Pattern Recognition 48 (2015) 1374–1390

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Co-occurrence probability-based pixel pairs background model for robust object detection in dynamic scenes

Dong Liang [a,*], Shun'ichi Kaneko [a], Manabu Hashimoto [b], Kenji Iwata [c], Xinyue Zhao [d]

[a] Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan
[b] School of Information Science and Technology, Chukyo University, Aichi, Japan
[c] National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan
[d] Department of Mechanical Engineering, Zhejiang University, Zhejiang, China

ARTICLE INFO

Article history:
Received 5 April 2013
Received in revised form
4 July 2014
Accepted 9 October 2014
Available online 24 October 2014

Keywords:
Object detection
Sudden illumination fluctuation
Burst motion
Background modeling
Co-occurrence probability

ABSTRACT

An illumination-invariant background model for detecting objects in dynamic scenes is proposed. It is robust in the cases of sudden illumination fluctuation as well as burst motion. Unlike the previous works, it uses the co-occurrence differential increments of multiple pixel pairs to distinguish objects from a non-stationary background. We use a two-stage training framework to model the background. First, joint histograms of co-occurrence probability are employed to screen supporting pixels with high normalized correlation coefficient values; then, K-means clustering-based spatial sampling optimizes the spatial distribution of the supporting pixels; finally the background model maintains a sensitive criterion with few parameters to detect foreground elements. Experiments using several challenging datasets (PETS-2001, AIST-INDOOR, Wallflower and a real surveillance application) prove the robust and competitive performance of object detection in various indoor and outdoor environments.

1. Introduction

Detecting moving objects plays a very important role in an intelligent surveillance system. It is often integrated with various tasks, such as tracking objects [1,2], recognizing their behaviors [3,4] and alerting when abnormal events occur [5]. However, object detection suffers from non-stationary scenes in surveillance videos, especially in two potentially serious cases: (1) sudden illumination variation, such as outdoor sunlight changes and indoor lights turning on/off; (2) burst physical motion, such as the motion of indoor artificial objects, which include fans, escalators and auto-doors. If the actual background includes a combination of any of these factors, it becomes even more difficult to perform detection. State-of-the-art algorithms [6–10] can handle gradual illumination changes by updating the statistical background models progressively as time goes by. In practice, however, this kind of model update is usually relatively slow to avoid mistakenly integrating foreground elements into the background model, making it difficult to adapt to sudden illumination changes and burst motion.

In this study, we propose a novel framework to build a background model for object detection, which is brightness-invariant and able to tolerate burst motion. We name it Co-occurrence Probability-based Pixel Pairs (CP3). It is inspired by the previous work in [11,12]. In the work of Haralick et al. [11], gray-level co-occurrence matrices (GLCM) were employed to measure the spatial co-occurrence of pixels to produce an image texture feature (Haralick feature). In the work of Hashimoto and Saito [12], pixels with low spatial co-occurrence probability and with high temporal co-occurrence probability were preferentially extracted as spatially distinctive and temporally stable features to reduce computational complexity for template matching. In this study, in order to model the dynamic background, spatial pixel pairs with high temporal co-occurrence probability are employed to represent each other by using the stable intensity differential increment between a pixel pair which is much more reliable than the intensity of a single pixel, especially when the intensity of a single pixel changes dramatically over time. A pixel pair consists of each pixel itself (called *target pixel* hereafter) and a selected pixel (called *supporting pixel* hereafter). As a pixel-wise background model, the target pixel P refers to all pixels in a scenario. The supporting pixels are neither arbitrary pixels in the scene, nor predefined fixed local structures around each target pixel; instead, the supporting pixels are selected based on their statistical stability with the target pixels.

* Corresponding author.
E-mail addresses: liang@ssc.ssi.ist.hokudai.ac.jp, donglcn@hotmail.com (D. Liang).

http://dx.doi.org/10.1016/j.patcog.2014.10.020
0031-3203/

The remainder of this paper is organized as follows. In the next section, some related works are discussed. Section 3 details the background model. Section 4 presents the object detection procedure. Section 5 presents the experimental results, and Section 6 concludes the main contributions of this work.

2. Related work

Since observations of the background in image sequences can be considered stochastic events, many statistical approaches have been employed to model effective backgrounds. The former background modeling approaches can be classified into two categories: (1) independent pixel-wise modeling, which employs the statistical processing of time-domain observations to each pixel. (2) Spatial-dependence modeling, which employs principles to exploit spatial-dependence among pixels to build a local or global model.

Most of the earlier background modeling approaches tend to fall into the first category. Wren [6] modeled the observations (YUV) of each pixel as a single Gaussian probability density function. To cope with periodic moving background patterns, the Gaussian mixture model (GMM) [7,13] was proposed. Elgammal [8] employed kernel density estimation (KDE) as a data-driven modeling method. Since KDE is a non-parametric model, it is closer to the real probability distribution than GMM. Hidden Markov models (HMMs) [14,15] have also been applied to model the background; topology free HMMs were described and several state splitting criteria were compared in the context of background modeling in [14], and a non-adaptive three-state HMM was used to model the background in [15]. The recent notable pixel-wise method by Kim [9] presented a real-time algorithm, which sampled background pixel values and quantized them into compressed codebooks (CBs). To improve the processing efficiency of the codebooks, Guo [16] presented a hierarchical scheme. All the above methods use a learning rate function for updating the background model online. However, because none of these methods is free from erroneous updating, they have a well-known trade-off problem: with a low learning rate, they can not adapt to sudden changes of illumination, e.g., turning on/off a light, while with a high learning rate, slowly moving objects or temporarily stopped objects will be detected as background.

The second category uses spatial information to exploit the spatial dependencies of pixels in the background. Matsuyama [17] proposed a regional block matching method against varying illumination, and Seki [18] proposed a co-occurrence-based block correlation method. The above two methods can only yield coarse region-level detection. Toyama et al. [19] proposed a three layers algorithm in which Weiner filters were employed. It used region and frame-level information to verify the pixel-wise background model. Oliver [20] employed eigenspace decomposition in which the background was modelled by the eigenvectors corresponding to the largest eigenvalues. Sheikh [10] used the joint representation of image pixels in a local spatial distribution (proximal pixels) and colour information to build both background and foreground KDE models competitively in a decision framework. Monnet [21] and Zhong [22] built an auto-regressive moving average (ARMA) model in dynamic scenes, which is used to incrementally learn (using PCA) and then predict motion patterns in the scene. Heikkilä and Pietikäinen [23] used a local binary pattern (LBP) to subtract the background and detect moving objects in real time. This method models each pixel as a group of adaptive LBP histograms that were calculated over a predefined circular region around the pixel. Similarly, the statistical reach feature (SRF) [24] builds a local texture model for each target pixel to be brightness-invariant. A recent spatial-dependence approach [25] utilized a tensor subspace learning algorithm to represent spatial correlations between pixel values, and modeled appearance changes by incrementally learning a tensor subspace representation by adaptively updating the sample mean and an eigenbasis for each unfolding matrix of the tensor.

In our previous research, we proposed a background model called grayscale arranging pairs (GAP) [26,27] which falls into the second category. GAP employed an alignment of supporting pixels for the target pixel which held a stable intensity subtraction in training frames without any restriction of locations. The intensity subtraction of the pixel pairs allowed the background model to tolerate noise and be illumination-invariant. However, this fixed intensity subtraction influenced the sensitivity of the background model, especially when the dynamic range was compressed due to low illumination; it was also not an optimal way to search for supporting pixels by using a fixed intensity subtraction in that most co-occurrence pixels were not considered. In addition, the GAP method mainly focused on illumination-invariance, so that the dynamic background caused by burst motion was not discussed sufficiently. In this study, the proposed method addresses these open problems. Compared with GAP, the proposed method employs a co-occurrence histogram to describe the relationship of a pixel pair, which is free from any intensity differences, and calculates normalized correlation coefficients for measuring the degree of co-occurrence which can deal with a dynamic background. It also introduces a spatial clustering operation to select optimal supporting pixels and then provides a more accurate parameterized detection criterion instead of a fixed double-sided threshold.

3. Background modeling

The algorithm is described for gray-scale imagery; however, it can also be used for colour or multi-modality imagery with minor modification. Fig. 1 shows the fundamental definitions of the image data. Suppose we are given a training image sequence $B=\{I_1, I_2, \ldots, I_T\}$ with a total of T images, and each image has $M=U\times V$ pixel positions. In the three-dimensional space $\Gamma=\{(u,v,t)|1\le u\le U, 1\le v\le V, 1\le t\le T\}$, we have $U\times V\times T$ intensity values within a gray-scale level range $[0, L-1]$. In the following, the intensities over time at each pixel position are regarded as samples from a stochastic process. We define P as a *target pixel* at location (u, v). The location of P varies to cover all pixels of a frame, and its intensity sequence over time is denoted as $\{p_t(u,v)\}_{t=1,2,\ldots,T}$. In the same way, we define $Q(u',v')$ as an *arbitrary pixel* with intensity sequence $\{q_t(u',v')\}_{t=1,2,\ldots,T}$ at location (u', v'). For simplicity, we have omitted most of the (u, v) and (u', v') in the following discussion.

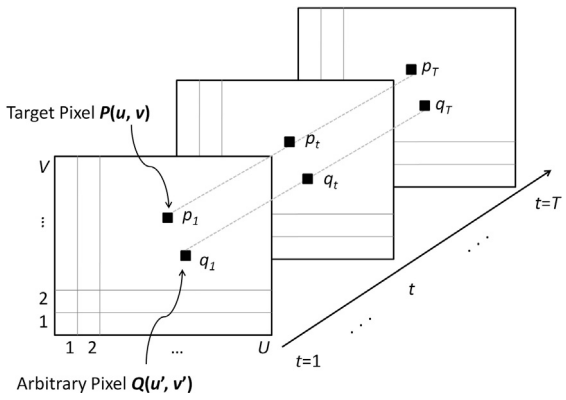

Fig. 1. Fundamental definitions of the image data. Target pixel P and an arbitrary pixel Q with their intensity sequence over time p_t and q_t.

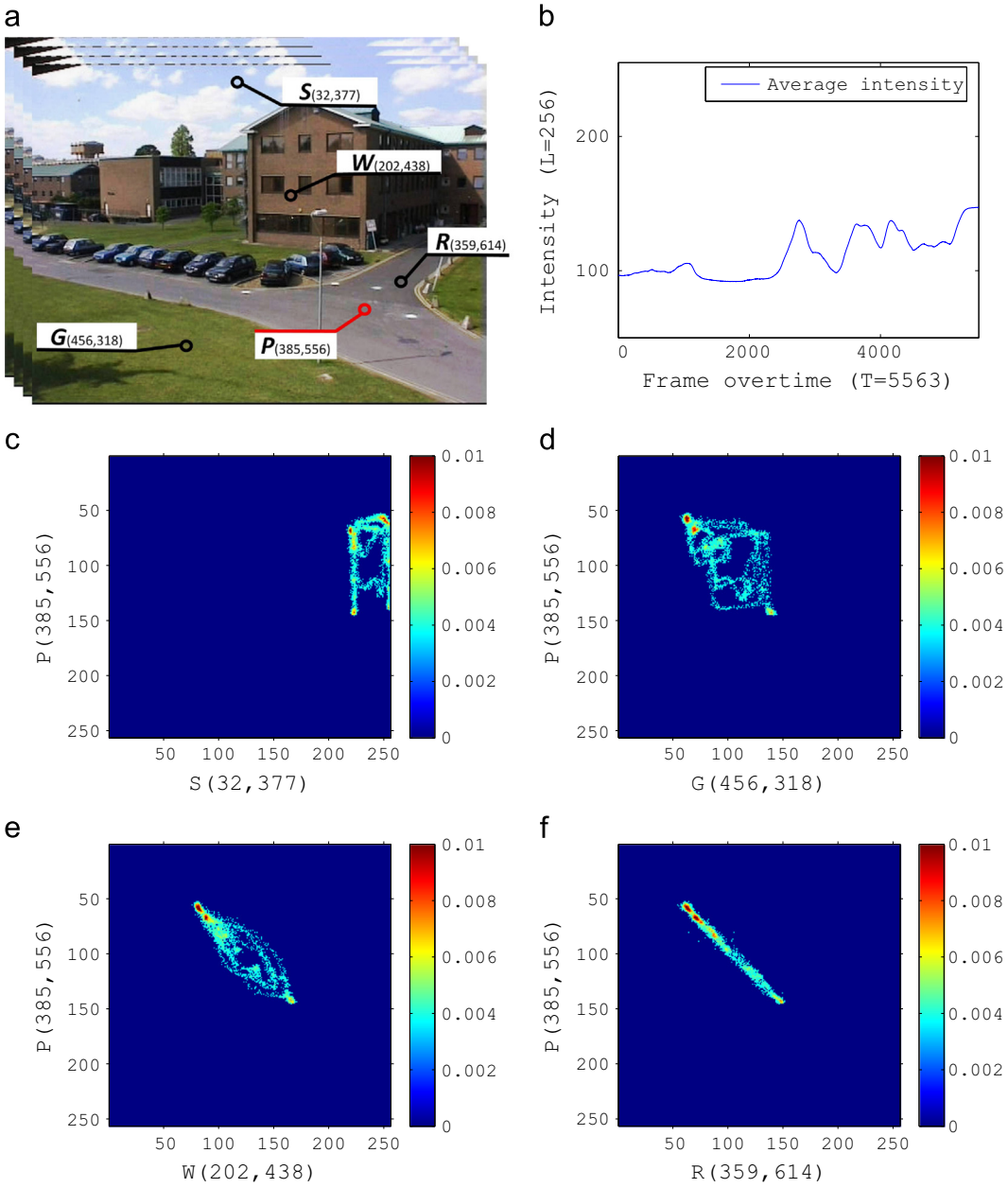

Fig. 2. Co-occurrence joint histograms use PETS2001-dataset3 camera1 training dataset with 5563 frames ($T=5563$) and the gray-scale level range is [0, 255] ($L=256$). (a) Location of the target pixel P selected from the road, and four arbitrary pixels S, G, W, and R, selected from the sky, grass, wall, and road respectively. (b) The average intensity change of each frame in this scenario. (c)–(f) The joint histograms $\boldsymbol{h}_{PS}$, $\boldsymbol{h}_{PG}$, $\boldsymbol{h}_{PW}$, and $\boldsymbol{h}_{PR}$ of each pixel pair.

