NIC: A Robust Background Extraction Algorithm for Foreground Detection in Dynamic Scenes

Thien Huynh-The, *Student Member, IEEE*, Oresti Banos, *Member, IEEE*, Sungyoung Lee, *Member, IEEE*, Byeong Ho Kang, Eun-Soo Kim, and Thuong Le-Tien, *Member, IEEE*

***Abstract*— This paper presents a robust foreground detection method capable of adapting to different motion speeds in scenes. A key contribution of this paper is the background estimation using a proposed novel algorithm, neighbor-based intensity correction (NIC), that identifies and modifies the motion pixels from the difference of the background and the current frame. Concretely, the first frame is considered as an initial background that is updated with the pixel intensity from each new frame based on the examination of neighborhood pixels. These pixels are formed into windows generated from the background and the current frame to identify whether a pixel belongs to the background or the current frame. The intensity modification procedure is based on the comparison of the standard deviation values calculated from two pixel windows. The robustness of the current background is further measured using pixel steadiness as an additional condition for the updating process. Finally, the foreground is detected by the background subtraction scheme with an optimal threshold calculated by the Otsu method. This method is benchmarked on several well-known data sets in the object detection and tracking domain, such as CAVIAR 2004, AVSS 2007, PETS 2009, PETS 2014, and CDNET 2014. We also compare the accuracy of the proposed method with other state-of-the-art methods via standard quantitative metrics under different parameter configurations. In the experiments, NIC approach outperforms several advanced methods on depressing the detected foreground confusions due to light artifact, illumination change, and camera jitter in dynamic scenes.**

***Index Terms*— Adaptive Otsu thresholding, background subtraction, foreground detection, neighbor-based intensity correction (NIC).**

I. Introduction

During recent years, the low price of camera devices has contributed to the popularity of video-based surveillance systems. Despite the benefits in surveillance and security applications, an important challenge of such systems is accuracy [1]. Surveillance systems have to process large amounts of data from multiple cameras at the same time, which complicates their real-time application. In the analysis of human or traffic flow, the efficiency of a surveillance system significantly depends on its object detection capability. The appearance of objects in a video sequence can be detected using many techniques, which are generally classified into three main categories: 1) frame difference [2]; 2) optical flow [3]; and 3) background subtraction [4]. In the frame difference technique, objects in the current frame are detected by thresholding the difference of two adjacent frames. However, differences might not be strong enough to fully describe objects if they are moving slowly. The optical flow technique builds on the velocity field which warps one image into another. In this technique, the movement of pixels in a frame sequence is detected by analyzing the change of pixel intensity. Difficulties in calculating the true velocity and representation of the moving information are the main limitations of this approach. Finally, the background subtraction technique, the most commonly used method for foreground detection, determines objects by extracting the difference between the current frame and the background. This technique consists of two stages: 1) background estimation from the frame sequence and 2) foreground extraction by subtracting the estimated background.

Despite the widespread use of background subtraction techniques, this class of approaches presents some crucial drawbacks, including its performance in terms of computational costs and the accuracy of the background estimation [5]. The background is fundamentally defined as the reference frame, in which pixel intensities appear with the highest probability. Several existing estimation algorithms for foreground detection [6], such as the Gaussian mixture model (GMM) and kernel density estimation (KDE), are susceptible to harsh conditions such as luminance changes and repetitive movements. Although these models can enhance their accuracy by using more frames for the background estimation, they are ineffective in most practical situations because of their high computational costs. Some developments have been considered to overcome the problems; however, those options also increase the complexity of the calculations [7]. Another problem in background subtraction is thresholding the difference image for foreground segmentation. The threshold value is identified to separate the grayscale difference image into two parts: 1) the scene background with pixel intensity smaller than the

Manuscript received October 28, 2015; revised December 10, 2015; accepted January 26, 2016. Date of publication March 16, 2016; date of current version June 30, 2017. This work was supported by the National Research Foundation of Korea within the Ministry of Science, ICT and Future Planning through the Korean Government under Grant NRF-2014R1A2A2A01003914. This paper was recommended by Associate Editor J. Lu. *(Corresponding author: Sungyoung Lee.)*

T. Huynh-The, O. Banos, and S. Lee are with the Department of Computer Engineering, Kyung Hee University, Yongin 17104, South Korea (e-mail: thienht@oslab.khu.ac.kr; oresti@oslab.khu.ac.kr; sylee@oslab.khu.ac.kr).

B. H. Kang is with the School of Computing and Information System, University of Tasmania, Hobart, TAS 7005, Australia (e-mail: byeong.kang@utas.edu.au).

E.-S. Kim is with the Department of Electronic Engineering, Kwangwoon University, Seoul 01897, South Korea (e-mail: eskim@kw.ac.kr).

T. Le-Tien is with the Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology, Ho Chi Minh City 700000, Vietnam (e-mail: thuongle@hcmut.edu.vn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2016.2543118

TABLE I
TAXONOMY OF EXISTING BACKGROUND SUBTRACTION METHODS

Method class	Research	Limitation
Statistical Model	[8]–[14]	Foreground objects are sometimes modeled into the estimated background in the case of non-moving objects remaining for a long time.
Gaussian Mixture Model	[15]–[23]	The estimated background is quite sensitively in high speed motion environments. The parameters cannot be flexibly estimated for real-world noise environments.
Codebook Model	[24]–[28]	The model requires a long time period of time for codebook construction and a high memory for codeword storage.
Kernel Density Estimation	[5], [29]–[33]	Modeling simultaneously moving objects with various speeds is limited by the first-in-first-out updating scheme.
Advanced Model: ViBE, $\sum -\Delta$	[34]–[37]	The estimated background from ViBE is restricted in the hash conditions in scenes due to its random replacement policy. The $\sum -\Delta$ background is quickly degraded in quality under slow or dense movement conditions.
Learning Network Model	[38]–[41]	The models requires highly computational cost and more memory for the learning stage.

threshold and 2) the foreground with pixel intensity greater than the threshold. This value directly affects the detection results: some objects may be neglected if this value is set too high, but unexpected noise such as shadows or light artifacts can be recognized as candidate objects if the value is set too low.

In this paper, we propose an efficient background subtraction method based on a novel background estimation algorithm, neighbor-based intensity correction (NIC), to enhance foreground detection accuracy in dynamic scenes. The idea comes from the consideration and comparison of two intensity patterns generated from the background and the current frame to modify the pixel intensity in the current background. First, the algorithm detects motion pixels in the scene and filters them with a steadiness factor. This factor, which measures the steadiness of every pixel in the current background, is consecutively updated for each frame. As the main contribution, the rule for intensity modification is then performed by comparing the standard deviation results calculated from neighbor pixels to decide whether a current pixel belongs to the background or the current frame. By adjusting the window size, the algorithm is flexible enough to detect multiple objects moving at different speeds in a scene. The estimated background is next provided to the background subtraction scheme to extract the foreground with an optimal threshold identified by the Otsu method. We benchmarked our proposed scheme on several well-known data sets for object detection and tracking area using standard statistical factors. Both qualitative and quantitative assessments show the superiority of our proposed method in dynamic environments. We also compare the foreground detection accuracy of the proposed method with that of recent state-of-the-art methods.

The remainder of this paper is organized as follows. Section II reviews existing background estimation algorithms. Section III describes the background subtraction scheme with the NIC algorithm for modeling. We present our experimental results in background estimation and foreground detection with discussion and comparison with other state-of-the-art algorithms in Section IV. The conclusion and suggestion for future work are given in Section V.

II. RELATED WORK

In background subtraction techniques, an observed image is compared with an estimated background image that does not contain objects. This comparison process, called foreground detection, separates an image into two sets: one contains the object area, denoted by 1-b pixels, and another contains the background of a scene, denoted by 0-b pixels. In the background subtraction technique, the estimated background has an important influence on the detection accuracy rate as evidenced by many studies devoted to this issue. Taxonomy of existing background subtraction methods is summarized in Table I.

A simple scheme to model the background of a scene is the statistical approach [8]–[12]. By observing several frames from the input video sequence, these schemes determine each pixel intensity using the highest probability. Thus, Li *et al.* [8] incorporated the most significant and frequent features of each pixel in the spatiotemporal dimension with a Bayesian network to describe the background appearance. The scheme proposed in [9] extended the work of [8] by replacing pixel-based statistics with region-based statistics by merging and splitting the dynamic background region. Akula *et al.* [11] generated a reference background from an initial set of N frames by taking the pixelwise median of each frame. The statistical background model for each pixel of the input set of N frames is constructed by calculating the weighted variance of the image. The algorithms described in [13] and [14] exploited color variance to dynamically retrieve the initial background through a series of frames. The limitation of statistical methods is that they may falsely combine foreground objects into the background when objects remain still for a long time.

