Grid-based Template Matching for People Counting

Jun-Wei Hsieh[1,*], Chi-Hung Chuang[2], Sin-Yu Chen[3],
Cheng-Shuang Peng[4], and Kuo-Chin Fan[4]

[1] Department of Computer Science and Engineering
National Taiwan Ocean University
No.2, Beining Rd., Keelung 202, Taiwan
shieh@mail.ntou.edu.tw

[2] Department of Learning and Digital Technology
Fo Guang University
No.160, Linwei Rd., Jiaosi , Yilan 26247, Taiwan

[3]Department of Electrical Engineering
Yuan Ze University
135 Yuan-Tung Road, Chung-Li 320, Taiwan

[4]Department of Computer Engineering
National Central University
Jung-Da Rd., Chung-Li 320, Taiwan

Received 1 May 2010; Revised 28 May 2010; Accepted 30 June 2010

Abstract. This paper presents a novel template matching method to detect and track pedestrians for counting people in real-time. Template matching is a time-consuming technique and performs weakly in matching targets if their appearances change larger. The result of unstable matching will increase lots of false detection and missing rates in people counting. To improve the effectiveness of this technique, a novel grid structure is then proposed for tackling the problem of pedestrian appearance changes. Since the technique is time-consuming, a novel ring structure with integral image is furthermore proposed for quickly filtering out impossible candidates and thus each pedestrian can be counted in real time. Different from training approaches which should train several classifiers and thus need several scanning processes to detect different pedestrians, this approach uses only one scanning process to detect each desired pedestrian from videos. In this system, a GMM (Gaussian Mixture model)-based subtraction technique is first used to detect different moving objects from videos. Then, a shadow elimination method is used for reducing shadow effects into a minimum. After that, the novel grid-based verification approach is then proposed for verifying and counting each moving pedestrian more robustly and accurately. To speed up the verification efficiency, a novel ring structure with integral images is then proposed to count people in real time. Finally, a tracking method is applied to tracking each moving pedestrian so that the real number of passing people per direction can be more accurately counted. Experimental results prove that the proposed method is a robust, accurate, and powerful tool in people counting.

Keywords: People counting, template matching, object tracking, background modeling

1 Introduction

Counting people in a noisy environment is an important task in video surveillance. It can be used in various public places like shopping malls, public transport stations, theaters, department stores, or trade fairs for security issues, service optimization, and resource allocation. However, due to the high variation of people appearance, it is very challenging to automatically detect and count people directly from videos. To tackle the problem, in the past decades, there have been many approaches proposed for people counting. According to the used inputs, these approaches can be further divided into two categories: intensity-based or stereo-based. The intensity-based approach counts persons using only one single camera which may be mounted overhead or in front of people. For example, in [1], Harasse et al. used a side-mounted camera to detect face and then to count people using a trajectory tracking technique. In addition to face, body contour is another important feature to locate pedestrians from videos. In [2], Benozzi et al. used a vertical projection scheme to obtain the vertical symmetry

[*] Correspondence author

of pedestrians so that a set of pedestrian candidates were obtained. Furthermore, Haritaoglu et al. [3] proposed a silhouette-based approach to detect and count pedestrians. Pai et al. [4] used an entropy feature to model pedestrians and then to count all walking pedestrians from videos. In [5], Rabaud and Belongie used a feature tracker to track pedestrians and then counted them for action event analysis. Similarly, in [6], Antonini and Thiran assumed that all the pedestrian trajectories had been extracted and then proposed a trajectory clustering technique to count pedestrians. In [7], Masoud and Papanikolopoulos used different rectangular patches to model pedestrians and then tracked them for traffic control. The above systems adopt a sided-mounted camera to count people and will fail when objects have occlusions. To tackle the occlusion problem, another choice of camera setting is using an overhead camera [8, 9, 10]. For example, Schofield et al. [10] used neural networks to learn pedestrian models and then detected walking pedestrians directly from an overhead camera. In [8], Snidaro et al. used a subtraction technique to detect pedestrian candidates and then verified them according to their areas. Furthermore, Adriano et al. [9] used neural networks to train a hair detector and then detected all possible pedestrians using their hair regions. However, when the color of clothes or shadow is similar to hair, the hair feature will become unstable for people counting.

To discriminate hair regions from the background, it is better to use stereo images for counting people since many false alarms can be effectively eliminated by stereo data. In [11], Zhao and Thorpe presented a counting system to detect possible pedestrian candidates from stereo images and then used neural networks to verify them. Darrell et al.[12] combined stereo, color, and face to count persons in crowded environments. In [13], Kelly et al. incorporated a biometric model to cluster stereo maps into individual pedestrian regions. Terada et al. [14] transformed the stereo images of passing people into a space-time image so that different pedestrians can be analyzed. In [15], Yang et al. proposed a real-time network consisting of multiple cameras to count people in crowds. Although 3-D features are more informative than 2-D image for people counting, the inherent correspondence problem and high computational cost make this approach inappropriate for real-time applications.