3.1. Bivariate statistical property of a pixel pair

To analyze the bivariate statistical property of a pixel pair, the joint histogram of intensity for a pixel pair is defined.[1] The i,jth bin of the joint histogram for an arbitrary pixel pair (P,Q) in T training images can be expressed as

$$h_{PQ}(i,j)=\sum_{t=1}^{T}\delta(p_t,q_t,i,j), \tag{1}$$

where $\delta(p_t,q_t,i,j)$ represents the two-dimensional discrete Kronecker delta function:

$$\delta(p_t,q_t,i,j)=\begin{cases}1 & \text{if } (p_t=i)\cap(q_t=j)\\ 0 & \text{otherwise}\end{cases}.$$

The bins $h_{PQ}(i,j)$ corresponding to $i,j\in[0,L-1]$ represent the co-occurrence probability of $p_t=i$ and $q_t=j$. The joint histogram $\boldsymbol{h}_{PQ}$ can be written compactly as an ordered array,

$$\boldsymbol{h}_{PQ}=\{h_{PQ}(i,j)\}_{i,j=0}^{L-1}. \tag{2}$$

As an example using PET2001-dataset3 camera1 (ftp://ftp.pets.rdg.ac.uk/pub/PETS2001/DATASET3/) shown in Fig. 2, we selected a target pixel P located on the "road", and four arbitrary pixels S, G, W and R from the "sky", "grass", "wall" and "road" respectively. The section $h_{PQ}(i,j)>0$ of joint histograms are illustrated as a few

[1] CP3 can also be called the pixel correlation model. However, we call it Co-occurrence Probability-based Pixel Pairs because it is designed on the basis of co-occurrence histogram of a pixel pair, and the following calculations also rely on the probability of the histogram bins.

representable examples in Fig. 2(c–f). $\boldsymbol{h}_{PS}$ shows the most irregular distribution, while $\boldsymbol{h}_{PG}$, $\boldsymbol{h}_{PW}$ and $\boldsymbol{h}_{PR}$ reveal a more regular distribution. It is obvious that the most regular distribution is the histogram $\boldsymbol{h}_{PR}$ shown in Fig. 2(f), and its co-occurrence bins are parallel to a diagonal line running downwards through the histogram. The corresponding intensity time sequences of the four pixel pairs are shown in Fig. 3. In Fig. 3(b and c), the intensity changes show an unexpected phase difference between the two time sequences, which disturbs the stable distribution of the joint histograms $\boldsymbol{h}_{PG}$ and $\boldsymbol{h}_{PW}$ shown in Fig. 2(d, e). For example, when a light source is shifting repeatedly, the shift speed would vary between different times, which would cause such a phase difference. The corresponding joint histograms in Fig. 2(d and e) indicate that, under illumination fluctuation, the intensity relationships of the pixel pair (P,G) and (P,W) follow multiple varying modes, which essentially releases the relationship constraint of the pixel pair. If we employ such a pixel pair to model the background, the sensitivity to detect objects will be low, because the background model would be so flexible that a large range of intensity pairs would gather to model illumination changes. On the other hand, plenty of pixels have a simpler and more explicit statistical relationship with the target pixel as shown by the pixel pair (P,R) in Fig. 2(f). Therefore, R can be selected as a *supporting pixel* Q^P to observe the intensity of P even under illumination changes. The training stage of the proposed method starts from selecting supporting pixels Q^P for each target pixel P, the associated calculation procedure is presented in the next subsection.

As a grayscale/single-channel image, the change in pixel intensity is proportional to the illumination increment. Hence, the statistical linearity of a pixel pair reduces to a stable intensity differential increment $\Delta(p_t,q_t)$ just as the example in Fig. 2(f), in which the slope of the regression line approaches 1. For one target pixel P, it is natural to expect that one or more pixels Q^P which maintain a stable intensity differential increments $\Delta(p_t,q_t)$ during training frames, exist even though P and Q^P might be at quite different locations. When detecting objects under a dynamic background, both the object occupation and the illumination change (or other forms of dynamics) can affect on the current intensity of a target pixel P. A background model only modeling the independent intensity change of P cannot distinguish the object from the dynamic background. Instead, Q^P pixels could be employed to estimate the intensity of the target pixel P in a current detection frame, i.e. $\hat{p}=\Delta(p_t,q_t)+q$, where q is the intensity of Q^P in a current detection frame. When the illumination changes on a target pixel P but no object exists on it, $\hat{p}$ simultaneously changes with q, so that the current intensity p will fall into the estimated range $\hat{p}$, then P will be considered to be a background element. If P is occupied by an object, the current intensity p will be out of the estimated range $\hat{p}$, then P will be considered to be occupied by an object. In our training stage, $\Delta(p_t,q_t)$ is modeled as a single Gaussian model, with a unique mean and variance. Using a Gaussian model allows the pixel pair to tolerate noise. More importantly, for different pixel pairs, the mean of $\Delta(p_t,q_t)$ could vary and the noise effect could also be at different ranges, so the mean and variance of a Gaussian calculated on specific pixel pair provide an accurate statistical constraint between them.

For robust detection, it is necessary to maintain a sufficient number of *supporting pixels*, denoted $\{Q_k^P\}_{k=1,2,\ldots,K}$, where K is the total number of supporting pixels. In our detection stage, a double layer probability-based decision is used to detect objects: the first layer is to identify whether an individual pixel pair (P,Q_k^P) matches its Gaussian model; and the second layer is to observe a matched ratio, i.e. to count the number of pixel pairs $(P,\{Q_k^P\})$ that can match their Gaussian model from total K pixel pairs for each P.

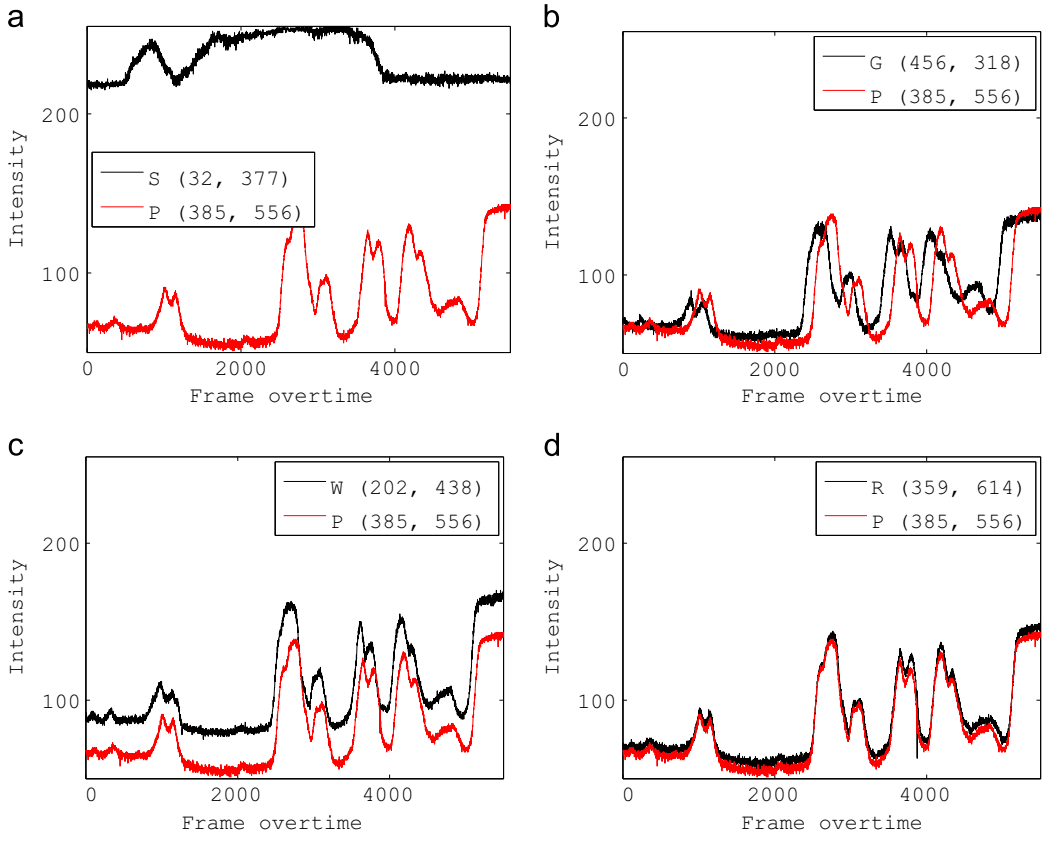

Fig. 3. Intensity change of target pixel P and four arbitrary pixels corresponding to Fig. 2(b–e).

Once the matched ratio decreases, P will be regarded as a foreground element. The associated calculation procedure is presented in Section 4.

3.2. Statistical measurement of co-occurrence pixel pairs

In this section, we introduce how to determine the supporting pixels from $M-1$ candidate pixels for each target pixel P. For an arbitrary pixel pair (P,Q), the one-dimensional histograms corresponding to their marginal probability distributions are

$$h_P(i)=\sum_{j=0}^{L-1}h_{PQ}(i,j) \tag{3}$$

and

$$h_Q(j)=\sum_{i=0}^{L-1}h_{PQ}(i,j). \tag{4}$$

The expectation values of P and Q are $\mathcal{E}(p_t)=T^{-1}\sum_{i=0}^{L-1}ih_P(i)$ and $\mathcal{E}(q_t)=T^{-1}\sum_{j=0}^{L-1}jh_Q(j)$ respectively, and their variances are

$$\sigma_{p_t}^2=\frac{1}{T}\sum_{i=0}^{L-1}[i-\mathcal{E}(p_t)]^2h_P(i) \tag{5}$$

and

$$\sigma_{q_t}^2=\frac{1}{T}\sum_{j=0}^{L-1}[j-\mathcal{E}(q_t)]^2h_Q(j). \tag{6}$$

The covariance of a (P,Q) pair can be defined as follows:

$$\mathcal{C}_{P,Q}=\frac{1}{T}\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}[i-\mathcal{E}(p_t)][j-\mathcal{E}(q_t)]h_{PQ}(i,j). \tag{7}$$

$\mathcal{C}_{P,Q}>0$ means P and Q has positive covariance value, which indicates they potentially have a high co-occurrence probability. We call this kind of pixel pair co-occurrence pixel pair hereafter.

In order to measure the independent co-occurrence quantitatively, we utilize Pearson product-moment correlation coefficient:

$$\gamma_{(P,Q)}=\frac{\mathcal{C}_{P,Q}}{\sigma_{p_t}\cdot\sigma_{q_t}} \tag{8}$$

where σ_{p_t} and σ_{q_t} are the standard deviations of P and Q respectively. Eq. (8) is used to estimate the linear dependence of spatial pixel pairs, and $\gamma_{(P,Q)}$ is ±1 in the case of a perfect positive/negative linear correlation. This normalized correlation coefficient does not involve the restraint between $\mathcal{E}(p_t)$ and $\mathcal{E}(q_t)$. Namely, the pixels Q with any mean values are likely to have large $\gamma_{(P,Q)}$ with P. Fig. 4 shows four examples of $\gamma_{(P,Q)}$ using PETS2001-dataset3, the black crosses stand for the location of P, and the red coloured area have high correlation coefficient values.