GMM, the most commonly used technique for background estimation, was first introduced in [15] and [16]. In this technique, each pixel value is estimated using a Gaussian mixture and is continuously updated by an online approximation. The downside of the GMM derives from its assumptions that the background area is larger and more frequently visible than the foreground area and that its variance is small enough. Several improved versions [17]–[23] have been proposed for object detection using the background

subtraction technique. For example, Zivkovic [17] considered an improved adaptive GMM (I-GMM), in which the parameters and components of the mixture model are constantly computed for each adaptive pixel. Shimada *et al.* [19] focused on an automatic mechanism to change the number of Gaussian kernels in each pixel to reduce computational cost. Another development of GMM [18] is to improve the convergence rate of the estimation process. Elqursh and Elgammal [20] modeled tracking content in a low-dimensional space by using GMM for motion estimation in a moving camera environment. To eliminate illumination changes and noise in intelligent video-based surveillance systems, a novel GMM-based solution was proposed in [21]. The approach has three key contents: 1) an explicit spectral reflection model analysis; 2) an online expectation-maximization algorithm; and 3) a two-stage foreground detection algorithm. Followed by probabilistic regularization, a Dirichlet process-based GMM [23] was proposed to estimate per-pixel background distributions. For continuous update of scene changes in the model, researchers have also developed some learning models to improve performance in specific cases. To automatically determine the parameters of the GMM algorithm, a particle swarm optimization method and an online self-adaptive mechanism using an iterative log-moment estimation [42] have also been considered. Although GMM-based improvements have been proposed for difficult scenes, they still have general limitations. First, the background sensitivity cannot be accurately adjusted, so the model can miss the detection of some low-speed objects. Second, parameter estimation is not an easy task in noisy real-world environments, especially those with variant features in the hardware implementation.

To avoid parameter tuning for the probabilistic model, some researchers focus on nonparametric approaches for background modeling. A real-time algorithm first proposed in [24] quantizes the background pixel values into codebooks that describe a compressed background model for a number of frames. Many approaches [25]–[28], [43] were proposed to improve accuracy in poor conditions. To tolerate environmental changes such as dynamic backgrounds and sudden changes of luminance, Lin *et al.* [27] described a single-layer codebook model. Including parameters of frequency for accessing, deleting, matching, and adding code words into the basic model, Ilyas *et al.*'s method [25] shows significant improvements in most settings. In another research, Guo *et al.* [26] proposed a multilayer codebook model to remove most of the nonstationary background and improve the processing efficiency. Although they can obtain high performance in real-time environments, the drawback of codebook approaches is the long time required to construct the model and the high amount of memory needed to store the code words.

Another nonparametric approach often used in background subtraction is KDE [44]. This technique estimates the probability density function with a histogram to redefine the values of current background pixels. Some authors have used KDE to decide whether a pixel belongs to the background or foreground, Mittal and Paragios [5] estimated density functions for foreground detection by integrating the intensity information and optical flow content, and Tang *et al.* [29] proposed weighted-KDE with a probability map construction algorithm to automatically extract the foreground. Li *et al.* [30] proposed a real-time moving object detection scheme developed on KDE to eliminate noise, remove shadow, and enhance the foreground contour. The foreground-adaptive background subtraction method [31] considers the KDE-based background model because of its simplicity and high performance. Liu *et al.* [33] presented a hybrid model integrating KDE and GMM to construct a probability density function for the background and moving object model. Although KDE-based methods can provide fast responses to high-speed movement, their ability to handle concomitant movement events at various speeds is restricted by their first-in first-out updating manner.

Barnich and Van Droogenbroeck [34], [35] proposed the visual background extraction (ViBE) background modeling technique. It determines whether a pixel belongs to the background by randomly comparing its intensity within neighborhoods. It then propagates the pixel intensity into the background model as the neighboring pixel for the next pixel examination. ViBE outperforms some state-of-the-art methods in computational speed and estimated background quality in experiments. The ViBE downscale version makes the computational cost small enough for it to be embedded into digital cameras at the expense of accuracy. Although ViBE can provide satisfactory detection results compared with existing approaches, it has issues under harsh conditions, such as scenes with a dark background, shadows, and frequent background changes.

In the sparse modeling domain, background subtraction techniques use variants of principle component analysis and matrix decomposition methods, in which the background is estimated with an adaptive subspace. In another research, Guyon *et al.* [45] addressed the spatial connections of the foreground pixels by using a low-rank and block-sparse matrix decomposition. For salient motion detection in a realistic environment, Gao *et al.* [46] contributed dynamic estimation to impose spatial coherence between regions in the foreground. Compared with GMM-based approaches, a recursive algorithm [47] based on the sigma-delta filter is an interesting attempt because of its high-speed processing and very low memory requirement. An improved version of the sigma-delta algorithm was suggested in [36] to cope with slow motion and objects that stop suddenly. It applies an intensity pixel updating mechanism based on the confidence level and image covariance at each pixel. In another study, Vargas *et al.* [37] developed a basic algorithm to reduce the computational cost of background modeling. To balance the computational cost and background robustness, it used a confidence measurement for each pixel in its selective background model updating mechanism.

To minimize memory requirements, Gruenwedel *et al.* [48] presented a dual-layer background model, one for low-adaptation speed for the long-term background and another for high adaptation speed for the short-term background. Although it was developed as a low-complexity algorithm to be used in real-time applications, parameter selection is a challenging issue to retain sufficient quality

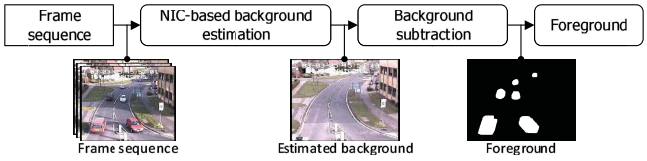


Fig. 1. Overview of the foreground detection method using the NIC based on the background subtraction scheme.

for the estimated background. Culibrk *et al.* [38] and Maddalena and Petrosino [39] used neural networks to model the background and temporally update it to reflect its observed pixelwise statistics. References [40] and [41] described the radial basis function as an unsupervised learning process using artificial neural networks for multibackground generation.

III. Neighbor-Based Intensity Correction for Foreground Detection

A. Method Overview

In general, the proposed method based on the background subtraction scheme includes two main phases: 1) background estimation and 2) foreground extraction. Fig. 1 shows the overview of the foreground detection method. For the background estimation stage, we propose the novel NIC algorithm to model a robust background in a dynamic scene. For foreground segmentation, we then use the background subtraction scheme to extract the foreground with an optimal threshold calculated by the Otsu algorithm.

B. Neighbor-Based Intensity Correction Algorithm

The NIC algorithm consecutively models and updates the background from the motion information of the current frame during foreground extraction. The workflow of the proposed algorithm is concretely represented in Fig. 2 with the frame sequence as input and the estimated background as output. As the prior knowledge, the first frame from the input video sequence is assumed to be the initial background

$$B_1(x, y) = F_1(x, y) \tag{1}$$

where $B_1(x, y)$ and $F_1(x, y)$ are the intensity values of a pixel at the coordinate (x, y) with $x \leq P$ and $y \leq Q$ in the initial background and the first frame, where P and Q are the width and height of the input frame.

For the ith frame ($\forall i \geq 2$), the background image B_i for the background subtraction will be estimated from B_{i-1}. In the first step, the difference between the current background and the current frame, denoted by D_i, is calculated through the following equation for the grayscale image:

$$D_i(x, y) = |F_i(x, y) - B_{i-1}(x, y)|; \quad \forall i \geq 2. \tag{2}$$

The difference image D_i contains information about moving objects and noise; therefore, D_i needs to be segmented into the background and moving object areas by the threshold τ

$$\mathcal{D}_i(x, y) = \begin{cases} 1; & \forall D_i(x, y) \geq \tau \\ 0; & \forall D_i(x, y) < \tau. \end{cases} \tag{3}$$

The binary image $\mathcal{D}_i$ has 0-b pixels representing the nonmotion areas and 1-b pixels representing the motion areas. In principle, moving objects have greater difference than noise and shadow artifacts. A high value for τ eliminates the noise, but some motion pixels can be unexpectedly misidentified. In contrast, if τ is low, noise pixels are sometimes misclassified as motion pixels. Therefore, τ, which clearly has an influence on $\mathcal{D}_i$, needs to be carefully selected through experimental evaluations.

Principally, the proposed algorithm will be applied for motion pixels in a set of $\mathcal{D}_i = 1$; however, to reduce the appearance of outliers as much as possible, we consider the steadiness of each pixel for filtering. The steadiness factor, denoted by $S(x, y)$, is calculated based on the number of times the intensity changes and is updated with each new frame. If the intensity changes in two consecutive frames, the pixel is less steady than that remains the same. The steadiness of each pixel is computed and accumulated in a sequence of frames by the following equation:

$$S_i(x, y) = \begin{cases} S_{i-1}(x, y) - 1; & \forall \mathcal{D}_i(x, y) = 1 \\ S_{i-1}(x, y) + 1; & \forall \mathcal{D}_i(x, y) = 0 \end{cases} \tag{4}$$

where S_i is the steady matrix at the ith frame. It is initialized with a value of zero, i.e., $S_1(x, y) = 0$. In accumulating frame by frame, the steady value of a nonmotion pixel becomes greater than the value of a motion pixel. For example, an arbitrary pixel p is detected as the motion pixel for t_1 frames and as the nonmotion pixel for t_2 frames after $T = t_1 + t_2$ frames. Calculated by (4), the steady value $S_T(x_p, y_p) = (-1)t_1 + (1)t_2$ is negative if $t_1 > t_2$, and positive if $t_1 < t_2$.