This paper proposes a novel grid-based template matching system to automatically count people directly from videos. As we know, template matching is a common technique in computer vision for building correspondences (or finding motion flows) between two images or frames. However, it is time-consuming and performs weakly in matching targets if their appearances change larger. The unstable matching result will increase lots of false detection or missing rates in counting people. Thus, this paper proposes two novel ideas to improve its effectiveness and efficiency in counting people. To improve its effectiveness, a grid structure is then proposed for tackling the problem of pedestrian appearance changes and thus reducing the perspective effects into a minimum. To improve its efficiency, a ring structure is then proposed for quickly filtering out impossible pedestrian candidates. This structure takes advantages of integral images to coarse-to-finely locate correct pedestrian positions in real time. Different from training approaches which should train several classifiers and thus need multiple scanning processes to detect different types of pedestrians, our method needs only one scanning process to detect various pedestrians from videos even though their types are different. In this system, we first use a GMM-based background subtraction technique to detect different moving objects. However, due to the problem of camera vibrations, many subtraction errors will be found along the boundaries of objects. Therefore, a novel subtraction method with a minimum filter is then proposed for eliminating these errors. After background subtraction, a shadow elimination method is then used for reducing shadow effects into a minimum. Once each moving region is extracted, the novel grid-based verification approach is then proposed for verifying and counting all possible pedestrians in it more robustly and accurately. To speed up the verification efficiency, a novel ring structure with integral images is then proposed for quickly filtering out impossible candidates and thus each pedestrian can be detected in real-time. After that, a tracking technique is used for tracking each pedestrian's direction so that people per direction can be more accurately counted. The average accuracy of our proposed method is 96.26%. The experimental results demonstrate both the grind-based template matching scheme and the ring-based verification method improve the performances of people counting very significantly in terms of its accuracy and efficiency.

The remainder of the paper is organized as follows. Section 2 introduces the overall flowchart of the proposed system. Then, details of the preprocessing schemes are described in Section 3. The grid-based method for people counting is proposed in Section 4. Finally, some conclusions will be presented in Section 5.

2 Flowchart of the Proposed System

You may prepare your camera-ready manuscript with MS-Word using this typeset together with the template joc.dot (see Sect. 3) or any other text processing system. In the latter case, please follow these instructions closely in order to make the volume look as uniform as possible.

We would like to stress that the class/style files and the template should not be manipulated and that the guidelines regarding font sizes and format should be adhered to. This is to ensure that the end product is as homogeneous as possible.

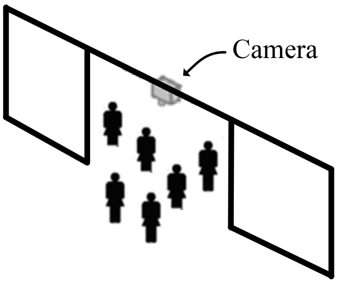


Fig. 1. An overhead camera is used for people counting

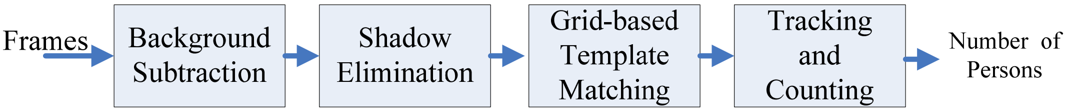


Fig. 2. Flowchart of the proposed system

This paper presents a novel system to count the number of persons walking through a door. Like Fig. 1, this system uses an overhead CCTV camera to capture videos and count people for avoid the occlusion problem. Although there were some systems [7][6] using a sided-mounted camera to observe an area, the occlusion problem makes them be unstable and inaccurate in counting people. The flowchart of the proposed system is shown in Fig. 2. Since the overhead camera is fixed on the ceiling, a novel background subtraction technique with a minimum filter is first proposed to detect different foreground objects from background. Then, a shadow elimination technique is used for removing unexpected shadows from each detected pedestrian. After that, a novel grid-based template matching technique is then proposed for verifying each pedestrian candidate. In this technique, a ring structure is also proposed for tackling the problem when pedestrians have appearance changes. With a tracking technique, the direction of each walking pedestrian can be furthermore estimated so that people per direction can be more accurately counted. In what follows, details of each component will be described.

3 Preprocessing

Before counting people, each moving object should be first extracted from the background. Thus, this paper proposes a new background subtraction technique with a minimum filter to extract desired foreground objects. For removing the effect of shadows, a shadow elimination method is also described. In what follows, details of these techniques are discussed.

3.1 Background Construction Using Gaussian Mixture Models

Use 10-point type for the name(s) of the author(s) and 9-point type for the address(es) and the abstract. For the main text, please use 10-point type and single-line spacing. We recommend using Computer Modern Roman (CM) fonts, Times, or one of the similar typefaces widely used in photo-typesetting. (In these typefaces the letters have serifs, i.e., short endstrokes at the head and the foot of letters.) Italic type may be used to emphasize words in running text. Bold type and underlining should be avoided. With these sizes, the interline distance should be set so that some 45 lines occur on a full-text page.

Given a surveillance video, extracting foreground objects from background is an important step in video-related surveillance applications. A standard method of adaptive background modeling is running average of successive images over time, in which a created approximated background is similar to the current static scene except where motion occurs. This method is not robust to scenes with many moving objects particularly when they move slowly. Therefore, rather than explicitly estimating the values of all the pixels, we model a pixel as a mixture of Gaussians [3] instead. Thus, a pixel that does not match the weighted sum of the background distributions is considered foreground.