In practice, Eq. (8) can be calculated based on a correlation matrix instead of calculating pixel-by-pixel serial processing. The correlation matrix is the covariance matrix of the standardized random variables $\tilde{p}_t=p_t/\sigma(p_t)$. With a total of $M=U\times V$ pixel positions, the image sequence can be arranged progressively as a column vector set $\chi^M=\{\tilde{p}_t(m)\}_{m=1,2,\ldots,M}$. The correlation matrix of size $M\times M$ is

$$\begin{aligned}\Upsilon(\chi^M)&=\mathcal{C}(\chi^M,(\chi^M)^T)\\&=\begin{bmatrix}\tilde{p}_t(1)-\mathcal{E}(\tilde{p}_t(1))\\ \tilde{p}_t(2)-\mathcal{E}(\tilde{p}_t(2))\\ \vdots\\ \tilde{p}_t(M)-\mathcal{E}(\tilde{p}_t(M))\end{bmatrix}\cdot\begin{bmatrix}\tilde{p}_t(1)-\mathcal{E}(\tilde{p}_t(1))\\ \tilde{p}_t(2)-\mathcal{E}(\tilde{p}_t(2))\\ \vdots\\ \tilde{p}_t(M)-\mathcal{E}(\tilde{p}_t(M))\end{bmatrix}^T\end{aligned} \tag{9}$$

where $\mathcal{C}(\cdot)$ is the covariance operation. The correlation matrix is symmetric so that each row and column of the $\Upsilon(\chi^M)$ is an array of $\gamma_{(P,Q)}$ for each $P(u,v)$.

For each target pixel $P(u,v)$, $M-1$ values of $\gamma_{(P,Q)}$ need to be calculated at different locations (u',v'). Then Q_n corresponding to the highest N components in the array $\gamma_{(P,Q(u',v'))}$ can be selected as

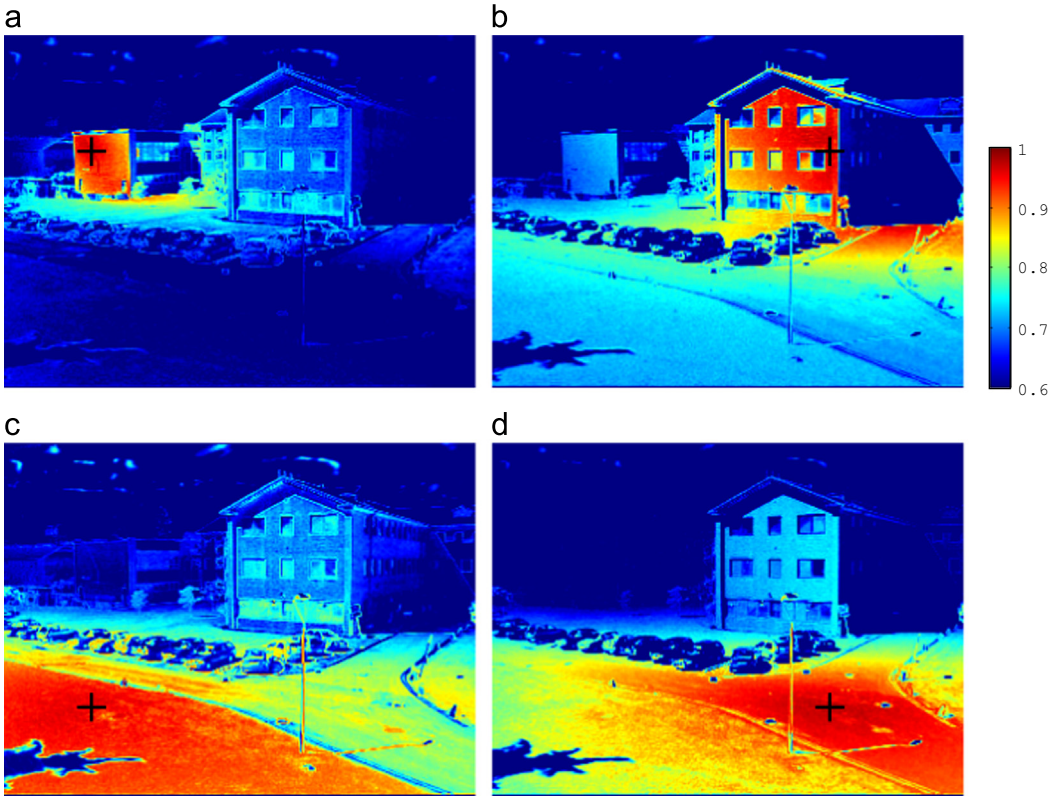

Fig. 4. Diagram of $\gamma_{(P,Q)}$ using PETS2001-dataset3. The black cross markers are the locations of P. (For interpretation of the reference to color in this figure caption, the reader is referred to the web version of this paper.)

the candidates of preferred supporting pixels, namely

$$\{Q_n\} = \{Q(u', v') | \gamma_{(P,Q)} > \check{\gamma}\}, \quad n = 1, 2, \ldots, N, \tag{10}$$

where N is the number of candidate supporting pixels Q_n for each target pixel P, and $\check{\gamma}$ is the lower limit for the co-occurrence pixel pair. $\check{\gamma}$ is an adaptive threshold depending on the variation characteristic of P, which is discussed in Section 3.3.

3.3. Lower limit for a co-occurrence pixel pair

Due to sensor noise and encoding noise, any p_t and q_t cannot maintain a full co-occurrence relation. Therefore, the lower limit $\check{\gamma}$ for choosing the high co-occurrence pixel pairs is a key parameter. This parameter is generally solved through preparatory experiments using typical samples of target images as a training dataset. To estimate the lower limit more reasonably, we need a theoretical formalization as a guidance with a natural prospective view.

Our approach to formalization is to assume that, $p_t = p'_t + e_1$ and $q_t = q_t' + e_2$, where p'_t and q_t' are the intensities without any noise; e_1 and e_2 are the additive noise, independent of each other but with the same density function $\mathcal{N}(0, \sigma_n^2)$. Then we assume p'_t and q_t' have a perfect positive linear correlation with a constant $b = \Delta(p'_t, q_t')$, namely $p'_t = q_t' + b$, and analyse $\check{\gamma}$ as a statistic for investigating how large degradation is raised by the noise. For the computation of $\gamma_{(P,Q)}$, disconcordance between p_t and q_t can degrade the $\check{\gamma}$ value. The correlation coefficient $\check{\gamma}$ can be represented by the next expression according to Eq. (8),

$$\check{\gamma} = \frac{C(p'_t + e_1, p'_t + e_1 - e_2 - b)}{\sigma_{p'_t + e_1} \cdot \sigma_{p'_t + e_1 - e_2 - b}}. \tag{11}$$

According to the properties of covariance and variance, the above formula is developed as

$$\check{\gamma} = \frac{\sigma_{p'_t}^2 + \sigma_n^2}{\sigma_{p'_t + e_1} \cdot \sigma_{p'_t + e_1 - e_2 - b}}. \tag{12}$$

When p'_t is independent of e, Eq. (12) can be rewritten as

$$\check{\gamma} = \frac{\sigma_{p'_t}^2 + \sigma_n^2}{[(\sigma_{p'_t}^2 + \sigma_n^2)(\sigma_{p'_t}^2 + 2\sigma_n^2)]^{1/2}} = \left(\frac{\sigma_{p'_t}^2 + \sigma_n^2}{\sigma_{p'_t}^2 + 2\sigma_n^2}\right)^{1/2} \tag{13}$$

$$\check{\gamma} = \left(\frac{\sigma_{p_t}^2}{\sigma_{p_t}^2 + \sigma_n^2}\right)^{1/2} = \left(1 + \frac{\sigma_n^2}{\sigma_{p_t}^2}\right)^{-1/2}. \tag{14}$$

When the noise level is significantly smaller than the dynamic range of p_t, namely $\sigma_{p'_t}^2 \gg \sigma_n^2$, Eq. (13) approximates to 1, which reveals that with large-scale intensity variation in the training dataset, the noise effect for correlation measurement can be reduced. On the other hand, if the intensities of P keep steady, namely $\sigma_{p'_t}^2 \to 0$, Eq. (13) will level off to $1/\sqrt{2}$, then the candidate supporting pixels can be selected from the stationary elements of the background. We directly use Eq. (14) instead of Eq. (13), in which $\sigma_{p_t}^2$ can be calculated based on Eq. (5) and σ_n^2 can be determined according to the noise level of the image sequence. From the theoretical analysis, the lower limit can be determined from the comprehensive conditions combining with σ_n^2 which can be easily provided by users and a computable $\sigma_{p_t}^2$. Fig. 5 shows four

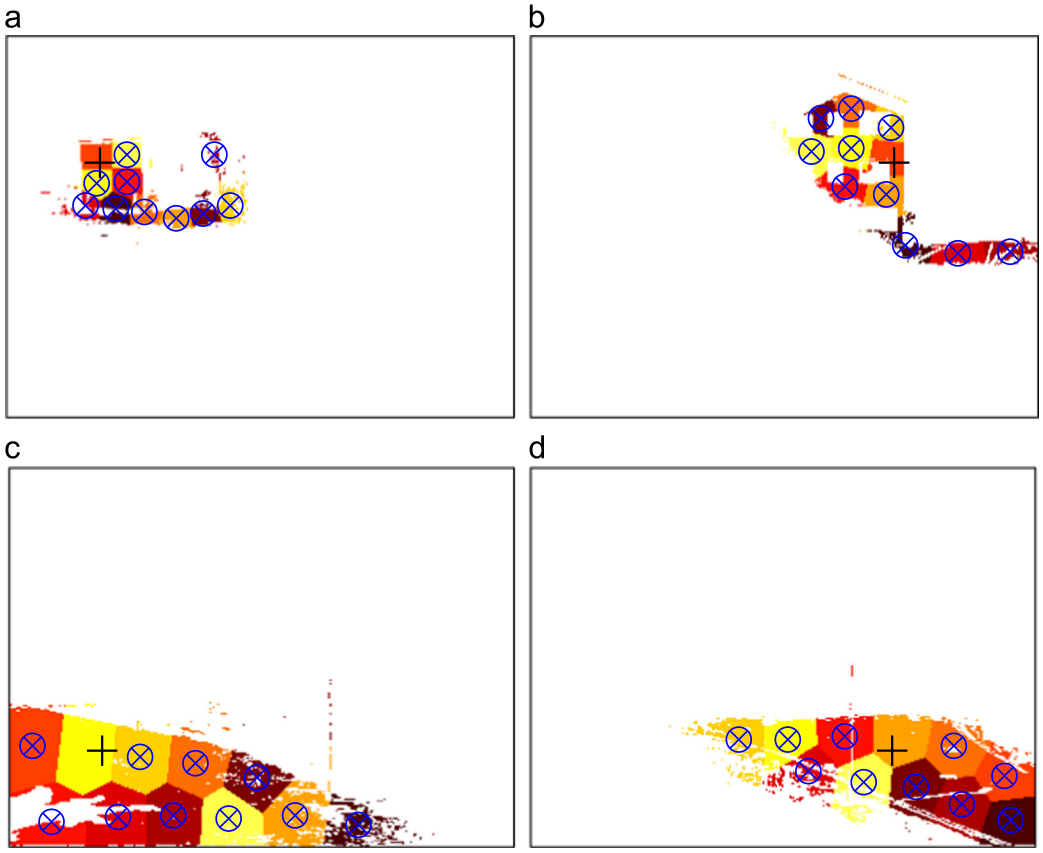

Fig. 5. Diagram of selecting Q_k^P using K-means in spatial domain. The black markers correspond to the P pixels in Fig. 4(a–d). The different-coloured areas, which stand for the clustering subsets, come from the high co-occurrence pixel Q_n with total N pixels. The blue circles are the centres of each clustering subset, which are selected as Q_k^P. In these examples, $K = 10$. (For interpretation of the reference to color in this figure caption, the reader is referred to the web version of this paper .)

example of selected Q_n (the coloured area), which demonstrate that the rules to choose Q_n according to the lower limit $\check{\gamma}$ allow the spatial distributions of Q_n to follow irregular illumination variation patterns.

3.4. Background model of pixel pairs

In this section, we discuss how to produce a limited number of supporting pixels Q_k^P and build the background model.

In Section 3.3, the selected Q_n using $\check{\gamma}$ resulted in an indeterminate number of N. To perform an executable algorithm, we need to confirm a series of a limited number of supporting pixels. As the spatial distribution of Q_n follows irregular patterns, we cannot implement any former spatial interpolation approach to select high representative Q_k^P from Q_n. To solve this issue, K-means clustering is employed to partition Q_n into K clusters, depending on the nearest clustering centres [28]. With clustering convergence, the pixel that is closest to the k-th cluster centre is selected as a unique Q_k^P. The details of K-means clustering for optimizing the spatial distribution is described in the Appendix. Four demonstrations of the Q_k^P optimization are shown in Fig. 5 in which $K=10$. It is reasonable to assume that selecting more supporting pixels will contribute to a robust result. The supporting pixels are essentially a group of statistical samples. Theoretically speaking, a larger number of samples would estimate a more robust statistical model. On the other hand, the number of supporting pixels directly affects the computation cost for object detection. This issue is discussed in the form of quantitative experiments in Section 5.2. Without loss of generality, the number of K for a given video scene is set at 10 for the experimental comparison in Sections 5.1 and 5.3.