Extended from the pixel steadiness, the NIC algorithm can evaluate the robustness of the current background. During the estimation process, intensity modification might not be necessary to reduce the computation if the current background approaches the background truth. Concretely, the decision-making is controlled as follows: if the minimal value of the steadiness factor is greater than a steady threshold, denoted by $\delta > 0$, i.e., $\min_{(x,y)}(S_i(x, y)) \geq \delta$, the algorithm ignores the modification process. The current background is therefore maintained for consideration of the next frame, i.e., the $B_i = B_{i-1}$ for the $(i + 1)$th frame and then directly entered to the foreground extraction stage. In the opposite case, $\min_{(x,y)}(S_i(x, y)) < \delta$, the NIC algorithm modifies the current background. The decision-making is briefed as follows:

$$\begin{aligned} &\text{Apply\_NIC;} \quad \text{if } \min_{(x,y)}(S_i(x, y)) < \delta \\ &\text{No\_update;} \quad \text{if } \min_{(x,y)}(S_i(x, y)) \geq \delta \end{aligned} \tag{5}$$

where δ is set based on the density of moving objects in a scene, i.e., a greater value for crowded scenes and vice versa. Clearly, more frames are required for the estimation process to reach to a robust background.

For the NIC algorithm, we consider the combination of two conditions for selecting pixels to modify their intensity.

1) They belong to the current motion set $\mathcal{D}_i(x, y) = 1$.
2) They have a negative steadiness value $S_i(x, y) < 0$.

We give this reidentification to reduce the computational cost in processing the NIC algorithm for inappropriate points generated from sudden light changes or artifacts. The pixels

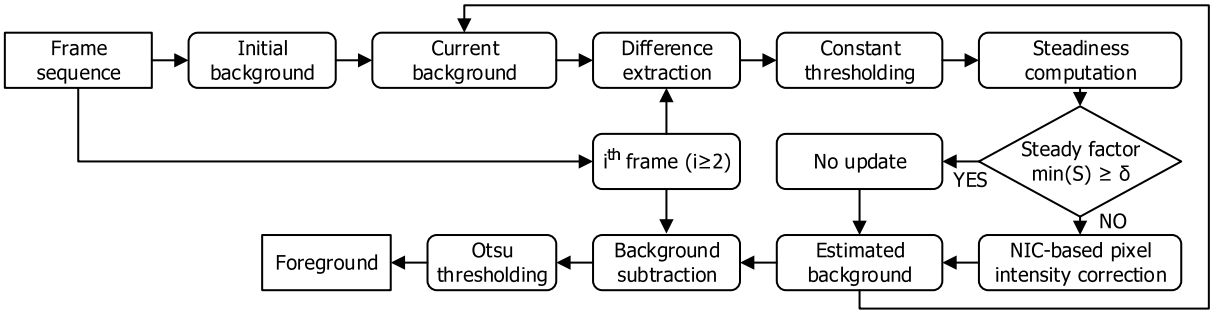


Fig. 2. Workflow of the foreground detection using the NIC algorithm for background estimation.

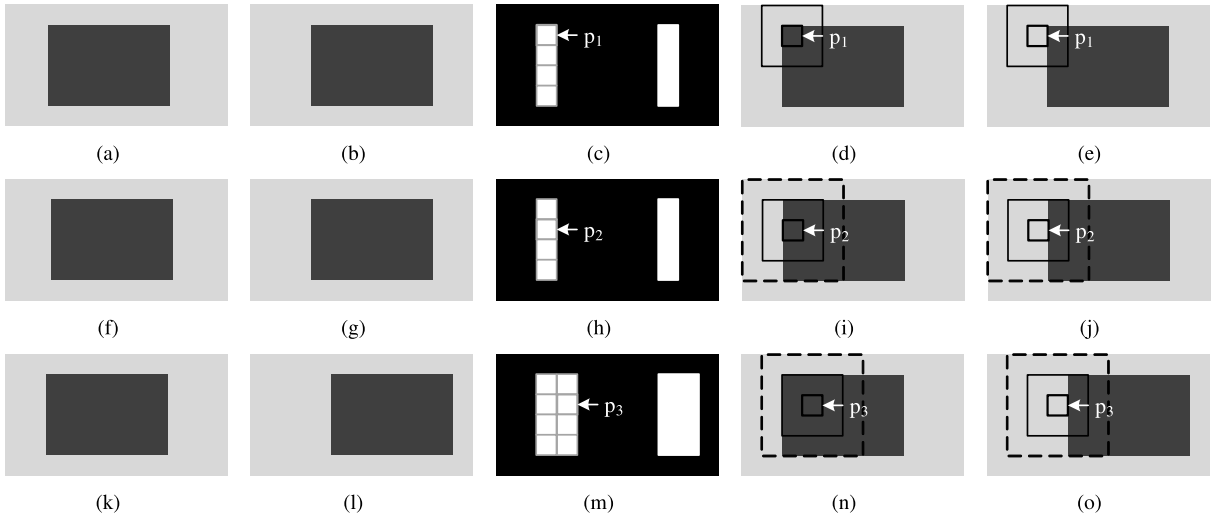


Fig. 3. Illustration of NIC operation in different cases of moving objects. First and second rows: the single-pixel shifting. Third row: the multipixel shifting. Left to right: the background image, the current frame, the difference image, and the background and current frames with (3×3) and (5×5) windows captured surrounding a motion pixel.

are filtered by two conditions

$$\mathcal{P}_i = \{(x, y) | [\mathcal{D}_i(x, y) = 1] \cap [S_i(x, y) < 0]\} \tag{6}$$

where $\mathcal{P}_i$ is a set of filtered pixels. The intensity correction algorithm is executed for pixels in the set $\mathcal{P}_i$. The main idea of the algorithm is illustrated in Fig. 3. A first case for the single-pixel shift is represented between the first row of the background image in Fig. 3(a) and the current frame in Fig. 3(b). The pixels belonging to the set $\mathcal{P}$ are shown in Fig. 3(c). In this step, eight pixels in the set $\mathcal{P}$ need to be modified to the correct value. Concretely, four pixels on the left side should be adjusted to the values of the corresponding pixels in the current frame and four pixels on the right side should be adjusted to the background pixel value. Accordingly, we construct two windows from the background image and the current frame surrounding each pixel, denoted by $W^B_{(x,y)}$ and $W^F_{(x,y)}$, respectively. For instance with pixel p_1 in Fig. 3(c), two windows are identified in Fig. 3(d) and (e). The intensity patterns of the two windows are different, i.e., the ratio between the number of motion and nonmotion pixels are dissimilar. This difference is exploited for correction; consequently, we calculate the standard deviation values of the samples in the two sets (known as pixels in the windows). A small value indicates that the pixel values tend to be closer to the average, and a greater value shows that the pixel values are dispersedly spread. Fundamentally, the standard deviation σ of a square window is generally calculated by the following equation:

$$\sigma = \sqrt{\frac{1}{N} \sum_{p=1}^{n} \sum_{q=1}^{n} (I(p, q) - \mu)^2} \tag{7}$$

where n is the size of a square window and $N = n^2$ is the number of pixels. The mean μ of the window is determined from the pixel intensity in image I

$$\mu = \frac{1}{N} \sum_{p=1}^{n} \sum_{q=1}^{n} I(p, q). \tag{8}$$

For each pixel in $\mathcal{P}_i$, we calculate two standard deviation values, denoted by $\sigma^B_{(x,y)}$ and $\sigma^F_{(x,y)}$, from the two windows of the current background and the current frame. The rule for the correction algorithm is based on the result of the comparison process between the two standard deviation values as follows:

$$B_i(x, y) = \begin{cases} B_{i-1}(x, y); & \forall (x, y) \notin \mathcal{P}_i \\ B_{i-1}(x, y); & \forall (x, y) \in \mathcal{P}_i \big| \sigma^F_{(x,y)} \geq \sigma^B_{(x,y)} \\ F_i(x, y); & \forall (x, y) \in \mathcal{P}_i \big| \sigma^F_{(x,y)} < \sigma^B_{(x,y)}. \end{cases} \tag{9}$$

In the window construction, the size of $(n \times n)$ can be adjusted based on the movement speed. To investigate the influence of window size on the result of the intensity correction rule, we assume some particular values for particular cases. Let n_0 and n_1 be the number of nonmotion and motion pixels with the intensity g_0 and g_1, respectively. So the window contains $(n_0 + n_1)$ pixels. The mean of the pixel values is calculated by (8) is

$$\mu = \frac{g_0 n_0 + g_1 n_1}{n_0 + n_1}. \tag{10}$$

The standard deviation can be derived under the variant developed from (7)

$$\begin{aligned} \sigma &= \sqrt{\frac{1}{n_0 + n_1}[n_0(g_0 - \mu)^2 + n_1(g_1 - \mu)^2]} \\ &= \frac{\sqrt{n_0 n_1}}{n_0 + n_1}|g_0 - g_1|. \end{aligned} \tag{11}$$

Using (11) for the standard deviation calculation, the terms $|g_0 - g_1|$ and $(n_0 + n_1)$ are constant components, so the result is decided by the term $(n_0 n_1)^{1/2}$, i.e., the standard deviation value depends on the number of motion and nonmotion pixels. Consider the first case represented in the first row of Fig. 3 with a window size of (3×3). The window $W^B_{p_1}$ captured from the current background contains five nonmotion pixels and four motion pixels in Fig. 3(d), and the window $W^F_{p_1}$ covers seven nonmotion pixels and two motion pixels captured from the current frame in Fig. 3(e). Because $\sigma^F_{(x,y)} < \sigma^B_{(x,y)}$, the p_1's intensity is modified from g_1 to g_0 by referring to (9).