In [3], the probability that an observed pixel has intensity value x_t at time t is estimated by K Gaussian distributions defined as follows:

$$P(x_t)=\sum_{l=1}^{K}\frac{\omega_{l,t}}{(2\pi)^{1/2}}e^{-\frac{1}{2}(x_t-\mu_l)^T\Sigma_l^{-1}(x_t-\mu_l)}, \quad (1)$$

where $\omega_{l,t}$ is the weight of the l^{th} distribution of pixel x_t 's mixture model, μ_l its mean, and Σ_l its covariance matrix. To update the model, each new pixel is checked if a match is found against the existing Gaussians. To adjust the weight of each distribution, the weight $\omega_{l,t}$ is updated by

$$\omega_{l,t}=(1-\alpha)\omega_{l,t-1}+\alpha(M_{l,t}), \quad (2)$$

where α is the learning rate which controls the speed of the learning, and M a Boolean value indicating whether a match is found or not. The definition of M is as follows: $M_{l,t}$=1 when a match is confirmed at the l^{th} distribution at time t; otherwise, $M_{l,t}$=0. The parameters μ and σ are updated by

$$\mu_{l,t}=(1-\beta)\mu_{l,t-1}+\beta x_t \text{ and } \sigma_{l,t}^2=(1-\beta)\sigma_{l,t-1}^2+\beta\left(x_t-\mu_{l,t}\right)^T\left(x_t-\mu_{l,t}\right), \quad (3)$$

where $\beta=\alpha P\left(x_t \mid \mu_{l,t-1},\sigma_{l,t-1}\right)$. For those pixels that are far away from the background distributions will be recognized as foreground.

3.2 Background Subtraction Using a Minimum Filter

Assume $I_k(p)$ and $B_k(p)$ are intensities of the kth frame and background of a pixel p, respectively. Then, the difference image $D_k(p)$ between $I_k(p)$ and $B_k(p)$ can be defined as follows:

$$D_k(p)=\begin{cases}0, \text{ if } |I_k(p)-B_k(p)|\le T_d; \\ I_k(p), \text{ otherwise,}\end{cases} \quad (4)$$

where T_d is a predefined threshold and chosen the average of all subtractions $|I_k(p)-B_k(p)|$. However, this subtraction technique is not stable if the camera has small vibrations. The perturbations will lead to many subtraction errors found along objects' boundaries. In what follows, this paper proposes a simple but effective method to remove all above subtraction errors so that different moving object can be more robustly detected. For any point p in B_k, due to the camera vibrations, its position in I_k will have some shift and lead to a large subtraction error. Assume that this shift is less than a window $w\times w$. Then, we can calculate $D_k(p)$ by subtracting $I_k(p)$ not only from the intensity $B_k(p)$ but also from the neighborhoods within p in B_k. Thus, Eq.(4) can be rewritten using the form

$$D_k(p)=\begin{cases}0, \text{ if } d_k(p)\le T_d; \\ I_k(p), \text{ otherwise,}\end{cases} \quad (5)$$

where $d_k(p)=\min_{q\in Ne(p)}|I_k(p)-B_k(q)|$ and $Ne(p)$ is the neighborhood of p defined within a window $w\times w$. In real implementation, w is set to 11 which is determined according to the assumption that the magnitude of camera vibration is less than 11 pixels. It is noticed that the subtraction errors found in Eq.(4) happen only when the subtraction $|I_k(p)-B_k(p)|$ is larger. If $|I_k(p)-B_k(p)|$ is small, $d_k(p)$ is not necessarily calculated. Based on this idea, Eq.(5) can be more efficiently performed using the following form:

$$D_k(p)=\begin{cases}0, \text{ if } |I_k(p)-B_k(p)|\le T_d; \\ I_k(p)\delta(d_k(p)-T_d), \text{ otherwise,}\end{cases} \quad (6)$$

where $\delta(x)$ is a step function whose value is 1 if x > 0 and 0 otherwise. If there are less than 10% pixels having larger intensity changes in I_k, Eq.(3) can save more than 90% unnecessary mask operations than Eq.(2).

3.3 Shadow Elimination

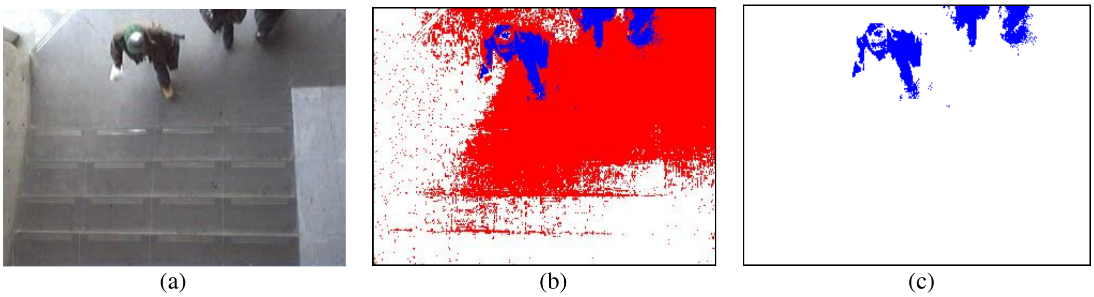


Fig. 3. Result of shadow elimination (a) Original Image (b) Background subtraction (c) Result of shadow elimination

After background subtraction, different moving objects can be well extracted from the background. However, the existence of shadow will affect the accuracy of people counting. Therefore, we adopt a deterministic approach proposed in [16] for reducing the effect of shadow into a minimum. Let $I_k^h(p)$, $I_k^s(p)$, and $I_k^v(p)$ denote the hue, saturation, and value channels of a pixel p in the kth input frame I from the HSV color space, respectively. Similarly, $B_k^h(p)$, $B_k^s(p)$, and $B_k^v(p)$ denote the hue, saturation, and value channels of p in the background. The decision rule for determining whether p is a shadow is defined as:

$$S(p)=\begin{cases}\text{True}, & \text{if } 1<B_k^v(p)/I_k^v(p)<R, \\ & \left|I_k^h(p)-B_k^h(p)\right|<\tau_H, \\ & \text{and } I_k^s(p)-B_k^s(p)<\tau_S, \\ \text{False}, & \textit{otherwise},\end{cases} \tag{7}$$

where both R and τ_H are set to 3, and τ_S < 0. Fig. 3 shows an example of shadow elimination, where (a) is the input frame, (b) the result of background subtraction, and (c) is the result of shadow elimination. Clearly, the shadow pixels were almost eliminated.