Each Q_k^P keeps a bivariate differential increment with P,

$$p_t \sim \mathcal{N}(q_{t(k)}+b, \sigma_\varepsilon^2), \tag{15}$$

where σ_ε^2 follows a normalized distribution $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. We use this Gaussian function to model the intensity distribution of a pixel pair instead of a Gaussian mixture model [7] because we find that a single Gaussian works better since the selected pixel pair keeps highly steady difference except for noise. We calculate a noise standard deviation estimation as follows,

$$\hat{\sigma}_\varepsilon = \sigma_{p_t - q_{t(k)}}, \tag{16}$$

and the estimation of the differential increment b is,

$$\hat{b} = \mathcal{E}[p_t - q_{t(k)}]. \tag{17}$$

After the training step, the above two parameters $\hat{\sigma}_\varepsilon$, $\hat{b}$ are recorded for the following detection procedure. The background model is a look-up table (LUT) consisting of $[u', v', \hat{\sigma}_\varepsilon, \hat{b}]$ for $\{Q_k^P\}_{k=1,2,\ldots,K}$. The pseudo-code of CP3 for background modeling is shown in Algorithm 1.

Algorithm 1. CP3 for background modeling.

Data: $B=\{I_1, I_2, \ldots, I_T\}$ with total T images, and σ_n^2.
Result: look-up table of $\{Q_k^P\}_{k=1,2,\ldots,K}$ consisting of $[u', v', \hat{\sigma}_\varepsilon, \hat{b}]$.
Initialization:
Build the vector set $\chi^M = \{\tilde{p}_t(m)\}_{m=1,2,\ldots,M}$.
(I) Compute correlation matrix:

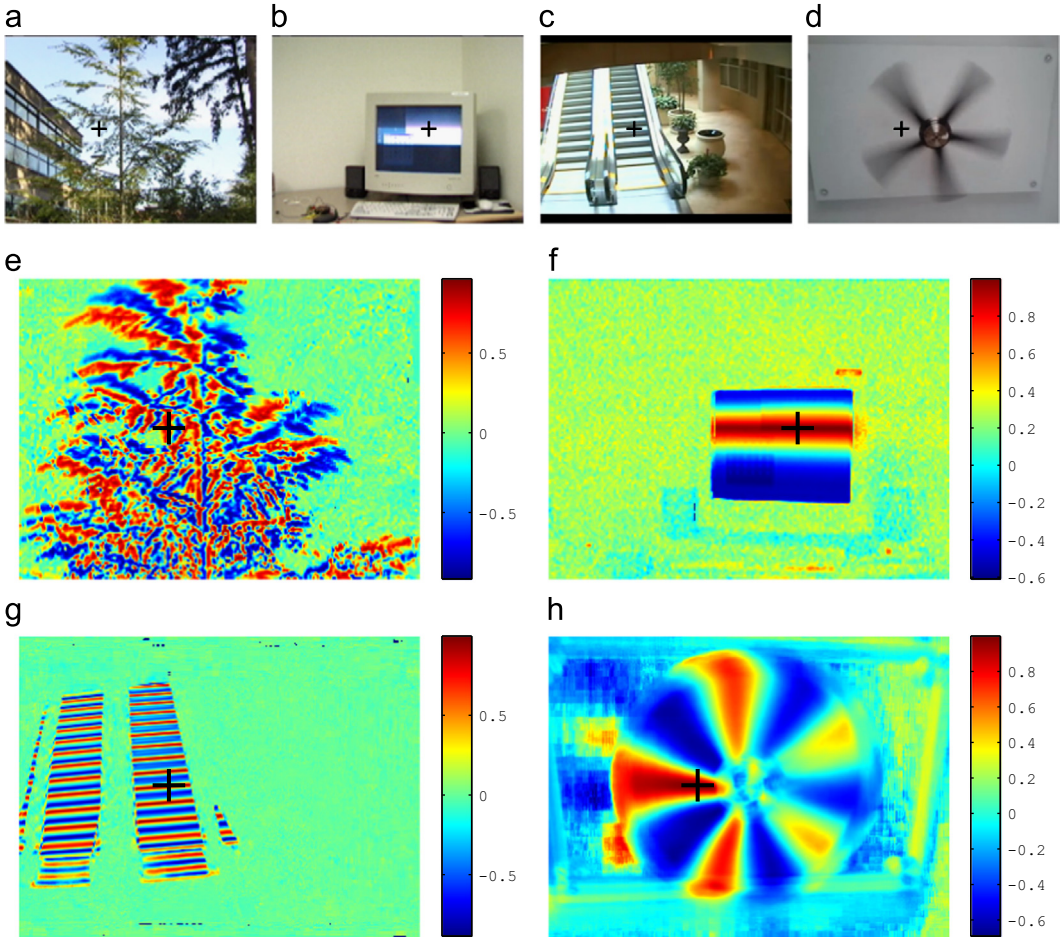

Fig. 6. Examples of various burst motion. (a) Tree swing. (b) Dynamic horizontal lines of a displayer. (c) Auto-induction escalator. (d) Speed-adjustable fan. (e-h) $\gamma_{(P,Q)}$ values of a selected target pixel in (a-d).

$Y(\chi^M) = C(\chi^M, (\chi^M)^T)$.
(II) Select supporting pixels:
for each $P(u, v)$ **do**
(a) Compute $\check{\gamma}$ based on Eq.(14).
(b) $\{Q_n\}_{n=1,2,\ldots,N} = \{Q | \gamma_{(P,Q)} > \check{\gamma}\}$.
(c) K−means sampling in spatial domain $\{Q_n\} \Rightarrow \{Q_k^P\}_{k=1,2,\ldots,K}$.
(d) Compute and record $[u', v', \hat{\sigma}_\varepsilon, \hat{b}]$ for $\{Q_k^P\}_{k=1,2,\ldots,K}$.

To train the background model of a scene, a training dataset including the dynamic background of the scene is necessary. In this work, we use three public datasets (PETS2001-dataset3 camera1, AIST-INDOOR, Wallflower), and each of them provides a separate training and detection dataset. We use each specified training and detection dataset for training and detection respectively.

3.5. Moving background case

In the case of a moving background, the moving parts cover several pixels in the same frame that also present co-occurrence. Therefore, we can search for the supporting pixels Q_k^P if the intensity changes of the pixel pairs are simultaneous. Using the proposed method to evaluate the intensity changes caused by a moving background makes no difference within the case of illumination variation. Hence, the earlier discussion based on illumination change is also appropriate for the case of a moving background.

A typical motion pattern in backgrounds is *burst motion*. This motion pattern can be described as a moving part of the background following regular directions but with an irregularly scheduled occurrence; hence, its speed and frequency can not be directly predicted. Plenty of moving background elements can be simplified into burst

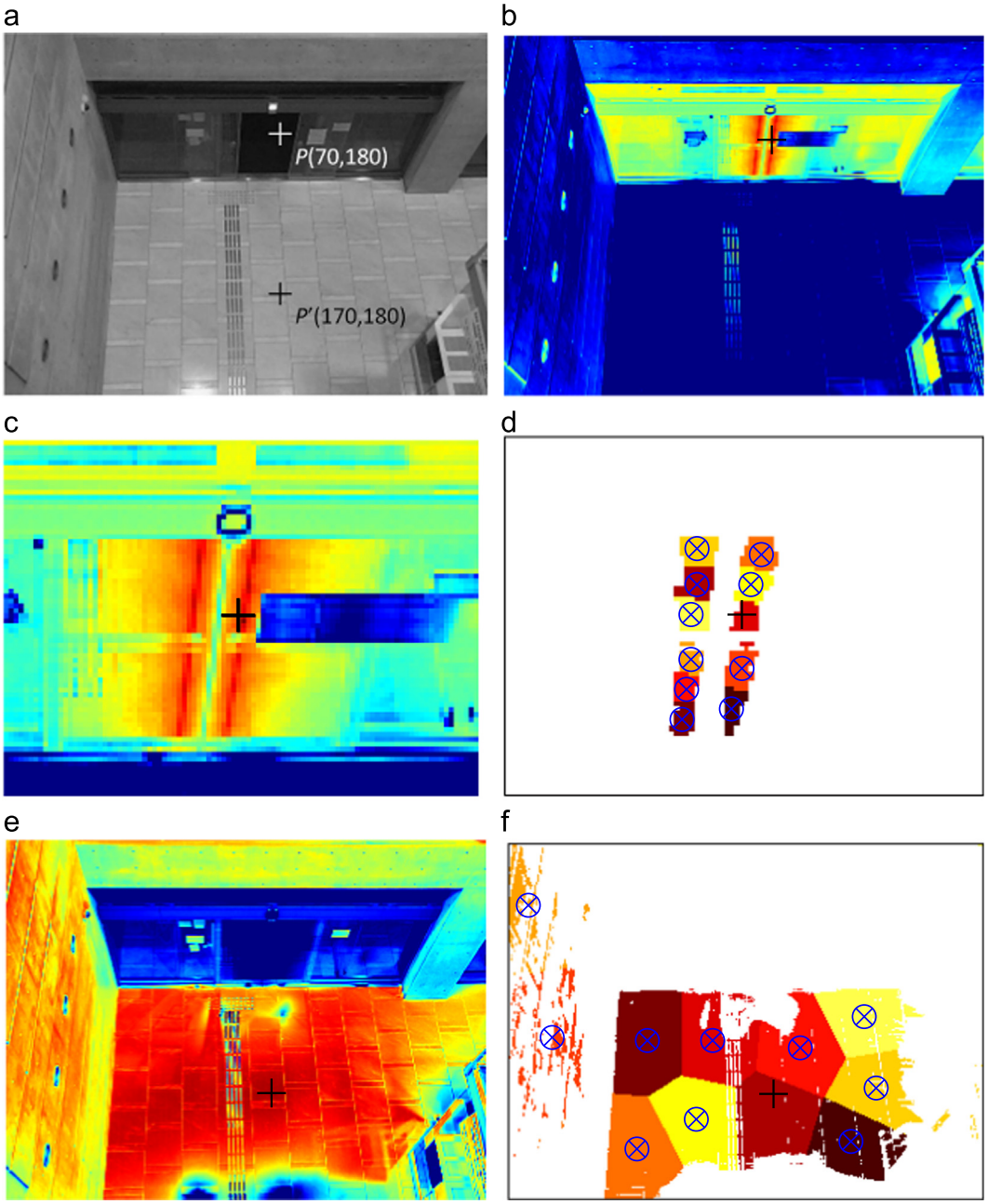

Fig. 7. Examples of the mixture of burst motion and sudden illumination change. (a) Location of the target pixels $P(70, 180)$ and $p'(170, 180)$. (b) $\gamma_{(P,Q)}$ of $P(70, 180)$. (c) Partial enlarged drawing of (b). (d) The supporting pixels of (c). (e) $\gamma_{(P,Q)}$ of $P'(170, 180)$. (f) The supporting pixels of (e).

motion. For example, tree swinging caused by random wind, fan speed change, dynamic horizontal lines of a displayer caused by its refresh rate, and auto-induction escalators and doors, all of which often appearing in outdoor/indoor surveillance scenes are examples of the burst motion. In general, applying independent pixel-wise methods (such as GMM [7] or Codebook [9]) to deal with motion backgrounds only employ pixel's history by continuously updating background, using a fixed learning rate as the background updating criterion; these background models are sensitive to burst motion. Our proposed method is a frequency and speed adaptive background model, which employs the spatial-dependence of pixel pairs to keep a stable differential increment regardless of the intensity of a single pixel under any frequency or speed of burst motion. The selected pixel pairs convert the non-stationary scene to a stable background model for offsetting the patterns of motion without any learning rate. Fig. 6 shows four examples of various burst motion backgrounds without severe illumination change. Fig. 6(a) and (b) is from the Wallflower dataset [19], and contain a waving tree and a cathode ray tube

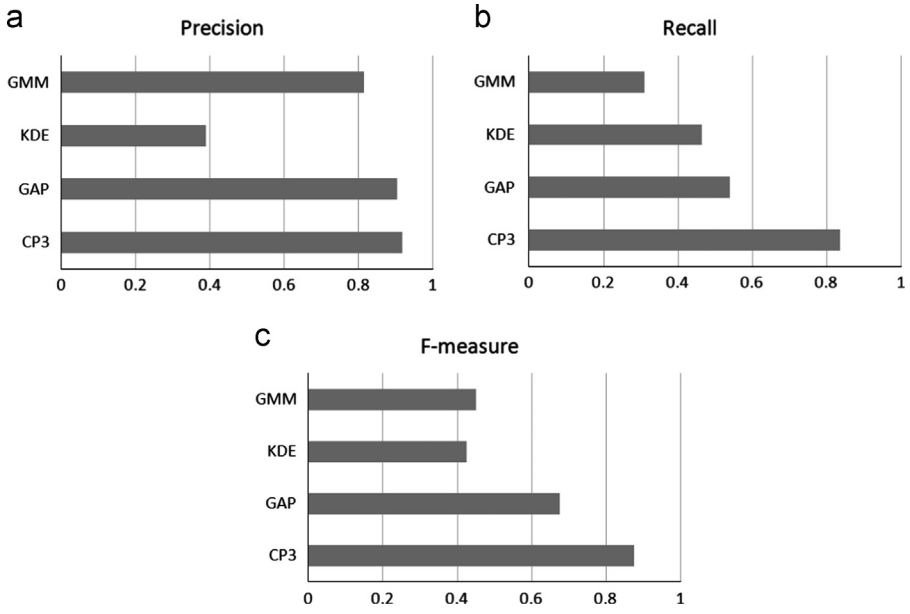

Fig. 8. The mean metrics of GMM, KDE, GAP and proposed CP3 using PETS2001-dataset3 camera1. (a) *Precision*, (b) *Recall* and (c) *F*-measure.