In the next example shown in the second row of Fig. 3, a problem occurs when the number of motion and that of nonmotion pixels in two (3×3) windows, denoted by the solid line boundary, established at the p_2-pixel are antithetic. That leads to an erroneous background correction, i.e., p_2 will be preserved by its intensity instead of being modified from g_1 to g_0. This problem can be overcome by changing the window size. In this example, we modify the size from (3×3) to (5×5), denoted by the hashed line boundary. The third case illustrates the multipixel shifting motion that usually occurs in the practical environment, shown in Fig. 3(m). The intensity modification for p_3 is incorrect with a (3×3) window. Similar to the second case, this drawback is solved if the window size is changed to (5×5). The terms of $(n_0 n_1)^{1/2}$ are recalculated to bring the correct modification. Since σ is influenced by the number of motion and nonmotion pixels in a window (11), the size of windows is an important impact in NIC algorithm. Therefore, modifying this parameter is needed to be adaptive with multimodal background scenes, such as low-, medium-, and high-speed object movement. Some experimental evaluations hereafter prove the influence of the window size on performance in terms of background estimation accuracy.

After the intensity correction process, the NIC algorithm submits the estimated background to the foreground extraction process as an input and stores it to continuously process in $(i + 1)$th frame.

C. Foreground Extraction Using Background Subtraction Scheme

In this stage, the foreground is detected based on the background subtraction scheme with an adaptive threshold. The difference is extracted from the current frame F_i and the estimated background image B_i by reusing (2)

$$D^*_i(x, y) = |F_i(x, y) - B_i(x, y)|; \quad \forall i \geq 2. \tag{12}$$

We then segment the foreground based on the difference image D^*_i using an optimum value identified from the Otsu method [49], which is fundamentally formulated to perform clustering-based image thresholding for segmentation when two pixel classes (foreground pixels and background pixels) are assumed to be sufficiently distinguishable. The Otsu method thus calculates the optimal threshold to separate an image into a background area, denoted by G_0, and a foreground area, denoted by G_1. Through the minimization of intra-class variance, the threshold can reduce the classification error. The threshold, which is exhaustively sought, minimizes the weighted sum of the variance of the two classes

$$\sigma^2_\omega(g) = \omega_{G_0}(g)\sigma^2_{G_0}(g) + \omega_{G_1}(g)\sigma^2_{G_1}(g) \tag{13}$$

where $\omega_{G_0}(g)$ and $\omega_{G_0}(g)$ are the class probabilities at the intensity g. The two corresponding pixel classes $\sigma^2_{G_0}$ and $\sigma^2_{G_1}$ are the individual class variances. The formulas for element calculation are defined in [49]. The threshold is defined as

$$\tau_{\text{opt}} = \arg\min_g \left(\sigma^2_\omega(g)\right). \tag{14}$$

The thresholding process is similar to (3) except that it replaces τ with τ_{opt}

$$\mathcal{D}^*_i(x, y) = \begin{cases} 1; & \forall D^*_i(x, y) \geq \tau_{\text{opt}} \\ 0; & \forall D^*_i(x, y) < \tau_{\text{opt}}. \end{cases} \tag{15}$$

It is important to note that D^*_i is the grayscale intensity image in this research after converting the outputs, i.e., F_i and B_i, from RGB color space. The foreground sometimes consists of disconnected edges caused by drastic luminance changes in dynamic scenes. To fuse narrow breaks and long thin gulfs, eliminate small holes, and fill gaps in the contour, some morphological operations [49] should be used during postprocessing.

IV. Experimental Results and Discussion

A. Experimental Setup

In the background subtraction technique, the estimated background plays an important role in the accuracy of the foreground detection. Therefore, we consider the performance of the NIC algorithm in assessing the estimated background quality. We performed all of our experiments on a desktop PC operating Windows 7 with a 2.67-GHz Intel Core i5 CPU and 4-GB RAM. We used MATLAB R2013a for the simulation. The foreground detection method is validated on several video sequences from well-known benchmark data sets: 1) CAVIAR [50]; 2) PETS 2009 [51]; 3) PETS 2014 [52]; 4) AVSS 2007 [53]; and 5) CDNET 2014 [54]. These video

sequences include indoor and outdoor scenes that represent typical situations in video surveillance, and they are widely used for evaluating object detection and tracking algorithms.

1) CAVIAR 2004 Data Set: We considered two sequences, Walk2 with 1055 frames and Walk3 with 1379 frames captured at resolution of 384 × 288, to describe the walking scenario from the CAVIAR-INRIA data set. In the sequences, some persons come into a laboratory, walk around, and leave under poor illuminance conditions with light artifacts and camouflage.

2) PETS 2009 Data Set: The PETS 2009 data Sets are multisensor sequences containing different crowd activities captured by different cameras at various vision angles. For experimental testing, we selected four sequences from the data set S2_L1 (*View_001*, *View_003*, *View_005*, and *View_006*), which are mostly used in tracking applications. These videos have medium density crowds in overcast, bright sunshine, and shadowed conditions. The set contains more than 3180 color images under a resolution of 768 × 576.

3) PETS 2014 Data Set: The PETS 2014 data sets, supported by the EU project ARENA, are multisensor sequences of different activities around a parked vehicle (PV) in a parking lot. For object detection, we chose three sequences, *TRK_RGB_2*, *TRK_RGB_3*, and *ENV_RGB_3* from categories 01_01 and 01_02. They present a low degree of difficulty, showing the various walking activities of a group of three people. More than 3700 video frames sampled at 30 frames/s are contained in the two categories with a resolution of 1280 × 960 pixels.

4) AVSS 2007: The PV sequence in the AVSS 2007 shows common challenges in video processing with dynamic motion speed of objects, fast illumination change, and camera jitter in a strong windy environment. The video contains 3748 frames sampled at 25 frames/s with 720 × 576 pixels of resolution.

5) CDNET 2014: In the CDNET 2014 data set, two video sequences are selected for testing, namely, *Canoe* from dynamic background category and *Badminton* from camera jitter class. The first sequence has 1189 frames at a resolution of 320 × 240, while the second has 1150 frames with resolution at 720 × 576.

B. Evaluation Metrics

For the evaluation of the background estimation performance, we use the peak signal-to-noise ratio (PSNR), a standard metric, to measure the quality of the estimated background, i.e., the estimation accuracy when compared with the background truth. PSNR is calculated as follows:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{mse}}\right) \tag{16}$$

where mse is the mean square error. Moreover, two additional quality measurement metrics, structural similarity (SSIM) and color image quality measure (CQM) [55]–[57], are considered for a more comprehensive quantitative analysis. The SSIM value is obtained as

$$\mathrm{SSIM} = \frac{1}{K}\sum_{t=1}^{K}\mathrm{SSIM\_MAP}(B_i, B_T) \tag{17}$$

where K is the number of local windows in the image. B_i and B_T describe the estimated background and the background truth, respectively. The SSIM_MAP function [55] is here defined as the product of the luminance, contrast, and SSIM between two windows extracted from the original background and the estimated background. The CQM value is calculated as follows:

$$\mathrm{CQM} = (\mathrm{PSN}R_Y \times R_W) + \left(\frac{\mathrm{PSN}R_U + \mathrm{PSN}R_V}{2}\right) \times C_W \tag{18}$$

where $R_W = 0.9449$ and $C_W = 0.0551$ are the weights on the human perception of these cone and rod sensors; the PSNR quality of each YUV channel is calculated separately.

Besides background estimation evaluation, we also perform a quantitative assessment of the foreground detection and compared our results with those of state-of-the-art methods using the receiver operating characteristic (ROC) diagram to describe the relationship between true positive rate (TPR) and false positive rate (FPR). They are calculated as follows:

$$\mathrm{TPR} = \frac{tp}{tp + fn} \tag{19}$$

$$\mathrm{FPR} = \frac{fp}{fp + tn} \tag{20}$$

where tp is the total number of true positive pixels and $(tp + fn)$ represents the total number of objects in the ground truth. fp is the total number of false positive pixels, and $(fp + tn)$ represents the total number of negative pixels in the ground truth.