3.4 Connected Component Analysis

After shadow elimination, different moving objects can be well extracted. However, there are still many noisy regions which are not suitable for people counting. Therefore, a connected component analysis [17] is used for removing all smaller regions. If a region contains less than 10 pixels, it will be considered a noise and removed out. In addition to the connected component analysis, we also use some morphological operations to fill some holes found in the remained regions. This paper uses a closing operation with a 3×3 structure element to remove unwanted holes.

4 Ring Structure for Candidate Filtering

Once different moving regions have been extracted, this paper uses a template matching scheme to verify and count all possible pedestrians in them. This technique is very time-consuming and will fail to match objects if they have large appearance changes. For applications (like people counting) which consider "real-time" as an important issue, the technique will not be their good choice. However, it is still a good scheme to build correspondences between two images for applications in 3D vision. This paper makes two contributions for improving this technique's efficiency and effectiveness. For improving its "efficiency", this section will present a novel ring structure to quickly filter out impossible pedestrian candidates. As to the "effectiveness", Section 5 will propose a grid-based structure for tackling the problem when pedestrians have large appearance changes.

4.1 Template Matching Using Distance Transform

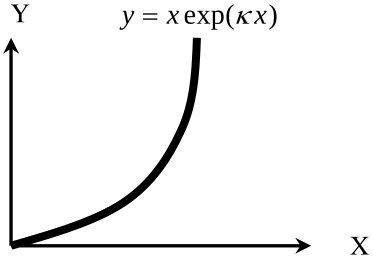


Fig. 4. The value of *y* is nonlinearly increased when *x* increases

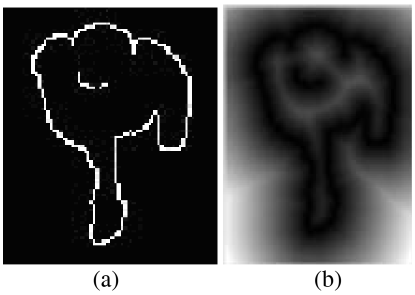


Fig. 5. Result of distance transform (a) Original Image (b) Distance transform of (a)

Due to lighting, noise, or other camera problems, a moving region (extracted by background subtraction) will not correctly correspond to a real pedestrian. To solve this problem, a full searching scheme is adopted here for scanning each pixel in this region as possible pedestrians. Then, this section will describe a distance transform for verifying these candidates. Since a full search is adopted, the scheme will become very time-consuming. Thus, in Section 4.2, a new ring structure with integral images will be proposed for improve its efficiency.

Assume that B_R is a set of edge pixels extracted from R. Then, the distance transform of a pixel p in R is defined as

$$DT_R(p) = \min_{q\in B_R} d(p,q)\,, \tag{8}$$

where $d(p,q)$ is the Euclidian distance between p and q. In order to enhance the strength of distance changes, Eq.(8) is further modified as follows

$$\overline{DT}_R(p) = \min_{q\in B_R} d(p,q)\times\exp(\kappa d(p,q))\,, \tag{9}$$

where $\kappa = 0.1$. Like Fig. 4, when x increases more, the value of y will increase more rapidly than x. Fig. 5(b) shows the result of the distance transform of Fig. 5(a). Thus, according to Eq.(9), a set F_R of contour features can be extracted from R. If we scan all pixels of R in a row major order, F_R can be then represented as a vector, i.e.,

$$F_R = [\overline{DT}_R(p_0),....,\overline{DT}_R(p_i),....]\,, \tag{10}$$

where all p_i belong to R and i is the scanning index. In addition to the outer contour, a pedestrian usually contains many inner edge points. To verify a pedestrian candidate more accurately, its outer shape should play a more important role than its inner shapes. Thus, for each pixel p_i, a weight w_i is included for weighting its importance, where w_i increases according to the distance between p_i and the central of R. Assume that r_i is the distance between p_i and the central of R, and the circumcircle of R has the radius z. Then, w_i is defined by the form:

$$w_i = \begin{cases} \exp(-|r_i - z|^2), \text{ if } r_i \le z, \\ 0, \qquad \text{elsewise.} \end{cases}$$

Then, Eq.(10) can be rewritten as follows:

$$\bar{F}_R = [w_0\overline{DT}_R(p_0),....,w_i\overline{DT}_R(p_i),....]\,. \tag{11}$$

Like Fig. 6, the original image R shown in (a) has two different inner and outer shapes. The yellow circle is the circumcircle of R. (b) is the weighting function used to enhance the outer contour of R. (c) is the weighted version of (a) when (b) is multiplied into (a).