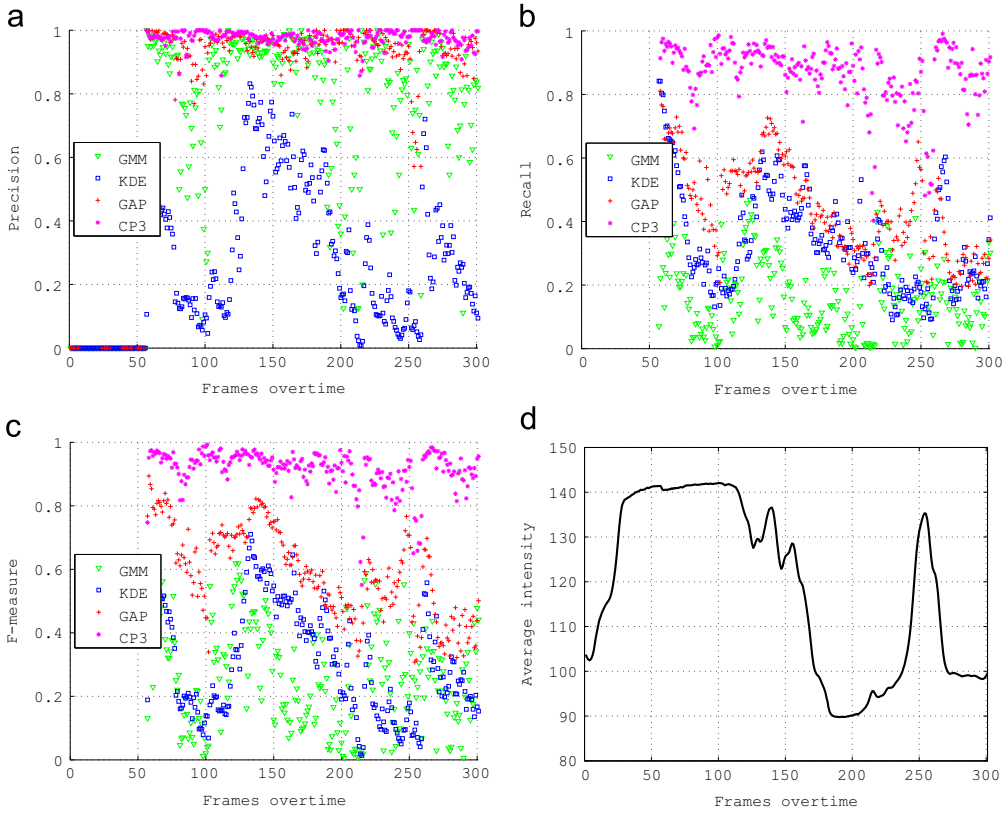

Fig. 9. (a) *Precision*, (b) *Recall* and (c) *F*-measure of CP3, GAP, KDE and GMM of PETS2001-dataset3 camera1. (d) The average intensities over time through 301 testing frames.

displayer respectively; Fig. 6(c) and (d) contains an escalator in operation and a fan in operation respectively, which can be downloaded from the author's web links: http://ssc-lab.com/ liang/CP3_project/escalator.rar and http://ssc-lab.com/~liang/CP3_project/fan.rar. Compared with the case of severe illumination change shown in Fig. 4, $\gamma_{(P,Q)}$ appear a part of negative values along the vertical direction of the movement locus shown in Fig. 6. Fig. 7 shows examples of the mixture case of burst motion and sudden illumination change using AIST-INDOOR dataset (http://ssc-lab.com/~liang/CP3_project/AIST_INDOOR_DATASET.rar). In this scene, a target pixel repeatedly passes by an auto-induction door while a light turning on. The supporting pixels with high co-occurrence are located along the vertical direction of the door to meet the simultaneity of its burst motion (shown in Fig. 7(b–d)), rather than around a regular neighborhood. As a comparison, Fig. 7(e and f) shows the case of a target pixel in a static part of the same scene.

3.6. Accelerated background modeling

In Section 3.2, in order to calculate $\gamma_{(P,Q)}$ for each P in a training dataset with $M = U \times V$ pixel positions and T frames, the computational complexity for Eq. (9) is $O(TM^2)$. In contrast, in the independent pixel-wise models [7,8], it is unnecessary to consider spatial dependence, so the computational complexity is only $O(TM)$ in the training stage. With the same memory cost, the time consumption of our proposed method is $M-1$ times higher than independent pixel-wise models.

Hoping to have a faster version of background modeling, we modified Eq. (9) using a hierarchical structure of a covariance-matrix: χ^M can be sampled uniformly using an integral sample interval Λ, the sub-set $\chi^{[M/\Lambda^2]} \subset \chi^M$; thus, we have

$$\Upsilon(\chi^{[M/\Lambda^2]}) = C(\chi^{[M/\Lambda^2]}, (\chi^{[M/\Lambda^2]})^T). \tag{18}$$

In order to cover all the target pixels, we have Λ^2 hierarchical correlation matrices $\Upsilon(\chi_\lambda^{[M/\Lambda^2]})$ and

$$\chi_\lambda^{[M/\Lambda^2]} = \{\tilde{p}_t(\omega\Lambda^2+\lambda)\}_{\omega=1,2,\ldots,[M/\Lambda^2]}, \tag{19}$$

where $\lambda = 1, 2, \ldots, \Lambda^2$. In this way, both the memory cost and time consumption is $O(TM^2\Lambda^{-2})$, which means the hierarchical structure of a covariance-matrix reduces computational complexity by Λ^2. The speed-up algorithm is based on the pixel sampling operation, which reduces the number of candidates of supporting pixels. However, we are willing to accept the possible loss in information, because it allows us to achieve high speed processing when dealing with high resolution surveillance videos. We recommend that the high resolution applications choose the accelerated background modeling, and the low data-volume applications choose the standard CP3 algorithm. The experimental discussion of the sample interval Λ is presented in Section 5.3.

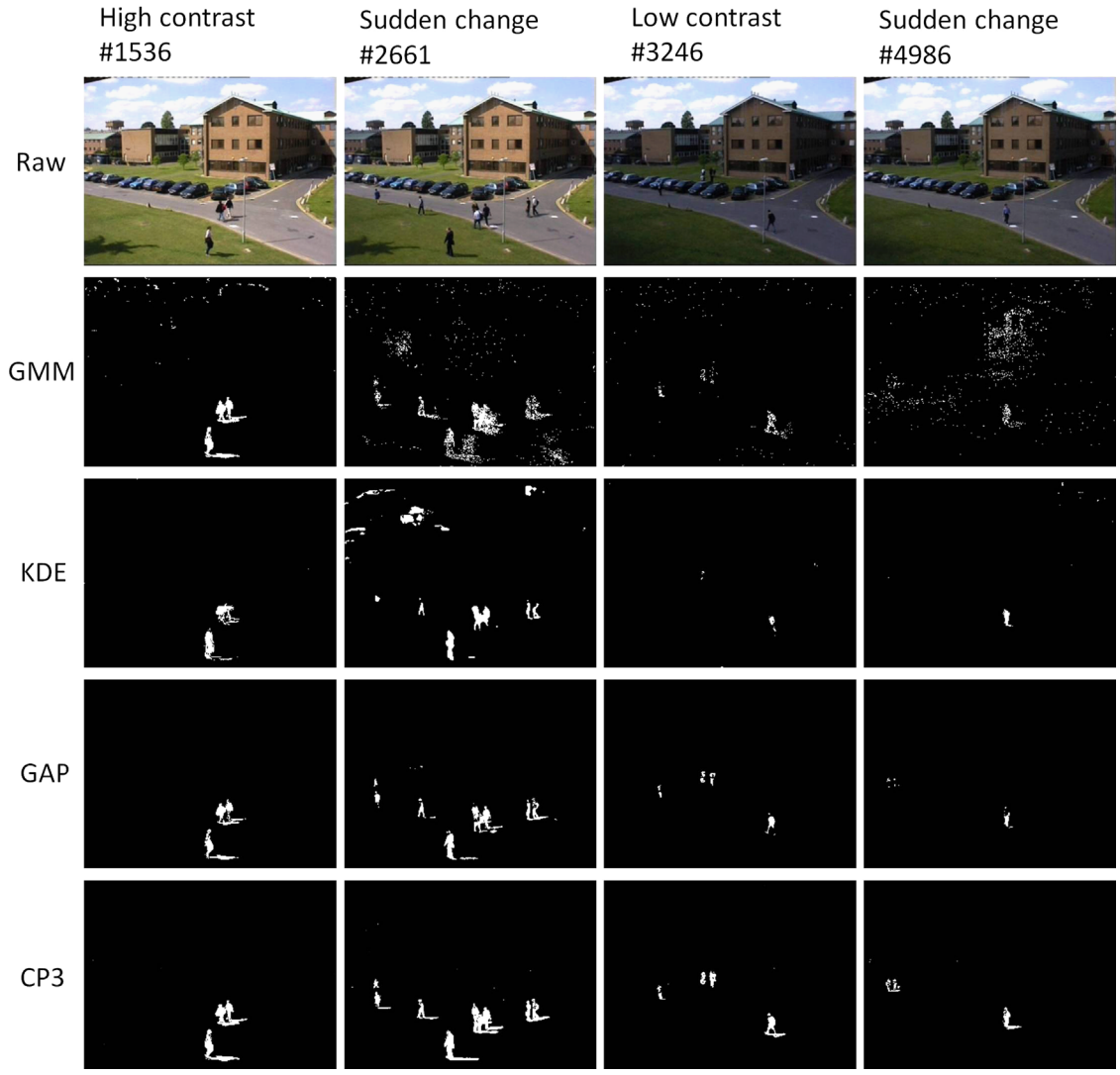

Fig. 10. CP3, GAP, KDE and GMM using PETS2001-dataset3 camera1.

4. Object detection

The proposed background model converts the object detection problem into a competitive binary classification problem [29] by comparing the intensity pairs $(P, \{Q_k^P\}_{k=1,2,\ldots,K})$ in turn. It includes two stages: (1) to identify the normal/abnormal state of the pixel pair (P, Q_k^P); (2) to identify the foreground/background state of P.

For each pixel pair (P, Q_k^P), the binary function $\beta(Q_k^P)$ for discriminating the normal/abnormal state can be estimated as the following condition according to Eq. (15):

$$\beta(\{Q_k^P\}_{k=1,2,\ldots,K}) = \begin{cases} 1 & \text{if } \|(p-q_k)-\hat{b}\| < C\cdot\hat{\sigma}_\varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

where p and q_k are the intensity values of P and Q_k^P in the input frame J respectively, and C is a constant. It is important to note that using the bivariate normal distribution of the differential increment of the pixel pair is different from using the traditional single Gaussian *pdf*-based identification function of a single pixel [6]; in a single Gaussian *pdf*-based method, the ideal threshold should be changed following the latest intensity variation. For example, the standard deviation should be larger when the illumination fluctuations become more intense. In our proposed version, the stable differential increment of a pixel pair provides an adaptive observation so that $\hat{\sigma}_\varepsilon$ is only related to the noise acting on each pixel. Therefore, we do not need an adjustable C to adapt to changes caused by illumination changes or background motion. The constant C can be set from 1.0 to 3.0 in order to contain approximately an area of 68–99% of its probability density function. In addition, the recording of $\hat{\sigma}_\varepsilon$ and $\hat{b}$ from the modeling stage provides a more accurate parametrized criterion than a fixed double-sided threshold in GAP method [26]. This will be confirmed in the experiment section.