C. Background Estimation Assessment

We investigate the three key parameters of the NIC: 1) the threshold τ; 2) the steady threshold δ; and 3) the window size based on the quality of the estimated backgrounds. The quantitative results of the estimated backgrounds for the *View_001* sequence of PETS 2009 using PSNR, SSIM, and CQM metrics are plotted in Fig. 4. These results are obtained by comparing the estimated backgrounds with the true background image. For verifying the influence of the threshold τ, we fix the window size to (5 × 5) and δ to 5, and sweep the value of τ from 5 to 50 as in Fig. 4(a), (d), and (g) corresponding to the quantitative metrics. The background quality is improved in increasing τ (from 5 to 20), and the highest quality background is achieved at $\tau = 20$. However, the results deteriorate when the threshold is larger than 20. The average values (aPSNR, aSSIM, and aCQM) for the whole sequence are further provided in Fig. 4. NIC algorithm is then benchmarked with the window size WS $= \{3, 5, 7, 9, 11, 13\}$ to validate its effect on the estimation result when setting $\tau = 20$ and $\delta = 5$ as constant parameters. From the results in Fig. 4(b), (e), and (h), the larger the size of the window, the higher the background quality attained by NIC. Although, the larger window size can adapt to medium- and high-speed motion to deliver a greater quality, it unfortunately requires a higher computational cost to calculate σ (7). Compared with τ, the window size parameter has a lower impact on background quality, 0.86 versus 2.56 dB in PSNR metric. In the last parameter investigation, we change

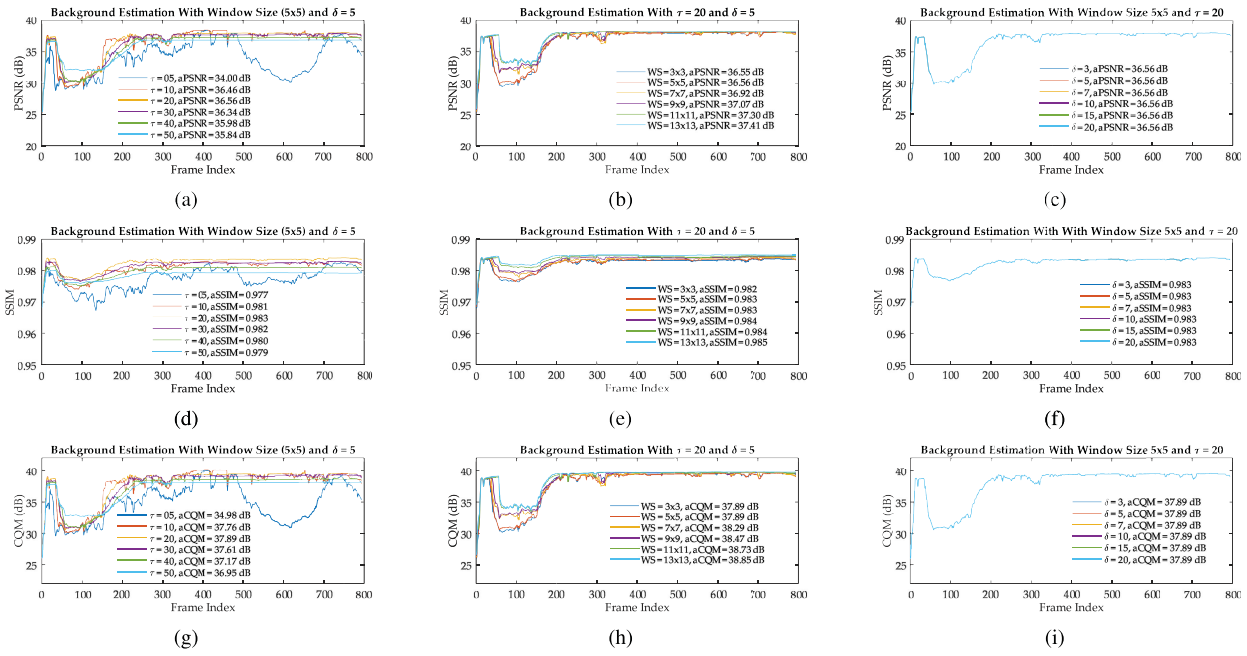


Fig. 4. Evaluating quality of estimated backgrounds on the sequence *View_001* of PETS 2009 under different values of threshold τ, window size, and steady threshold using the standard metrics. (a)–(c) PSNR. (d)–(f) SSIM. (g)–(i) CQM.

the values of steady threshold $\delta = \{3, 5, 7, 10, 15, 20\}$. The algorithm presents the same result for each quantitative metric for different values of δ as shown in Fig. 4(c), (f), and (i). This special behavior demonstrates the least importance of δ in the proposed algorithm. It should be noted that obtaining a robust background in the dynamic motion scene is difficult because some background objects, such as flag, leaf, and ribbon, can incessantly move by natural wind in the outdoor environment. Therefore, δ can be omitted in the NIC parameter configuration for outdoor cases. According to the obtained results, we recommend to use $10 \leq \tau \leq 30$ and $5 \leq \text{WS} \leq 9$ for a good tradeoff between estimation accuracy and computational cost.

D. Foreground Extraction Assessment

This section evaluates the foreground detection on 12 video sequences by pixelwise accuracy assessment.

1) Qualitative Results: The visual results of foreground detection are given in Fig. 5 including the original frame (the first column), the ground truth (the second column), and detected foreground (the last column), are obtained under the best performing parameter configuration, i.e., $\tau = 20$, $\text{WS} = 9$, and $\delta = 5$. For the two CAVIAR sequences, we randomly selected frame #142 in Walk2 and frame #548 in Walk3 to validate and analyze the detection performance. The left wall in the scene has glass windows with the light radiated directly from outdoor. The illumination changes of reflection and shade on the floor present challenges to foreground extraction. Frames #398, #723, #399, and #715 selected from PETS 2009 sequences are then plotted in the next four rows in Fig. 5. The scene, a small campus area where some people walk at a medium speed, is recorded by different cameras placed at various directions and distances. The wind-blown motions of tree branches in *View_003* and the ribbon in *View_001* explain the decrease of accuracy. Furthermore, some object shapes in *View_006* are wrongly detected because of similar intensity between the background and the object areas. Also, moving objects are sometimes occluded by a fixed background object, which leads to shape splitting in the detected objects, for example, for the sign pole in *View_003*. Another objective challenge in this data set is the overall chrominance in videos captured during different parts of the day. Three sequences from PETS 2014 capture the walking activity in a vehicle parking area for tracking evaluation. The foregrounds extracted from frames #630, #606, and #206 corresponding to three sequences are shown in the next three rows in Fig. 5. One problem in these videos is the perspective projection, in which the objects become smaller as their distance from the observer increases. Object segmentation therefore becomes more difficult, especially with objects far from the observer. Furthermore, the foreground quality is degraded by the constant unpredictable motions of plants in the scene. Foregrounds extracted from frames #3349, #940, and #922 of *PV_Medium*, *Canoe*, and *Badminton* sequences are, respectively, shown in the last rows. Camera jitter, dynamic background, and homogeneous chroma in these videos evidently become challenges for extracting a desirable foreground.

2) Quantitative Results: The quantitative results are considered through the ROC curves plotted in Fig. 7. Each sample in Fig. 5 is evaluated under various parameter configurations and each configuration corresponds to a depicted curve point. We fix $\delta = 5$ in this experiment due to its insignificance. The threshold τ varies from 10 to 30, and the window size varies from 5 to 9 as suggested in the previous section. Batch TPR and FPR results from frames #800 to #1150 of the

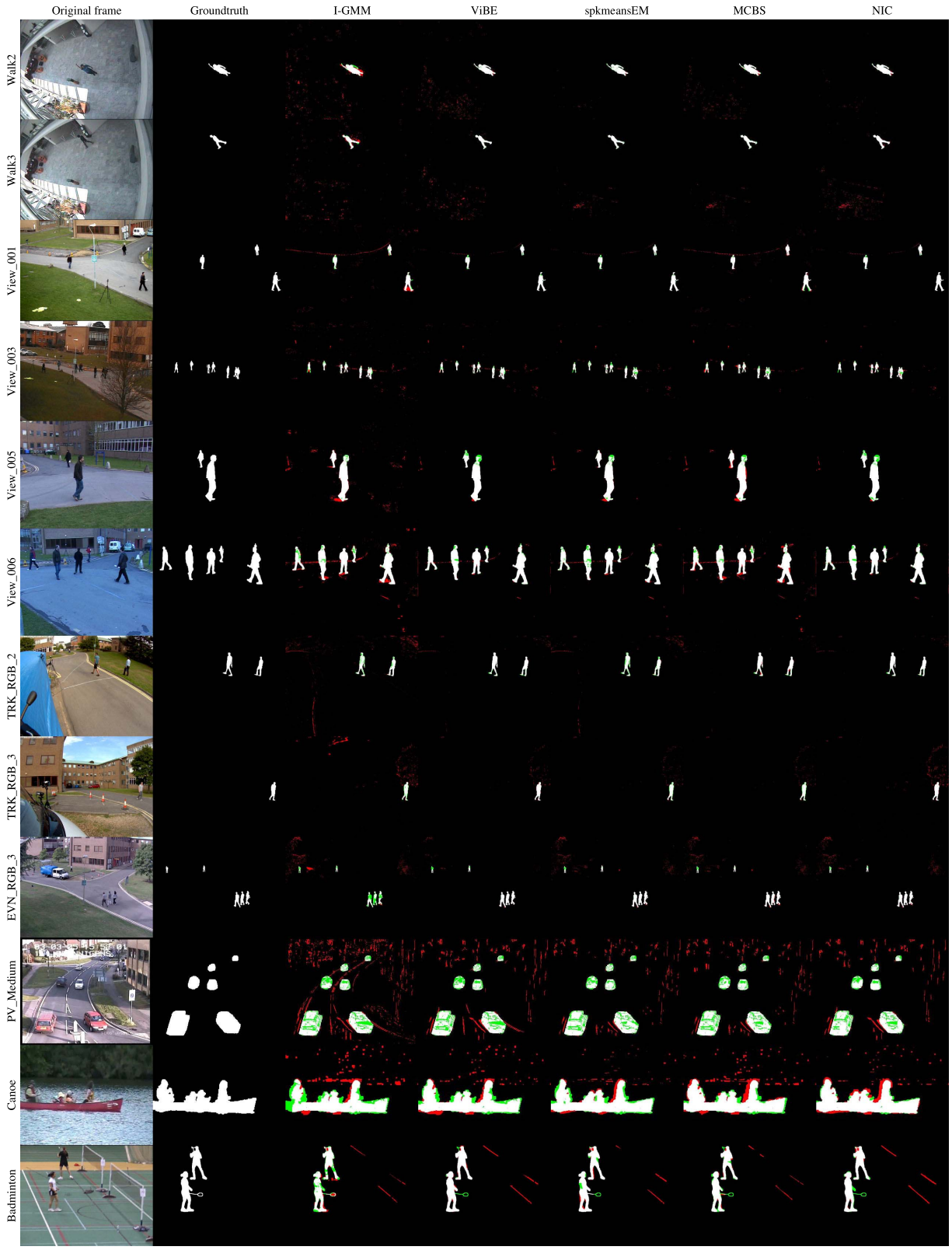


Fig. 5. Foreground detection results with true positive (TP) pixel in white, true negative (TN) pixel in black, false positive (FP) pixel in red, and false negative (FN) pixel in green on benchmarked sequences.