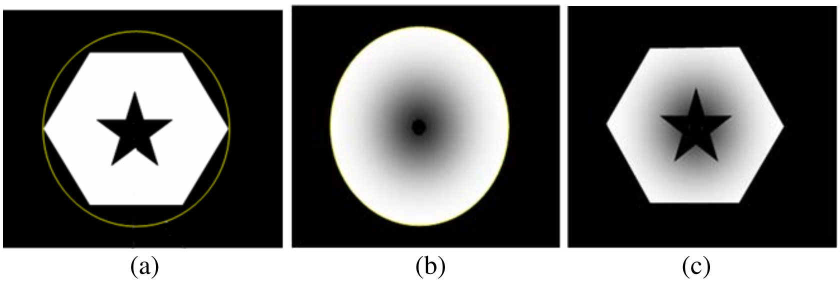


Fig. 6. Different weights used for enhancing the outer contours (a) Original Image (b) Weighting function (c) Weighted version of (a)

In practice, due to different environmental changes (like lighting), given a moving pedestrian C, it will have different visual appearance changes. To tackle these changes, more than one templates are collected for representing C. Assume that there are K templates in C. Then, given a candidate H, the distance between H and C can be measured by this equation:

$$S(H,C)=\min_{R\in C}[\frac{1}{|H|}\sum_{p\in R}\bar{F}_H(p)+\frac{1}{|R|}\sum_{r\in H}\bar{F}_R(r)], \qquad (12)$$

where $|H|$ and $|R|$ denotes the numbers of edges in H and R, respectively.

After describing the template matching technique, we also analyze its time complexity as follows. Assume that the dimensions of the used template and the analyzed image frame I are $m\times m$ and $n\times n$, respectively. If each pixel in I forms a pedestrian candidate, the time complexity for template matching is $O(m^2n^2)$. It will become $O(Km^2n^2)$ if K templates are collected for the verification process. Although the subtraction technique can reduce the number of verified pedestrian candidates, its time complexity is still $O(Km^2n^2)$. Thus, in what follows, a ring structure will be proposed for speeding up the verification process.

4.2 Ring Structure Using Integral Image for Fast Candidate Verification

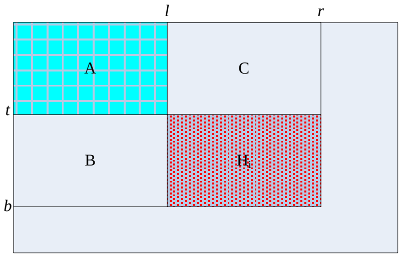


Fig. 7. Calculation of integral image

For speeding up the efficiency of verification, this section will propose a novel ring structure for avoiding lots of redundant verifications. Thus, different pedestrians can be more efficiently detected and counted. In practice, if a pedestrian candidate H_i satisfies Eq.(12), it will also include many edge pixels. In other word, before checking Eq.(12), we require H_i satisfying the following equation:

$$\text{edge}(H_i) > \theta_e, \qquad (13)$$

where $\text{edge}(H_i)$ denotes the number of edge points in H_i and θ_e a threshold to filter out impossible candidates if they have no enough edge points. When verifying, $\text{edge}(H_i)$ should be calculated by scanning all pixels in H_i. If the dimension of H_i is $m\times m$, the time complexity to calculate $\text{edge}(H_i)$ is $O(m^2)$. It can be reduced into $O(1)$ if the concept of integral image is used. After describing the integral image, a ring structure will be introduced for recursively filtering out impossible pedestrian candidates. Given an edge map B, its integral image Ig(x, y) contains the sum of edge points in B accumulated from the original (0, 0) to the pixel (x, y), i.e.,

$$Ig(x,y)=\sum_{i=0}^{x}\sum_{j=0}^{y}B(i,j),\text{ where } B(i,j)=\begin{cases}1, \text{ if } (i,j) \text{ is an edge point,}\\ 0, \quad\quad\quad\quad \text{otherwise.}\end{cases}$$

The integral image can be computed recursively, by the form

$$Ig(x,y)= Ig(x,\ y\text{-}1)+Ig(x\text{-}1,\ y)+Ig(x,\ y)\text{-}B(x\text{-}1,\ y\text{-}1)\,, \tag{14}$$

with the boundary condition: $B(\text{-}1,\ y) = B(x,\text{-}1) = B(\text{-}1,\text{-}1) = 0$. Clearly, the computation of $Ig(x,y)$ can be finished using only one scan over the edge map B. Given a pedestrian candidate Hi bounded by (l, t, r, b), its sum of edge points can be very efficiently achieved by taking advantages of the integral image Ig. Like Fig. 7, $\text{edge}(H_i)$ can be easily calculated with the form

$$\begin{aligned}\text{edge}(H_i)&=(A+B+C+H_i)+A\text{-}(A+B)\text{-}(A+C)\\ &=I(r,b)+Ig(l,t)\text{-}Ig(l,b)\text{-}Ig(r,t).\end{aligned} \tag{15}$$

Based on Eq.(15), Eq.(13) can be performed very efficiently using one addition and two subtractions. Only the candidate H_i passing Eq.(13) needs to be further verified using Eq.(12).