After identifying the normal/abnormal state of the pixel pair, K bits of $\beta(Q_k^P)$ are produced for the following decisions of each P. In order to classify whether P is a foreground pixel, the probability $\xi(P)$ of a pixel being in the background is defined as,

$$\xi(P) = \frac{1}{K}\sum_{k=1}^{K}\beta(Q_k^P). \tag{21}$$

Target pixel P in the input image is considered as a foreground pixel only if $\xi(P) < PF$, where PF is a global threshold that can be adjusted to achieve the desired result. Otherwise, P is considered a background pixel. For instance, if $PF=0.5$, and the number of abnormal Q_k^P is larger than $K/2$, namely, $\xi(P) < 0.5$, then P should be a foreground pixel in the input frame J. The procedure for calculating $\xi(P)$ for every target pixel is performed by a bits counting operation, along with a look-up table for calculating $\beta(Q_k^P)$, which is easy to implement on

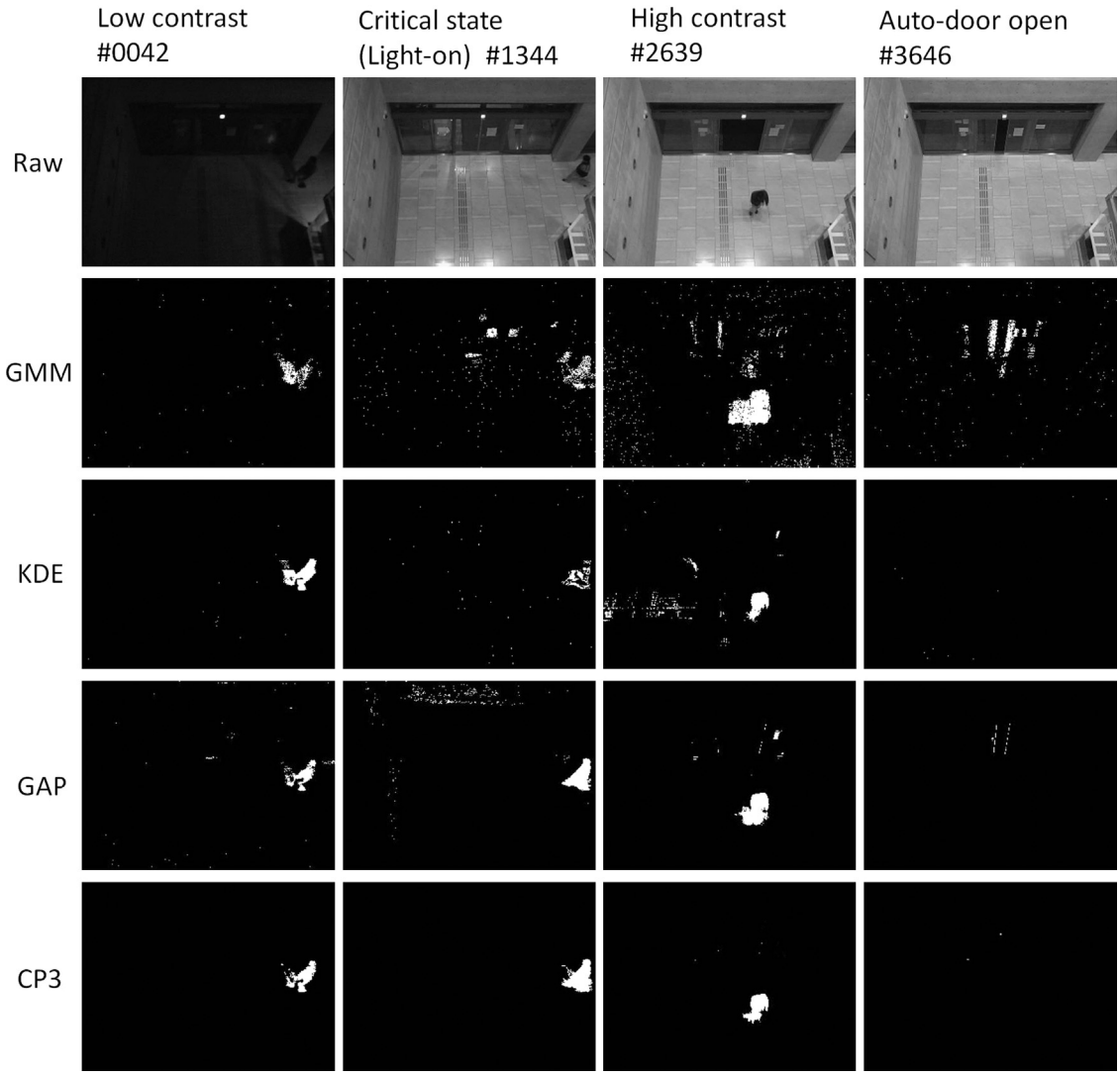

Fig. 11. CP3, GAP, KDE and GMM using AIST-INDOOR dataset.

any conventional hardware. The pseudo-code of the object detection is shown in Algorithm 2.

Algorithm 2. Object detection.

Data: Testing frame J, parameters C and PF.
Result: *Foreground*/*Background*
for Each pixel P in J **do**
(I) Initialize :
Load look−up table $\sim[u', v', \hat{\sigma}_\varepsilon, \hat{b}]$ of $\{Q_k^P\}_{k=1,2,\ldots,K}$.
(II) Pixel pair identification :
for $k = 1, 2, \ldots, K$ **do**
if $\|(p-q_k)-\hat{b}\| < C\cdot\hat{\sigma}_\varepsilon$ **then**
$\beta(Q_k^P) = 1$
else
$\beta(Q_k^P) = 0$
(III) Object identification :
Compute $\xi(P) = \frac{1}{K}\sum_{k=1}^{K}\beta(Q_k^P)$.
if $\xi(P) < PF, (0 < PF < 1)$ **then**
$P \rightarrow$ *Foreground*
else
$P \rightarrow$ *Background*

5. Experimental results

To evaluate the performance of the proposed method, we tested it on video datasets including a variety of indoor and outdoor environments for both qualitative and quantitative analysis. For all the experiments, $\sigma_n^2 = 100$ in the training stage and the two thresholds were set as $C=2.5$ and $PF=0.5$ respectively in the detection stage.

For quantitative analysis, the three evaluation metrics, *Precision* (also known as positive predictive value), *Recall* (also known as sensitivity) and *F*-measure were utilized. *Precision*, *Recall* and *F*-measure are widely used in pattern recognition and information extraction with binary classification [30,31]. Since pixel-level object detection is a typical binary classification problem, the three metrics also have been used for the quantitative analysis of object detection [10,26,32]. *Precision* can be seen as a measure of exactness or fidelity, and *Recall* can be seen as a measure of completeness of foreground,

$$Precision = \frac{TP}{TP+FP} \tag{22}$$

and

$$Recall = \frac{TP}{TP+FN} \tag{23}$$

where *TP*, *FP* and *FN* stand for the number of true positive pixels, false positive pixels and false negative pixels, respectively. *F*-measure is a weighted harmonic mean of the *Precision* and *Recall*,

$$F = \frac{2Precision\cdot Recall}{Precision+Recall}. \tag{24}$$

5.1. Experimental comparison with other approaches

We compared our algorithm with three methods: (1) GMM [7,13] method, which is a standardized method among independent pixel-wise models; (2) Sheikh's KDE method [10] as a representative method among spatially dependent models, which is different from the original nonparametric Kernel Density Estimation method (KDE) in that it employs KDE over the joint domain(location) and range (intensity) representation of image pixels; (3) our previous method GAP [26]. The parameters for the GMM algorithm were set as defaults in the OpenCV tool; the parameters for Sheikh's KDE algorithm were set according to the author's recommendations in [10], and the size of model is [26,21,31]; In GAP method, $W_G = 20, W_P = 0.9, W_H = 0.3$.

First, we use the sequences from PETS2001-dataset3 camera1 to test the outdoor scenes captured under severe illumination fluctuation. The sudden partial illumination variations caused by moving clouds are obvious in this scene which are clearly

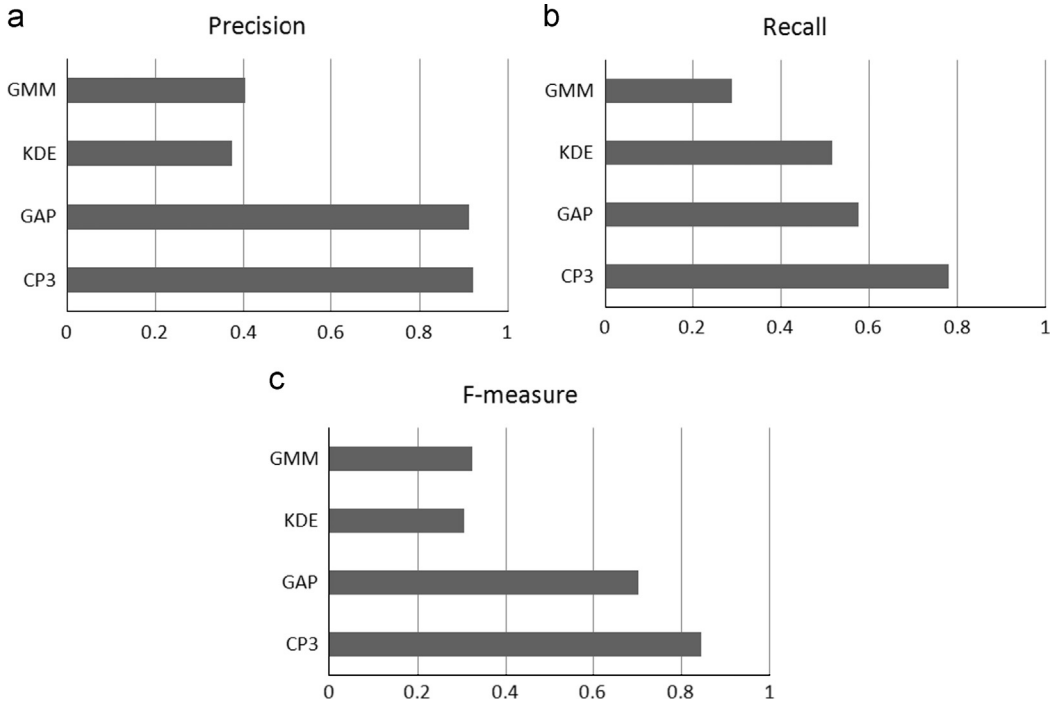

Fig. 12. The mean metrics of GMM, KDE, GAP and proposed CP3 using AIST-INDOOR dataset. (a) *Precision*, (b) *Recall* and (c) *F*-measure.

represented as average intensity change shown in Fig. 9(d). The dataset PETS2001-dataset3 camera1 includes a total of 5563 frames for training and 5336 frames for detection. The available ground truth data of PETS2001 allows us to complete the comparison (http://limu.ait.kyushu-u.ac.jp/en/dataset/). The average *Precision*, *Recall* and *F*-measure of the four methods are shown in Fig. 8. The *Precision*, *Recall* and *F*-measure over time of the four methods are shown in Fig. 9(a–c). Clearly, both CP3 and GAP show a higher level of *Precision* than the other two methods; CP3 has an obviously higher *Recall* and *F*-measure value than other methods which means it has higher sensitivity for detecting foreground. In addition, during the frames from 150 to 200, GAP and Sheikh's KDE methods show clearly decreasing performance of *Recall*, as the test video comes into a darker phase after the frame 150, and the dynamic range of the intensity is compressed, as shown in Fig. 9(d). Since the GAP method uses a fixed double-sided threshold, it is more sensitive to changes in dynamic range than CP3, which leads to more false negative detections (incomplete foreground). The performance of Sheikh's KDE and GMM methods show the weakness that they are sensitive to the rapid changes of illumination while updating. In addition, we have to point out another weakness of Sheikh's KDE method: its foreground modeling processing highly depends on the accuracy of foreground detection of the latest frames. Once the detection fails, the false foreground model will lead to unexpected results including large areas of false positive pixels (noise) or false negative pixels (incomplete foreground). Fig. 10 shows the qualitative results of the four methods under high contrast, low contrast and rapid changes of illumination, respectively. It is stressed that no morphological operators like erosion/dilation were used in the presentation of these results. The above results indicate that, our proposed method has a better illumination invariance compared with the state of the art methods, even under sudden illumination changes and a low contrast background.

The second dataset for testing indoor environments is the AIST-INDOOR dataset (http://ssc-lab.com/~liang/CP3_project/AIST_INDOOR_DATASET.rar). It contains several indoor extreme conditions: low contrast illumination, low texture, turning on lights and an auto-door rapidly opening and shutting. The dataset AIST-INDOOR includes total 300 frames for training and 4890 frames for detection. The average *Precision*, *Recall* and *F*-measure of the four methods are shown in Fig. 12. Fig. 11 shows four demo frames of detection with CP3, GAP, Sheikh's KDE and GMM respectively. Compared with other approaches, CP3 is not only insensitive to varying illumination but also robust to reciprocating motion of the auto-door.