Badminton sequence are represented in Fig. 6. Using the same window size WS $= 5$, $\tau = 20$ provides higher TPR values in most test frames than $\tau = 30$, meanwhile, keeping the FPR at a lower level. If the size of the window is increased to 9, TPR might be increased with undesirable FPR increment and higher computational requirement.

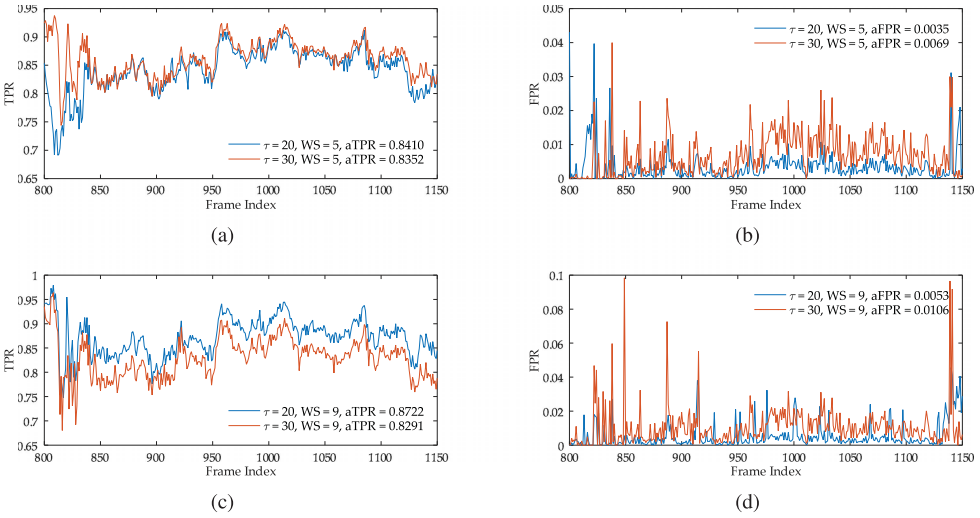


Fig. 6. TPRs and FPRs of the proposed method for the Badminton sequence using different parameter configurations. (a)-(b) TPRs and FPRs with WS = 5. (c)-(d) TPRs and FPRs with WS = 9.

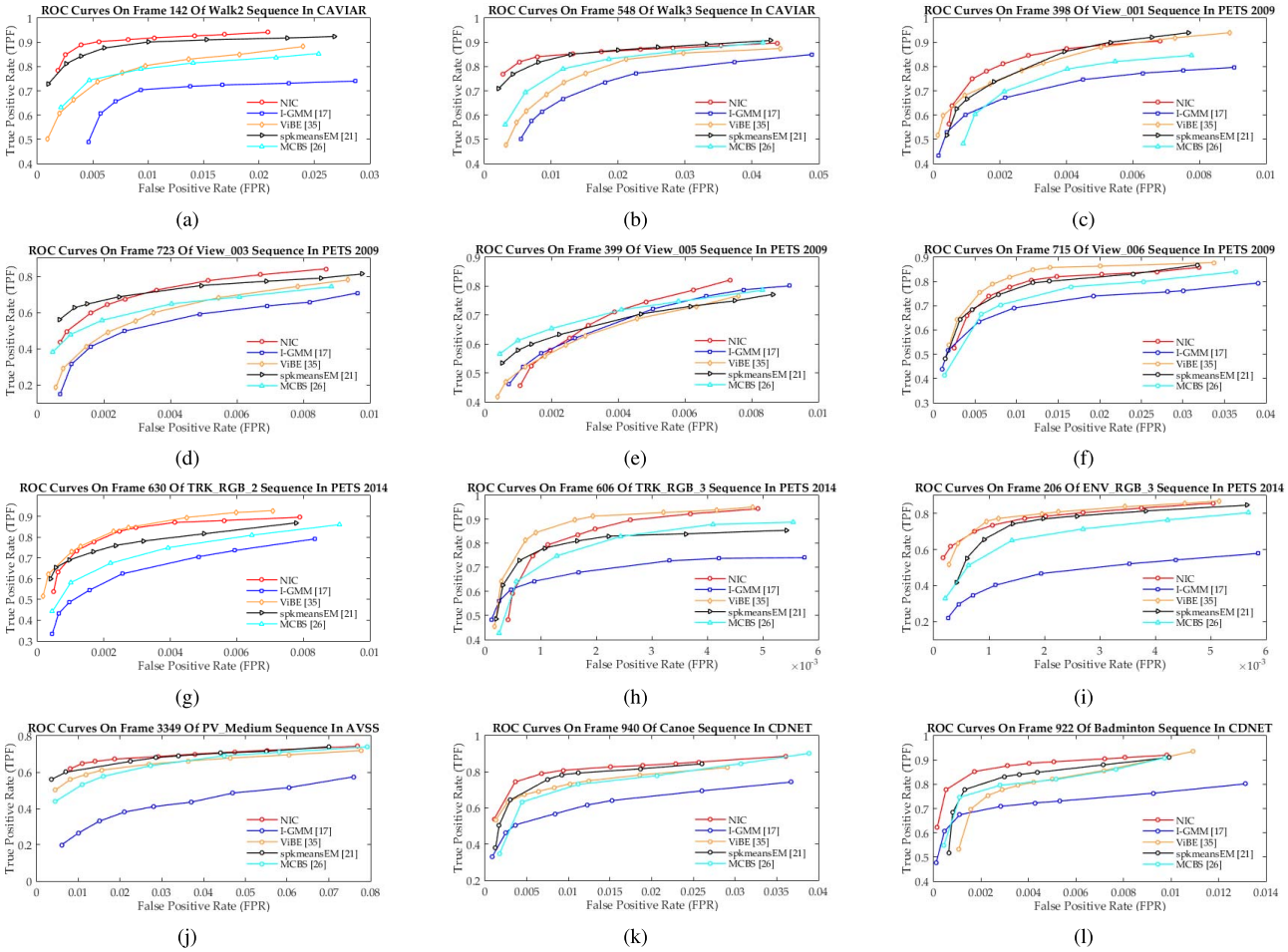


Fig. 7. ROC curves of the proposed method with different parameter configurations and the state-of-the-art methods on the benchmarked data sets: (a)-(b) CAVIAR 2004. (c)-(f) PETS 2009. (g)-(i) PETS 2014. (j) AVSS 2007. (k)-(l) CDNET.

E. Comparison With Existing Foreground Detection Methods

In this section, we compare the proposed method with four of the best-known foreground detection methods: 1) the I-GMM in [17]; 2) the original ViBE in [35]; 3) the spherical K-means expectation-maximization method (spkmeansEM) in [21]; and 4) the multilayer codebook model (MCBS) in [26]. All of the benchmarked algorithms are realized by our own implementation.

The I-GMM algorithm is employed with the number of components $M = 4$ and different threshold values c_{thr}.

TABLE II
COMPARISON OF PROCESSING SPEEDS EXPRESSED IN FRAMES PER SECOND

Sequence	Resolution	I-GMM	ViBE	spkmeansEM	MCBS	NIC
Walk2	384×288	10.4	**17.2**	10.9	15.8	14.5
Walk3	384×288	12.3	**23.8**	12.5	22.9	19.6
View_001	768×576	6.8	**9.3**	8.0	8.8	8.6
View_003	768×576	6.5	**8.4**	7.8	8.1	7.6
View_005	768×576	6.6	7.5	**8.3**	7.7	8.1
View_006	768×576	6.0	**8.7**	8.4	7.9	7.1
TRK_RGB_2	1280×960	2.6	**4.5**	3.3	3.2	3.4
TRK_RGB_3	1280×960	2.5	**3.4**	2.8	3.0	2.9
ENV_RGB_3	1280×960	2.6	**4.8**	4.6	4.2	3.7
PV_Medium	720×576	6.9	**8.6**	7.2	7.6	6.6
Canoe	320×240	14.5	**22.6**	13.3	16.2	15.2
Badminton	720×480	7.8	**10.4**	8.8	9.4	8.8

In the ViBE algorithm, we fix the default values as per recommendation in [35] comprising the time subsampling factor $\phi = 16$ and the cardinality of the set intersection $\#_{\min} = 2$, and vary the number of distances $N = \{5, 10, 20\}$ and the radius of the sphere $R = \{5, 10, 20\}$. We evaluate the spkmeansEM algorithm with the following parameters for the outdoor case: 1) $K_{\max} = 5$; 2) $\eta = 0.005$; 3) $d = 2$; 4) $\beta = 10^{-6}$; 5) $T \in [2, 5]$; 6) $\mathrm{TH}_P = 0.1$; 7) $\mathrm{TH}_I = 70$; 8) $\mathrm{TH}_D \in [18, 20]$; and 9) $\tau = 0.5$. With MCBS approach, the learning α and proportion of codeworks η are varied $\{0.05, 0.1\}$ and $\{0.5, 0.7, 0.9\}$ in the background model construction, while $\beta = 1.25$, $\gamma = 0.7$, and $\theta_{\text{color}} = 3$ are fixed as recommendation in the pixel classification stage [26]. Because the test sequences are rifle with objects moving constantly, NIC algorithm is evaluated for different parameter configurations including only $\tau = \{10, 20, 30\}$ and $\mathrm{WS} = \{5, 7, 9\}$.