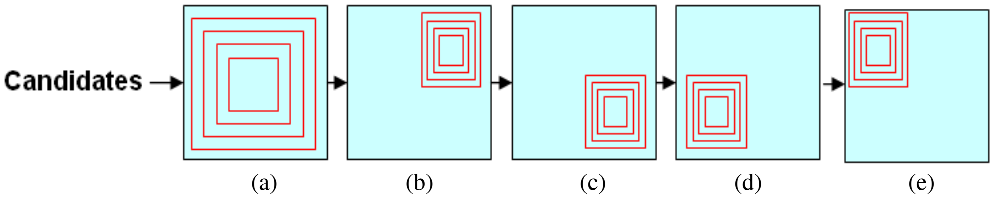


Fig. 8. Candidate checking using different ring structures

Integral image is a global descriptor to roughly verify each pedestrian candidate. In what follows, a ring structure is proposed for verifying a candidate more accurately according to its local and spatial features. In the ring structure, to filter out impossible candidates, a set of window masks with different sizes are used. Like Fig. 8(a), there are different sizes of masks centered at the origin of the template T to recursively remove unwanted candidates. Details of the ring structure for the candidate verification are described as follows.

Assume that the dimension of T is $w_T \times h_T$ for pedestrian verification. Then, the similarity between H_i and T is defined as

$$similarity(H_i,T)=\frac{\min(eRatio(H_i),eRatio(T))}{\max(eRatio(H_i),eRatio(T))}, \tag{16}$$

where $eRatio(X)$ is the ratio of edge points in X. It can be efficiently calculated using the technique of integral image. Eq.(16) is not stable when the numbers of edge points in H_i and T are few. For dealing with this case, Eq.(16) is further modified as follows:

$$\begin{aligned}similarity(H_i,T)=\min(&\frac{\min(eRatio(H_i),eRatio(T))}{\max(eRatio(H_i),eRatio(T))},\\ &\frac{1\text{-}\max(eRatio(H_i),eRatio(T))}{1\text{-}\min(eRatio(H_i),eRatio(T))}).\end{aligned} \tag{17}$$

Based on (17), we first use a rectangle window with the size $0.5w_T \times 0.5h_T$ to filter out impossible candidates. Then, like Fig. 8(a), the size of the rectangle window will gradually become larger with one pixel for candidate filtering. At the kth checking process, the dimensions of H_i and T will be $w^k \times h^k$, where $w^k = w^{k\text{-}1}+1$ and $h^k = h^{k\text{-}1}+1$. The checking process will be iteratively performed until the maximum size $w_T \times h_T$ is reached. Therefore, w^k ranges from $0.5w_T$ to w_T and h^k ranges from $0.5h_T$ to h_T. The rule for filtering out impossible candidates is:

$$H_i \text{ is filtered out if } similarity(H_i^k,T^k) < 0.5,$$

where H_i^k and T^k are the kth versions of H_i and T with the dimension $w^k \times h^k$, respectively. For examining whether other local features of a candidate H_i are similar to T, we can move the set of masks to other positions. Like Fig. 8(b), the center of the right-up block is used as the new origin to generate another ring structure for filtering impossible candidates. Other ring structures for the verification process are shown in Fig. 8(c), (d), and (e), respectively. With this structure, different impossible candidates can be very efficiently eliminated.

5 Grid-based People Counting

Due to the perspective effects, a pedestrian will have different visual appearances at different positions. The technique of template matching performs weakly to match two pedestrians if their appearances change significantly. To tackle this problem, this paper proposes a novel grid-based technique for people counting. In what follows, details of this grid-based technique are introduced.

5.1 Grid-based Template Matching

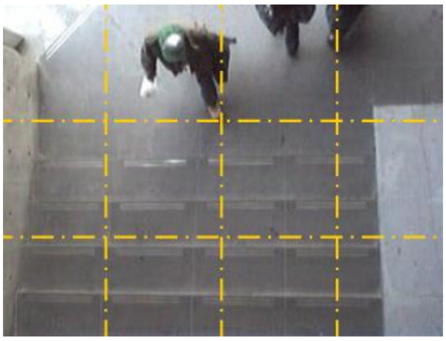


Fig. 9. Grid division of the observed region (Totally, twelve grids were used in this paper for people counting)

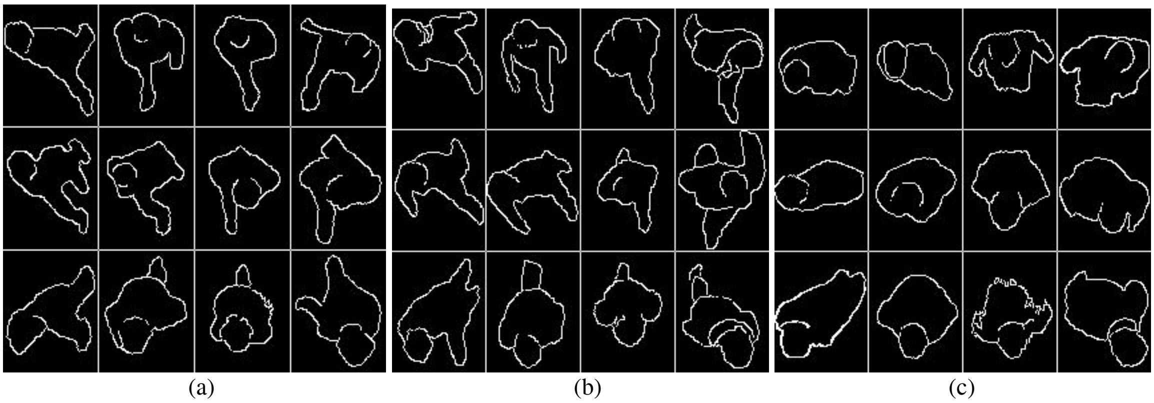


Fig. 10. Templates for walking down (a) Right foot moving first (b) Left foot moving first (c) Standing