Here, we would like to emphasize the differences between our method and Sheikh's KDE method from Ref. [10] because both of them belong to the spatial-dependence model. The most important difference is the difference in their modeling mechanisms. Sheikh's KDE method converts each pixel's intensity and location into a feature vector, and then models a joint probability distribution of all the pixels on a feature space. The method is based on the assumption that the pixels' correlation degree depends on the

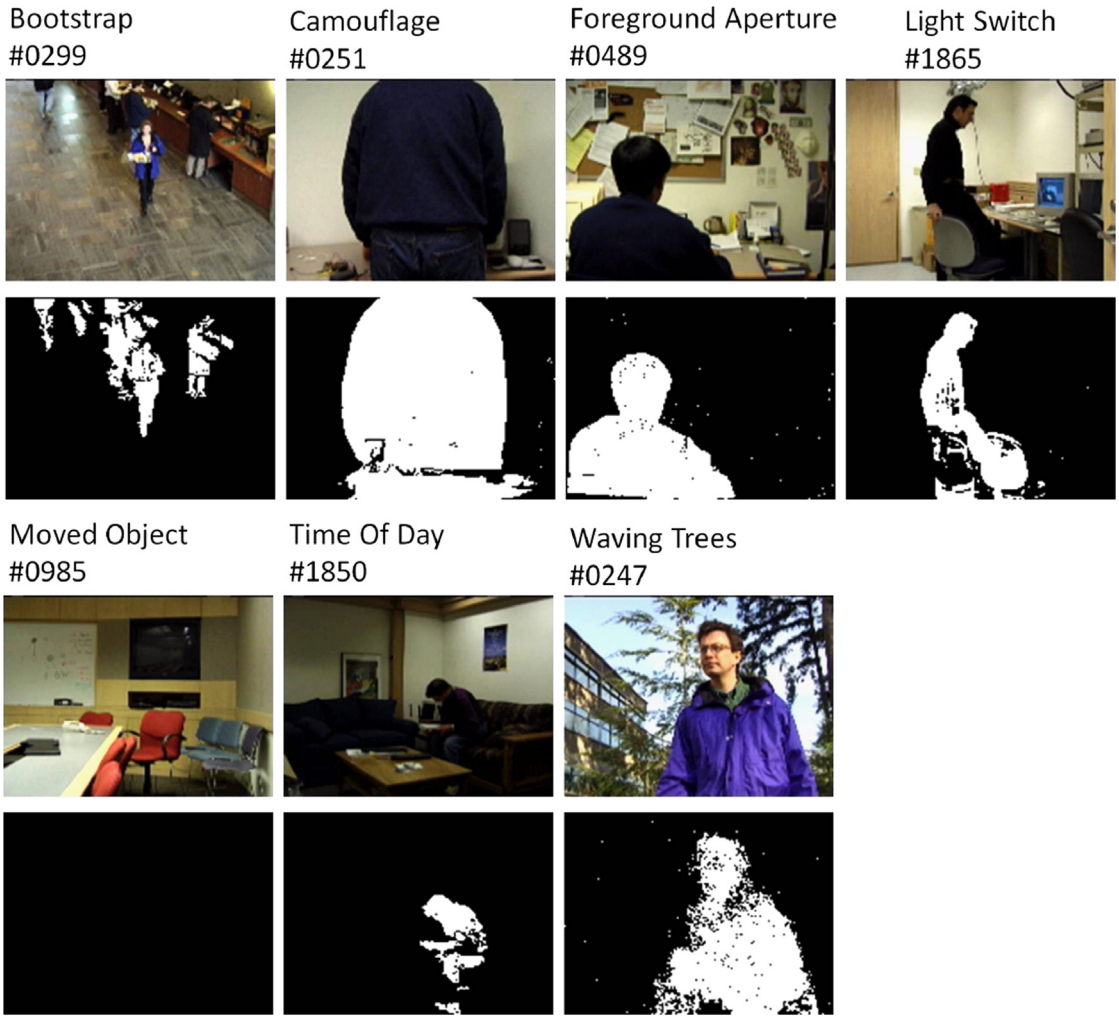

Fig. 13. CP3 results of Wallflower dataset.

spatial distance between them, (i.e. each observed pixel has higher correlation with its neighbouring pixels), but ignores the localized discrimination between pixels. Such a mechanism can deal well with global illumination changes, but is not robust to local illumination change, which can be clearly observed in the qualitative experimental results in Figs. 10 and 11. Compared with Sheikh's method, our proposed background model avoids any prior assumption of pixel's local correlation, but selects a group of supporting pixels which can maintain a stable statistical relationship with their target pixel. Such a mechanism is robust under both global and local illumination changes.

The third dataset is Wallflower, which is introduced in the work of Toyama et al. [19]. This dataset consists of seven video sequences, each of which addresses a special canonical background subtraction problem. We trained each video sequence using the frames specified by the instruction files in the Wallflower dataset. Our results are shown in Fig. 13. Compared with the results in [19], our method deals with the illumination changes and background fluctuations well. Note that some incomplete parts of the foreground exist mainly because the dark colors of both foreground and background come into similar intensities after covering them into gray-scale image. When using a multi-channel colour detection, the performance would be improved.

5.2. Work with different number of supporting pixels

To examine the detection performance of the proposed method using a different number of supporting pixels, we use PETS2001-dataset3 camera1 and calculate the mean value of *Precision*, *Recall* and *F*-measure shown in Fig. 14(a). The corresponding average runtime to process each frame is shown in Fig. 14(b). The runtime is measured on a computer with a Intel Xeon 3.0 GHz processor with a C language implement. From Fig. 14(a), we can observe that K mainly affects *Precision*: when $K=1$, *Precision* is around 0.5. According to the definition of *Precision*, this result means the number of false positive pixels (*FP*) and true positive pixels (*TP*) are similar which indicates that one supporting pixel Q_1^P cannot provide robust detection. When K becomes larger, *Precision* increases dramatically; and when K continues to grow (larger than 9), *Precision* tends to be stable, which means there is a small quantity of false positive pixels (*FP*). Fig. 14(a) also shows that *Recall* changes little under different K, which indicates that the completeness of the object could be preserved even K is small. The results of the above quantitative analysis can also be observed from the detection sample in Fig. 14(d–g). In addition, as shown in Fig. 14(b), the runtime of detection linearly increase with K. In conclusion, considering both the detection performance and

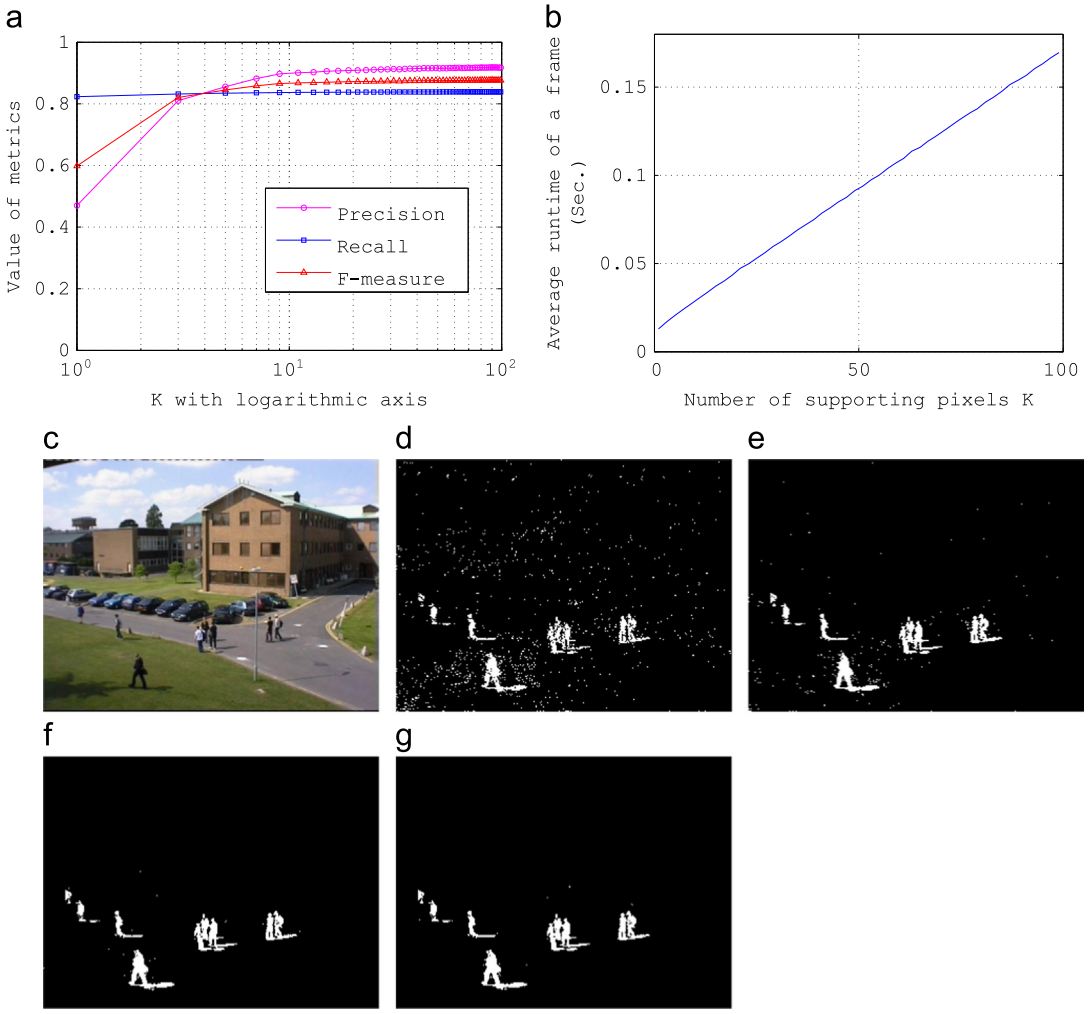

Fig. 14. Work with a different number of supporting pixels K. (a) *Precision*, *Recall* and *F*-measure under different K (to show the variation tendency clearly when K is small, the horizontal axis is logarithmic.). (b) Average detection runtime of a frame under different K. (c) Frame #2706. (d–f) Detection results with $K=1$, $K=5$, $K=15$, and $K=99$, respectively.

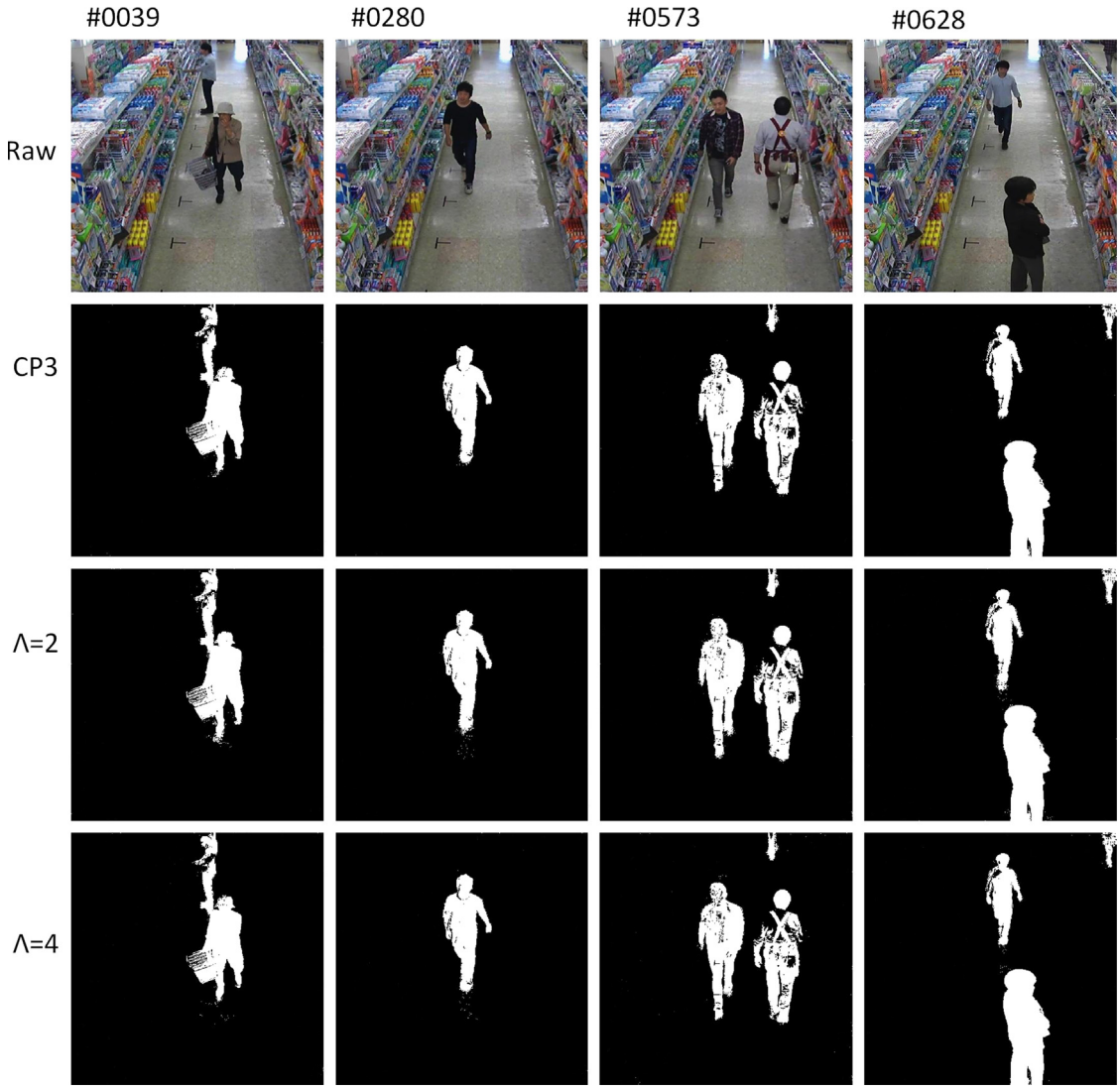

Fig. 15. Detection using CP3 and its accelerated algorithm.

Table 1
The comparison of CP3 and its accelerated version.

Interval	Runtime (Sec.)	Precision	Recall	*F*-measure
–	151.57	0.923	0.877	0.899
$\Lambda=2$	47.02	0.920	0.857	0.887
$\Lambda=3$	31.51	0.903	0.851	0.876
$\Lambda=4$	15.73	0.887	0.844	0.865
$\Lambda=5$	12.19	0.869	0.837	0.853

computation cost discussed above, we set the number of supporting pixels K between 10 and 20 in practice.