Both quantitative and qualitative comparisons for all these methods are presented in Figs. 5 and 7. The I-GMM method constantly updates the parameters and simultaneously selects the appropriate number of components for each pixel using recursive equations. This method is weak under the strong light changes of the CAVIAR data set and the dynamic backgrounds in the PETS 2014 and CDNET data sets. Although I-GMM can get quite high TPR results in some testing sequences, its FPR values are also high. The ViBE method uses random neighbor selection to correct pixel intensity and the lifespan policy to update the background model over time. The background estimated by ViBE is more robust against noise than GMM-based models. Based on a combination of spherical K-means clustering and the expectation-maximization algorithm, spkmeansEM is quite efficient in extracting foreground from sequences that suffer from camera jittering and dynamic backgrounds. In the MCBS approach, constructing a multilayer codebook with various block sizes at the same time brings satisfactory results. Compared with above foreground detection schemes, NIC is the winner in most testing sequences, except *View_005* and *View_006* in PETS 2009 and *TRK_RGB_3* in PETS 2014.

F. Computational Cost

In this section, we analyze and compare the processing time of the proposed method and others in terms of frames per second. Concretely, we use a profiling tool included in MATLAB 2013a to measure the time required for background estimation and foreground detection. From the results in Table II, it can be observed that larger frames generally require more time for processing. The proposed method detects the foreground faster than I-GMM for most of the sequences and achieves a speed equivalent to that of spkmeansEM and MCBS. NIC requires more time for calculating the standard deviation instead of the Euclidean distance in ViBE. For MCBS, it must be noted that the results do not involve the codebook construction time.

V. CONCLUSION

In this paper, we propose an NIC algorithm for robust background estimation. The NIC algorithm calculates the standard deviations from two windows, respectively, extracts the background and the current frame, and identifies the class of the center pixel as background or object for intensity modification. The algorithm uses a steadiness factor to measure the changeability of each pixel and assess the robustness of the current background. This measurement reduces the computational cost by reducing the number of pixels that require intensity modification. Compared with existing background modeling algorithms, NIC is more flexible and adaptive with medium- and high-speed motion because the window size is adjustable. In the detection stage, the background produced by NIC is taken as the input for the background subtraction technique. We calculated an adaptive threshold using the Otsu method to segment objects based on the frame difference. The proposed method outperforms state-of-the-art methods (I-GMM, ViBE, spkmeansEM, and MCBS) on most of the test data sets by ROC curve comparisons. The limitations of our method include poor detection performance of repetitive motions and relatively high computational cost at the estimation stage. In the future, we will focus on identifying unexpected background motions for removal and optimizing the method to make it adaptive in real-time environments.

REFERENCES

[1] J. C. Nascimento and J. S. Marques, "Performance evaluation of object detection algorithms for video surveillance," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 761–774, Aug. 2006.
[2] H. Liu and X. Hou, "Moving detection research of background frame difference based on Gaussian model," in *Proc. Int. Conf. Comput. Sci. Service Syst. (CSSS)*, 2012, pp. 258–261.
[3] S. Denman, V. Chandran, and S. Sridharan, "An adaptive optical flow technique for person tracking systems," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1232–1239, Jul. 2007.
[4] A. Sobral and A. Vacavant, "A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos," *Comput. Vis. Image Understand.*, vol. 122, pp. 4–21, May 2014.
[5] A. Mittal and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2. Jun./Jul. 2004, pp. II-302–II-309.
[6] T. Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Comput. Sci. Rev.*, vols. 11–12, pp. 31–66, May 2014.
[7] T. Bouwmans, F. El Baf, and B. Vachon, "Background modeling using mixture of Gaussians for foreground detection—A survey," *Recent Patents Comput. Sci.*, vol. 1, no. 3, pp. 219–237, 2008.
[8] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, Nov. 2004.

[9] K. O. De Beeck, I. Y. H. Gu, L. Li, M. Viberg, and B. D. Moor, "Region-based statistical background modeling for foreground object segmentation," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 3317–3320.

[10] J. Peng and J. WeiDong, "Statistical background subtraction with adaptive threshold," in *Proc. 5th Int. Congr. Image Signal Process. (CISP)*, Oct. 2012, pp. 123–127.

[11] A. Akula, N. Khanna, R. Ghosh, S. Kumar, A. Das, and H. K. Sardana, "Adaptive contour-based statistical background subtraction method for moving target detection in infrared video sequences," *Infr. Phys. Technol.*, vol. 63, pp. 103–109, Mar. 2014.

[12] C. R. Jung, "Efficient background subtraction and shadow removal for monochromatic video sequences," *IEEE Trans. Multimedia*, vol. 11, no. 3, pp. 571–577, Apr. 2009.

[13] N. A. Mandellos, I. Keramitsoglou, and C. T. Kiranoudis, "A background subtraction algorithm for detecting and tracking vehicles," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 1619–1631, Mar. 2011.

[14] H. Zhou, Y. Chen, and R. Feng, "A novel background subtraction method based on color invariants," *Comput. Vis. Image Understand.*, vol. 117, no. 11, pp. 1589–1597, Nov. 2013.

[15] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Jun./Jul. 1999, p. 252.

[16] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[17] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 2. Aug. 2004, pp. 28–31.

[18] D.-S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 827–832, May 2005.

[19] A. Shimada, D. Arita, and R. I. Taniguchi, "Dynamic control of adaptive mixture-of-Gaussians background model," in *Proc. IEEE Int. Conf. Video Signal Based Surveill. (AVSS)*, Nov. 2006, p. 5.

[20] A. Elqursh and A. Elgammal, "Online moving camera background subtraction," in *Proc. 12th Eur. Conf. Comput. Vis. (ECCV)*, vol. 4. 2012, pp. 228–241.

[21] D. Li, L. Xu, and E. D. Goodman, "Illumination-robust foreground detection in a video surveillance system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 10, pp. 1637–1650, Oct. 2013.

[22] Y. Xie, "Improved Gaussian mixture model in video motion detection," *J. Multimedia*, vol. 8, no. 5, pp. 527–533, Oct. 2013.

[23] T. S. F. Haines and T. Xiang, "Background subtraction with Dirichlet process mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 670–683, Apr. 2014.

[24] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground-background segmentation using codebook model," *Real-Time Imag.*, vol. 11, no. 3, pp. 172–185, 2005.

[25] A. Ilyas, M. Scuturici, and S. Miguet, "Real time foreground-background segmentation using a modified codebook model," in *Proc. 6th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2009, pp. 454–459.

[26] J.-M. Guo, C.-H. Hsia, Y.-F. Liu, M.-H. Shih, C.-H. Chang, and J.-Y. Wu, "Fast background subtraction based on a multilayer codebook model for moving object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 10, pp. 1809–1821, Oct. 2013.

[27] C.-W. Lin, W.-J. Liao, C.-S. Chen, and Y.-P. Hung, "A spatiotemporal background extractor using a single-layer codebook model," in *Proc. 11th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2014, pp. 259–264.

[28] Q. Shao, Z. Tang, and S. Han, "Hierarchical CodeBook for background subtraction in MRF," *Infr. Phys. Technol.*, vol. 61, pp. 259–264, Nov. 2013.

[29] Z. Tang, Z. Miao, Y. Wan, and J. Li, "Automatic foreground extraction for images and videos," in *Proc. 17th IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2010, pp. 2993–2996.

[30] D. Li, L. Xu, and E. Goodman, "A fast foreground object detection algorithm using kernel density estimation," in *Proc. IEEE 11th Int. Conf. Signal Process. (ICSP)*, Oct. 2012, pp. 703–707.

[31] J. M. McHugh, J. Konrad, V. Saligrama, and P.-M. Jodoin, "Foreground-adaptive background subtraction," *IEEE Signal Process. Lett.*, vol. 16, no. 5, pp. 390–393, May 2009.

[32] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proc. IEEE*, vol. 90, no. 7, pp. 1151–1163, Jul. 2002.

[33] Z. Liu, K. Huang, and T. Tan, "Foreground object detection using top-down information based on EM framework," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4204–4217, Sep. 2012.

[34] O. Barnich and M. Van Droogenbroeck, "ViBe: A powerful random technique to estimate the background in video sequences," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Taipei, Taiwan, Apr. 2009, pp. 945–948.

[35] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences," *IEEE Trans. Image Process.*, vol. 20, no. 6, pp. 1709–1724, Jun. 2011.

[36] S. Toral, M. Vargas, F. Barrero, and M. G. Ortega, "Improved sigma-delta background estimation for vehicle detection," *Electron. Lett.*, vol. 45, no. 1, pp. 32–34, Jan. 2009.