To improve the robustness of template matching, a set of girds are created for recording the template's appearance changes at different positions. Thus, when a pedestrian moves at a specific position, its corresponding template will be generated for candidate verification. Since the pedestrian is verified according to the template chosen by its position, the perspective effect of the used camera can be reduced into a minimum. With the above idea, this paper divides the observed region into several grids. Like Fig. 9, 3×4 grids are used in this paper for people counting. Each grid uses six templates to record different pedestrian's appearance changes. The templates for walking down are shown in Fig. 10 and the ones for walking up are shown in Fig. 11. For each direction, the used templates are further classified into three classes according to the types of foot movement, i.e., right foot moving first, left foot moving first, and standing. The dimension of each template is 70×90.

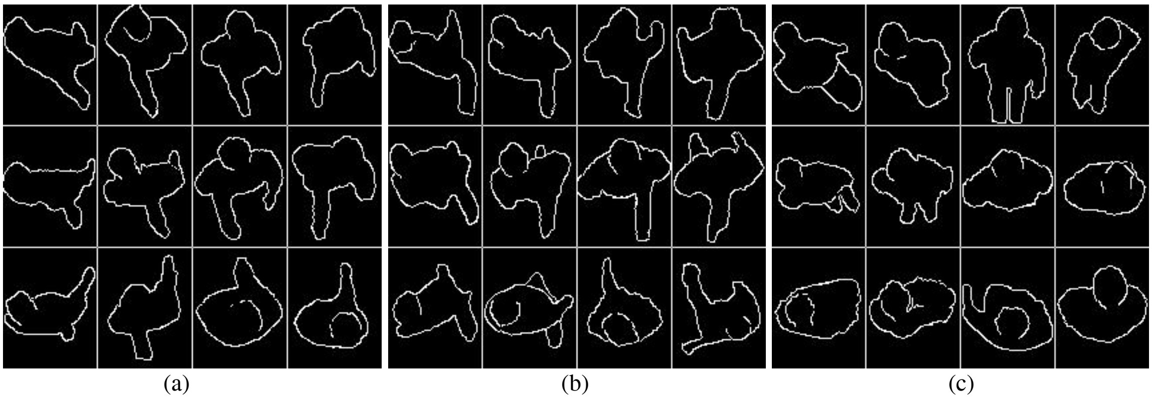


Fig. 11. Templates for walking up (a) Right foot moving first (b) Left foot moving first (c) Feet standing

This paper uses a grid-based template matching scheme to tackle the problem of perspective effects. When a moving object is detected at the kth grid, the kth category of pedestrian template is used for verification. Like Fig. 12, different grid uses different templates to verify different pedestrians. Assume that A is the set of extracted foreground pixels. For each pixel p_i in A, we generate a candidate H_i as a possible pedestrian. Although H_i may occupy more than one girds, since p_i is a pixel, it will occupy only one grid. If p_i is located in the kth gird, we use the kth category of template to verify p_i. Assume that C_k is the kth type of templates. Then, according to Eq.(12), H_i is a moving pedestrian if

$$S(H_i,C_k) < \theta_k, \tag{18}$$

where θ_k is the threshold to filter out impossible candidates from C_k. The value of θ_k can be obtained using hundreds of training pedestrian sequences.

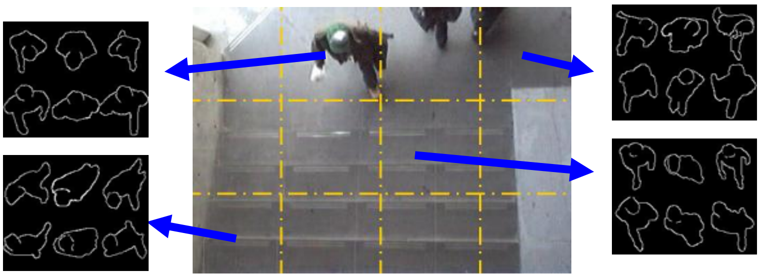


Fig. 12. Grid-based template matching for people counting; each grid uses different types of template to verify pedestrians

5.2 People Counting Using Tracking

When counting people, each pedestrian will appear in video frames several times. In order to avoid the problem of double counting, a tracking technique is used here for obtaining each pedestrian's trajectory. First of all, like Fig. 13, we divide the observed region into three strips. Here, zones 1 and 3 are called as "warning" areas and Strip 2 is named as "tracking" area. Assume that A is a pedestrian. If A moves from Strip 1 to Strip 2, it will be labeled as "entering". Otherwise, if it moves from Strip 3 to Strip 2, it is called "leaving". This information can provide two advantages for people counting. Firstly, since two categories "entry" and "leaving" are used to classify people, the task of people counting can be performed more accurately. Secondly, once the moving direction is obtained, a more efficient way can be used to calculate Eq.(18). In Eq.(18), if C_k is further classified into "walking down" and "walking up" classes, i.e., C_k^d and C_k^u, less templates can be used to verify H_i. If p_i is a walking-down pixel, H_i is a pedestrian if it satisfies

$$S(H_i,C_k^d) < \theta_k^d, \tag{19}$$

where $S(H_i,C_k^d)$ is defined in Eq.(12). If p_i is a walking-up pixel, H_i is a pedestrian if

$$S(H_i,C_k^u) < \theta_k^u. \tag{20}$$

Here, θ_k^d and θ_k^u are the thresholds to filter out impossible candidates from C_k^d and C_k^u, respectively. The number of templates used in Eq.(19) or Eq.(20) are only 3 which is less than Eq.(18).