5.3. Analysis of accelerated background modeling

We also use the accelerated version to carry out background modeling with a sample interval from $\Lambda=2$ to $\Lambda=5$, and compare its time consumption of background modeling and object detection performance with Algorithm 1 shown in Table 1. The dataset is a high resolution (1024×1024) surveillance video in a supermarket with 101 training samples, some detection samples are shown in Fig. 15. The runtime are measured on a computer with a Intel Xeon 3.0 GHz processor. From Table 1, we can observe that compared with the original CP3, the time cost for background modeling of the accelerated algorithm dramatically reduces even when only using a limited sampling interval. For instance, when $\Lambda=5$, the runtime reduced by 91.96%, while the *Precision*, *Recall* and *F*-measure only reduced by 5.85%, 4.56%, and 5.12%, respectively. The above three metrics can keep high levels mainly because when the sampling interval is relative small, there are still sufficient qualified supporting pixels to maintain

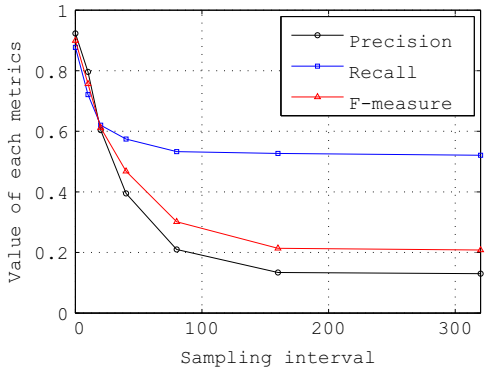

Fig. 16. *Precision*, *Recall* and *F*-measure using the accelerated algorithm of CP3 model with the sampling intervals from 0 to 320.

robust detection. Larger-range quantitative tests are shown in Fig. 16. We can observe that when the sampling interval Λ continues to increase, *Precision*, *Recall* and *F*-measure clearly decrease, and finally tend to be stable at low levels. In summary, when using the accelerated algorithm of CP3, it is suggested to firstly conduct a preliminary work using a relative small sampling interval, which would efficiently reduce the time cost of background modeling and maintain a fairly good detection performance. In addition, an optimized implementation of Algorithm 2 for object detection can process about 17 fps using a frame size of 1024×1024.

6. Conclusions

In conclusion, CP3 builds a novel background model for object detection based on co-occurrence pixel pairs. The model performs robust detection under outdoor and indoor extreme environments.

The key contributions of the algorithm are as follows:

- Compared with independent pixel-wise methods, CP3 determines stable co-occurrence pixel pairs, instead of building the parameterized/non-parametrized model for a single pixel. These pixel pairs maintain a reliable background model, which can be used to capture structural background motion and cope with local and global illumination changes.
- As a spatial-dependence method, CP3 does not predefine any local operator, subspace or block, and it provides an accurate detection criterion even though the gray-scale dynamic range is compressed under weak illumination.

We interpreted our method and conducted experiments using gray-scale data, however, using color imagery is also an option for object detection. The modification of CP3 for multi-channel use, such as *rgb*, can be realized through vectorized extension of the basic unit for a three-dimensional observation; for the Gaussian model of each pixel pair, the associated mean value and variance evolve to a 3D vector and a 3×3 covariance matrix respectively. In future work, we will integrate CP3 into an object tracking framework.

Conflict of interest

No conflict of interest.

Acknowledgements

The authors would like to thank Professor Takayuki Tanaka from Hokkaido University and Dr. Yutaka Satoh from Japan National Institute of Advanced Industrial Science and Technology (AIST) for their helpful comments, and Mr. Yoneda and Ms. Watanabe from COOP-Sapporo for providing the surveillance video, and Mr. William Xue from Massachusetts Institute of Technology for the language checking.

Appendix

K-means clustering for optimizing spatial distribution: Define a spatial distance between P and Q_n by $d(P, Q_n)$, and define a spatial distance between different Q_n by $d(Q_n, Q_{n'})$, where $n = 1, 2, \ldots, N$ and $n \neq n'$. There are different approaches for checking the distance between two observed values, such as Euclidean distance, Hamming distance or city-block distance. As a spatial distance measurement, we used Euclidean distance. A set to represent the local spatial distribution of P and Q_n is $\psi(Q_n) = \{d(P, Q_n), d(Q_n, Q_{n'})\}$.

(I) Initialize: Given random initial clustering centers $Q_k^P(0) \in \{Q_n\}$, $k = 1, 2, \ldots, K$, and one fixed clustering center of P, the total $K+1$ number of initial seed locations, the corresponding initial cluster $\{\psi(Q_k^P)_0\} = \Phi$.

(II) *Clustering*: Assign each $Q_n(u', v')$ to a cluster with the closest seed location:

$$\{Q_k^P\}_\eta = \{Q_n : d(Q_n, Q_k^P[\eta]) \leq d(Q_n, Q_k^P[\eta])\},$$

where η is iteration times, and $\forall k \neq k'$, until Q_n goes into exactly one $\{Q_k^P\}_\eta$.

(III) *Updating centroid*: The updated centroid of the cluster is calculated by the mean locations:

$$Q_k^P[\eta+1] = \#^{-1} \sum_{(u',v')} \{Q_k^P\}_\eta,$$

where # is the number of $Q_n \in \{Q_k^P\}_\eta$. It is important to note that the location of P without updating in step (III) also allows $Q_n \in \{P\}$ to calculate $d(P, Q_n)$ in step (II).

References

[1] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. 38 (4) (2006) 1–43.
[2] D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking, IEEE Trans. Pattern Anal. Mach. Intell. 25 (5) (2003) 564–575.
[3] S. Ali, A. Basharat, M. Shah, Chaotic invariants for human action recognition, in: IEEE 11th International Conference on Computer Vision, IEEE, Rio de Janeiro, 2007, pp. 1–8.
[4] I. Haritaoglu, D. Harwood, L.S. Davis, W4: real-time surveillance of people and their activities, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 809–830.
[5] A. Datta, M. Shah, N. Da Vitoria Lobo, Person-on-person violence detection in video data, in: 16th International Conference on Pattern Recognition, vol. 1, IEEE, Quebec, Canada, 2002, pp. 433–438.
[6] C.R. Wren, A. Azarbayejani, T. Darrell, A.P. Pentland, Pfinder: real-time tracking of the human body, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 780–785.
[7] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE, Fort Collins, 1999.
[8] A. Elgammal, R. Duraiswami, D. Harwood, L.S. Davis, Background and foreground modeling using nonparametric kernel density estimation for visual surveillance, Proc. IEEE 90 (7) (2002) 1151–1163.
[9] K. Kim, T.H. Chalidabhongse, D. Harwood, L. Davis, Real-time foreground–background segmentation using codebook model, Real-Time Imaging 11 (3) (2005) 172–185.
[10] Y. Sheikh, M. Shah, Bayesian modeling of dynamic scenes for object detection, IEEE Trans. Pattern Anal. Mach. Intell. 27 (11) (2005) 1778–1792.
[11] R.M. Haralick, K. Shanmugam, I.H. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cybern. (6) (1973) 610–621.
[12] M. Hashimoto, M. Saito, High-speed and robust image matching using spatially distinctive and temporally stable pixels, in: 2011 International Symposium on Optomechatronic Technologies (ISOT), IEEE, Hong Kong, China, 2011, pp. 1–8.
[13] C. Stauffer, W.E.L. Grimson, Learning patterns of activity using real-time tracking, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 747–757.
[14] J. Rittscher, J. Kato, S. Joga, A. Blake, A probabilistic background model for tracking, Computer Vis.—ECCV 2000 (2000) 336–350.
[15] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, J. M. Buhmann, Topology free hidden Markov models: application to background modeling, in: Eighth IEEE International Conference on Computer Vision, vol. 1, IEEE, Vancouver, British Columbia, Canada, 2001, pp. 294–301.
[16] J.-M. Guo, Y.-F. Liu, C.-H. Hsia, M.-H. Shih, C.-S. Hsu, Hierarchical method for foreground detection using codebook model, IEEE Trans. Circuits Syst. Vid. Technol. 21 (6) (2011) 804–815.
[17] T. Matsuyama, T. Ohya, H. Habe, Background subtraction for non-stationary scenes, in: Proceedings of Asian Conference on Computer Vision, 2000, pp. 662–667.
[18] M. Seki, T. Wada, H. Fujiwara, K. Sumi, Background subtraction based on cooccurrence of image variations, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE, Madison, Wisconsin, 2003, pp. II–65.
[19] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, Wallflower: principles and practice of background maintenance, in: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 1, IEEE, Kerkyra, Greece, 1999, pp. 255–261.
[20] N.M. Oliver, B. Rosario, A.P. Pentland, A Bayesian computer vision system for modeling human interactions, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 831–843.

[21] A. Monnet, A. Mittal, N. Paragios, V. Ramesh, Background modeling and subtraction of dynamic scenes, in: Ninth IEEE International Conference on Computer Vision, IEEE, Beijing, China, 2003, pp. 1305–1312.
[22] J. Zhong, S. Sclaroff, Segmenting foreground objects from a dynamic textured background via a robust kalman filter, in: Ninth IEEE International Conference on Computer Vision, IEEE, Beijing, China, 2003, pp. 44–50.
[23] M. Heikkilä, M. Pietikäinen, A texture-based method for modeling the background and detecting moving objects, IEEE Trans. Pattern Anal. Mach. Intell. 28 (4) (2006) 657–662.
[24] K. Iwata, Y. Sato, R. Ozaki, K. Sakaue, Robust background subtraction based on statistical reach feature method, IEICE Trans. Inf. Syst. 92 (2009) 1251–1259.
[25] W. Hu, X. Li, X. Zhang, X. Shi, S. Maybank, Z. Zhang, Incremental tensor subspace learning and its applications to foreground segmentation and tracking, Int. J. Comput. Vis. 91 (3) (2011) 303–327.
[26] X. Zhao, Y. Satoh, H. Takauji, S. Kaneko, K. Iwata, R. Ozaki, Object detection based on a robust and accurate statistical multi-point-pair model, Pattern Recognit. 44 (6) (2011) 1296–1311.
[27] X. Zhao, Y. Satoh, H. Takauji, S. Kaneko, K. Iwata, R. Ozaki, Robust adapted object detection under complex environment, in: 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance, 2011, pp. 261–266.
[28] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, California, USA, 1967.
[29] S.Y. Elhabian, K.M. El-Sayed, S.H. Ahmed, Moving object detection in spatial domain using background removal techniques-state-of-art, Recent Patents Comput. Sci. 1 (1) (2008) 32–54.
[30] T. Fawcett, An introduction to roc analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874.
[31] J. Makhoul, F. Kubala, R. Schwartz, R. Weischedel, Performance measures for information extraction, in: In Proceedings of the DARPA Broadcast News Workshop, 1999, pp. 249–252.
[32] A. Shimada, H. Nagahara, R.-i. Taniguchi, Background modeling based on bidirectional analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Portland, Oregon, 2013, pp. 1979–1986.

Dong Liang received the B.S. degree in Telecommunication Engineering and the M.S. degree in Circuits and Systems from Lanzhou University, China, in 2008 and 2011, respectively. He is studying for the Ph.D. course at System Sensing and Control Lab, Graduate School of Information Science and Technolgy, Hokkaido University, Japan. His research interests include computer vision and pattern recognition.

Shun'ichi Kaneko received the B.S. degree in Precision Engineering and the M.S. degree in Information Engineering from Hokkaido University, Japan, in 1978 and 1980, respectively, and then the Ph.D. degree in System Engineering from the University of Tokyo, Japan, in 1990. He had been a research assistant of the Department of Computer Science since 1980 to 1991, an associate Professor of the Department of Electronic engineering since 1991 to 1995, and an associate Professor of the Department of Bio-application and Systems Engineering since 1996 to 1996, in Tokyo University of Agriculture and Technology, Japan. He is currently a Professor at the Graduate School of Information Science and Technology, Hokkaido University, Japan. His research interests include machine and robot vision, image sensing and understanding, and robust image registration.

Manabu Hashimoto received the B.E. and M.E. degrees in Welding Engineering from Osaka University in 1985 and 1987, respectively. He joined Manufacturing Development Laboratory, Mitsubishi Electric Corporation in 1987, and engaged in research of vision system for factory automation. Since 2002 he has been a research manager of Advanced Technology R&D Center. His research interests include pattern recognition and 3-D object recognition for industrial robots, and face recognition for human-machine interfacing. He received the Ph.D. degree from Osaka University in 2000 and he is a member of the IEEE, IEICE Japan, IEE Japan, and Robotics Society of Japan.

Kenji Iwata received his Ph.D. degree in Engineering from Gifu University, Japan, in 2002. He is a research scientist of National Institute of Advanced Industrial Science and Technology (AIST), Japan, from 2005. His research interest includes computer vision and middleware for its systems.

Xinyue Zhao received the M.S. degree in Mechanical Engineering from Zhejiang University, China in 2008, and the Ph.D. degree in Graduate School of Information Science and Technology from Hokkaido University, Japan in 2012. She is currently an assistant professor at Department of Mechanical Engineering, Zhejiang University, China. Her research interests include computer vision and image processing.