[37] M. Vargas, J. M. Milla, S. L. Toral, and F. Barrero, "An enhanced background estimation algorithm for vehicle detection in urban traffic scenes," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 3694–3709, Oct. 2010.

[38] D. Culibrk, O. Marques, D. Socek, H. Kalva, and B. Furht, "Neural network approach to background modeling for video object segmentation," *IEEE Trans. Neural Netw.*, vol. 18, no. 6, pp. 1614–1627, Nov. 2007.

[39] L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1168–1177, Jul. 2008.

[40] B.-H. Do and S.-C. Huang, "Dynamic background modeling based on radial basis function neural networks for moving object detection," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2011, pp. 1–4.

[41] M. Rohanifar and A. Amiri, "The background modeling for the video sequences using radial basis function neural network," in *Proc. 6th Int. Conf. Comput. Sci. Converg. Inf. Technol. (ICCIT)*, Nov./Dec. 2011, pp. 371–374.

[42] H. Bhaskar, L. Mihaylova, and A. Achim, "Video foreground detection based on symmetric alpha-stable mixture models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 8, pp. 1133–1138, Aug. 2010.

[43] L. Geng and Z. Xiao, "Real time foreground-background segmentation using two-layer codebook model," in *Proc. Int. Conf. Control, Autom. Syst. Eng. (CASE)*, Jul. 2011, pp. 1–5.

[44] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *Proc. 6th Eur. Conf. Comput. Vis. (ECCV)*, 2000, pp. 751–767.

[45] C. Guyon, T. Bouwmans, and E.-H. Zahzah, "Foreground detection based on low-rank and block-sparse matrix decomposition," in *Proc. 19th IEEE Int. Conf. Image Process. (ICIP)*, Sep./Oct. 2012, pp. 1225–1228.

[46] Z. Gao, L.-F. Cheong, and Y.-X. Wang, "Block-sparse RPCA for salient motion detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1975–1987, Oct. 2014.

[47] N. J. B. McFarlane and C. P. Schofield, "Segmentation and tracking of piglets in images," *Mach. Vis. Appl.*, vol. 8, no. 3, pp. 187–193, 1995.

[48] S. Gruenwedel, N. I. Petrovic, L. Jovanov, J. O. Niño-Casta-Ñeda, A. Pižurica, and W. Philips, "Efficient foreground detection for real-time surveillance applications," *Electron. Lett.*, vol. 49, no. 18, pp. 1143–1145, Aug. 2013.

[49] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2006.

[50] (2004). *CAVIAR: Context Aware Vision Using Image-Based Active Recognition*. [Online]. Available: http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/

[51] J. Ferryman. (2009). *PETS 2009 Benchmark Data*. [Online]. Available: http://www.cvg.reading.ac.uk/PETS2009/a.html

[52] L. Patino and J. Ferryman, "PETS 2014: Dataset and challenge," in *Proc. 11th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2014, pp. 355–360. [Online]. Available: http://pets2014.net/

[53] (2007). *i-Lids Dataset for AVSS*. [Online]. Available: http://www.eecs.qmul.ac.uk/andrea/avss2007_d.html

[54] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "CDnet 2014: An expanded change detection benchmark dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2014, pp. 393–400.

[55] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[56] Y. Yalman and I. Ertürk, "A new color image quality measure based on YUV transformation and PSNR for human vision system," *Turkish J. Elect. Eng. Comput. Sci.*, vol. 21, no. 2, pp. 603–612, 2013.
[57] L. Maddalena and A. Petrosino, "Towards benchmarking scene background initialization," in *New Trends in Image Analysis and Processing* (Lecture Notes in Computer Science), vol. 9281. Switzerland: Springer, 2015, pp. 469–476.

Thien Huynh-The (S'15) received the B.Eng. degrees from the Department of Electronic Engineering, Ho Chi Minh City University of Technical and Education, Ho Chi Minh City, Vietnam, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Kyung Hee University, Yongin, South Korea.

His current research interests include image processing, computer vision, and machine learning.

Oresti Banos (M'15) received the M.Sc. degrees in telecommunications engineering, computer network engineering, and electrical engineering, and the Ph.D. degree in computer science from the University of Granada, Granada, Spain, in 2009, 2010, 2011, and 2014, respectively.

He was a Research Associate with the Research Center for Information and Communications Technologies and a Lecturer with the Department of Computer Architecture and Computer Technology, University of Granada. He was a Visiting Researcher at several prestigious institutions, such as the Swiss Federal Institute of Technology in Zurich, Zurich, Switzerland, in 2011, the University of Alabama, Tuscaloosa, AL, USA, in 2011, and the Technical University of Eindhoven, Eindhoven, The Netherlands, in 2012. He is currently a Post-Doctoral Research Fellow with the Department of Computer Engineering, Kyung Hee University, Yongin, South Korea. His current research interests include wearable, ubiquitous, pervasive, and mobile computing with a particular focus on digital health and wellbeing applications, pattern recognition and machine learning for probabilistic modeling of human behavior from multimodal sensor data, information fusion, and context-awareness, with a special interest in robust, and adaptive and opportunistic expert systems.

Sungyoung Lee (M'89) received the B.S. degree from Korea University, Seoul, South Korea, and the M.S. and Ph.D. degrees in computer science from the Illinois Institute of Technology, Chicago, IL, USA, in 1987 and 1991, respectively.

He was an Assistant Professor with the Department of Computer Science, Governors State University, University Park, IL, USA, from 1992 to 1993. He has been a Professor with the Department of Computer Engineering, Kyung Hee University, Yongin, South Korea, since 1993, where he has also been the Director of the Neo Medical Ubiquitous Life Care Information Technology Research Center since 2006 and is currently a Founding Director of the Ubiquitous Computing Laboratory. His current research interests include ubiquitous computing and applications, wireless ad-hoc and sensor networks, context-aware middleware, sensor operating systems, real-time systems, and embedded systems.

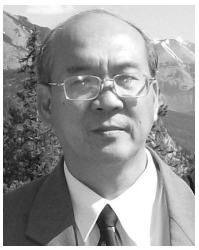
Byeong Ho Kang is currently an Associate Professor with the School of Computing and Information Systems, University of Tasmania, Hobart, TAS, Australia. He has been involved in research and development projects with industries and research organizations, the Smart Internet Collaborative Research Centre, Australia, the U.S. Air Force, Washington, DC, USA, Hyundai Steel, Seoul, South Korea, AutoEver Systems at Hyundai Corporation, LG-EDS, IPMS Technology, South Korea, and the Asian Office of Aerospace Research Department, Japan. He has also contributed to the development of several commercial Internet-based applications, AI products, an expert system development tool, intelligent help desk systems, and Web-based information monitoring and classification systems. He has authored over 200 papers in refereed journals, conference proceedings, and book chapters. His current research interests include basic knowledge acquisition methods and many applied research in Internet systems as well as medical expert systems.

Prof. Kang has been involved in many international conferences and workshops as the Conference Chair or a Program Committee Member. He has invited as a Keynote Speaker in several international conferences. He recently organized more than five international conferences in AI, Internet, and ubiquitous computing area.

Eun-Soo Kim received the Ph.D. degree in electronics engineering from Yonsei University, Seoul, South Korea, in 1984.

He joined as a Faculty Member with Kwangwoon University, Seoul, in 1981, where he is currently a Professor with the Department of Electronic Engineering. He has been engaged with about 200 M.S. and Ph.D. students. He has authored or co-authored over 500 papers in international journals and conferences and holds 50 patents. His current research interests include real 3D imaging and displays, 3D fusion technologies and their applications, 3D camera, and 3D information processing.

Dr. Kim was a recipient of numerous scientific and academic awards and the Order of Science and Technology Merit from R. Moo-Hyun, the President of Korea, in 2003. His research laboratory was honored as a National Research Laboratory of 3D Media by the Ministry of Science and Technology of Korea in 2000. In 2003, his research center of the 3D Display Research Center was also honored as an Information Technology Research Center by the Ministry of Communication and Information of Korea. He was the President of the Society of 3D Broadcasting and Imaging and the 3D Display Subsociety of the Korean Information Display Society. He is the President of the 3D Fusion Industry Consortium and the President-Elect of the Korean Information and Communications Society. He established the International Workshop on 3D Information Technology in 2004 and was the Co-Organizer of the event of International 3D Fair in collaboration with the 3D Consortium of Japan in 2006.



Thuong Le-Tien (M'96) was born in Ho Chi Minh City, Vietnam. He received the bachelor's and master's degrees in electrical engineering from the Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, in 1980 and 1995, respectively, and the Ph.D. degree in telecommunications from the University of Tasmania, Hobart, TAS, Australia, in 1998.

He has been a Faculty Member with the Telecommunications Department, HCMUT, since 1981. He was a Visiting Scholar with Ruhr University Bochum, Bochum, Germany, from 1989 to 1992. He served as the Deputy Department Head for many years and was the Telecommunications Department Head from 1998 to 2002. He was appointed as the Director of the Center for Overseas Studies from 1998 to 2010. He has authored over 130 research articles and many teaching materials for university students related to electronic circuits 1 and 2, digital signal processing and wavelets, antenna and wave propagation, and communication systems. His current research interests include communication systems, digital signal processing, and electronic circuits.

Dr. Le-Tien received the title as a National Distinguished Lecturer and various certificates for his engineering education contributions from the Academic State Council and the Chairman of National.