling is performed to make a sampling mask. Various experiments show that the proposed method can speed up about 6.6 times without detection performance deterioration. Also our method makes it possible for the conventional background subtraction algorithms to perform real-time detection in Full HD videos with a single core processor.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. 4

[2] T. Bouwmans, F. E. Baf, and B. Vachon. Statistical background modeling for foreground detection: A survey. *Handbook of Pattern Recognition and Computer Vision World Scientific Publishing*, 4:181–199, Jan. 2010. 1

[3] S. Brutzer, B. Hoferlin, and G. Heidermann. Evaluation of background subtraction techniques for video surveillance. In *In Proc. of CVPR*, pages 1937–1944, June 2011. 2, 5, 6, 7

[4] R. M. Castro. *Active Learning and Adaptive Sampling for Non-Parametric Inference*. PhD thesis, Rice University, Houston, TX, 2007. 3

[5] L. Cheng, M. Gong, D. Schuurmans, and T. Caelli. Real-time discriminative background subtraction. *IEEE Transactions on Image Processing*, 20(5), May 2011. 1

[6] J. Choi, Y. J. Yoo, and J. Y. Choi. Adaptive shadow estimator for removing shadow of moving object. *Computer Vision and Image Understanding*, 114:1017–1029, Sep 2010. 1, 6, 7, 8

[7] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996. 3

[8] A. Elgammal, R. Duraiswami, D. Harwood, L. S. Davis, R. Duraiswami, and D. Harwood. Background and foreground modeling using nonparametric kernel density for visual surveillance. In *Proceedings of the IEEE*, pages 1151–1163, 2002. 1, 6, 7, 8

[9] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. In *Machine Learning*, pages 133–168, 1997. 3

[10] P. Gorur and B. Amrutur. Speeded up gaussian mixture model algorithm for background subtraction. In *IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 386–391, Sept. 2011. 1, 6, 7, 8

[11] O. Griffiths and C. J. Mitchell. Selective attention in human associative learning and recognition memory. *Journal of experimental psychology General*, 137:626–648, 2008. 2

[12] P.-M. Jodoin, M. Mignotte, and J. Konrad. Statistical background subtraction using spatial cues. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(12):1758–1763, Dec. 2007. 2

[13] H.-K. Kim, Suryanto, D.-H. Kim, D. Zhang, and S.-J. Ko. Fast object detection method for visual surveillance. In *IEEE ITC-CSCC 2008*, 2008. 1, 7

[14] D.-S. Lee. Effective gaussian mixture learning for video background subtraction. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27, 2005. 1, 6, 7

[15] D.-Y. Lee, J.-K. Ahn, and C.-S. Kim. Fast background subtraction algorithm using two-level sampling and silhouette detection. In *Proceedings of IEEE ICIP*, pages 3177–3180, Nov. 2009. 1, 7

[16] J. M. McHugh, J. Konrad, V. Saligrama, and P. M. Jodoin. Foreground-adaptive background subtraction. *Signal Processing Letters, IEEE*, 16(5):390–393, May 2009. 1, 2

[17] J. Park, A. Tabb, and A. C. Kak. Hierarchical data structure for real-time background subtraction. In *Proceedings of IEEE ICIP*, 2006. 1, 7

[18] V. Pham, P. Vo, V. T. Hung, and L. H. Bac. GPU implementation of extended gaussian mixture model for background subtraction. In *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010 IEEE RIVF*, Nov. 2010. 1, 7

[19] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of CVPR*, pages 246–252, 1999. 1, 4, 5, 6, 7, 8

[20] G. Szwoch. Performance evaluation of the parallel codebook algorithm for background subtraction in video stream. *Multimedia Communications, Services and Security, 2011, Springer*, 149:149–157, 2011. 1

[21] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Patten Recognition Letters*, 27(7):773–780, 2006. 1, 6, 7, 8