After verification, we use a correlation technique [17] to find the relations of each detected pedestrian across different frames. Then the trajectory of each passing pedestrian can be obtained. According to the trajectory, if a pedestrian walks through the above three zones, it will be added for counting people.

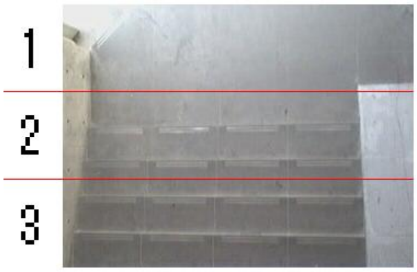


Fig. 13. Three zones used for people counting

6 Experimental Results

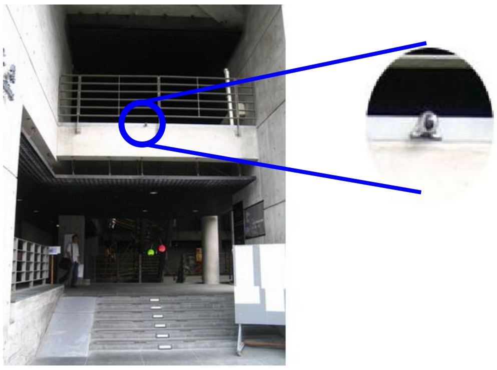


Fig. 14. One snapshot of our overhead camera system for people counting

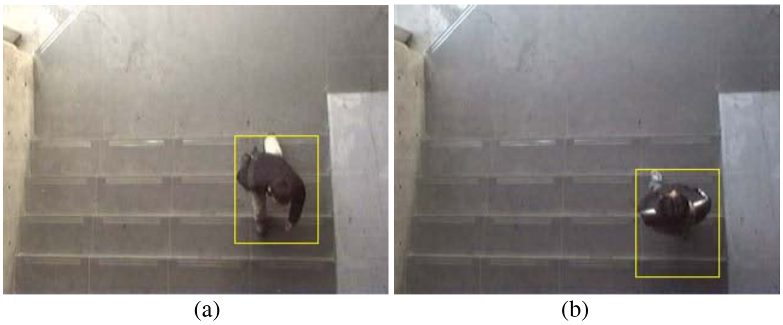

(a)
(b)

Fig. 15. Detection result when one person appeared

Fig. 16: Result of pedestrian detection when occlusions happened. All the pedestrians were correctly detected.

In order to demonstrate the performances of our proposed system to count people, thirty four video sequences were collected in this paper. Fig. 14 shows one snapshot of our camera system which used a overhead camera to count people. Fig. 15 shows the detection result when single pedestrian passed. Fig. shows the case when two persons appeared in the analyzed video frame. In the two cases, all the walking pedestrians were correctly detected. Fig. 16 shows the detection result of pedestrians when occlusions happened. It is noticed that there are different shadows appearing in (c) and (d). However, all the pedestrians were correctly detected.

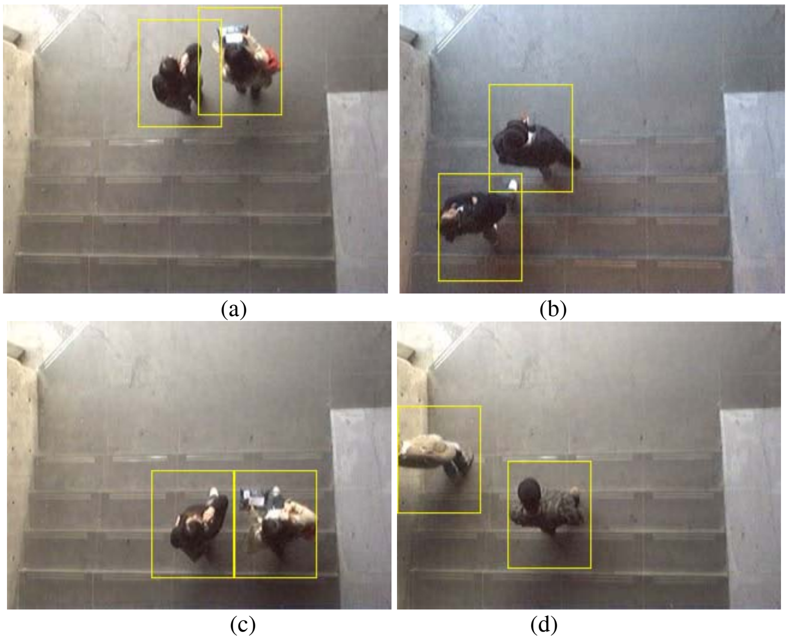


Fig. 16. Result of pedestrian detection when two persons appeared (All the pedestrians were correctly detected)

Fig. 17 shows the result of pedestrian detection when multiple persons appeared. In Fig. 17(a) and (b), there was no occlusion among the walking persons. But, in (c) and (d), persons had different occlusions. All these walking persons were correctly found and counted. Fig. 18 shows another set of detection results when multiple pedestrians passed. It is noticed that there were different shadows and occlusions appearing in the video sequences. However, our method still worked very well to detect them.

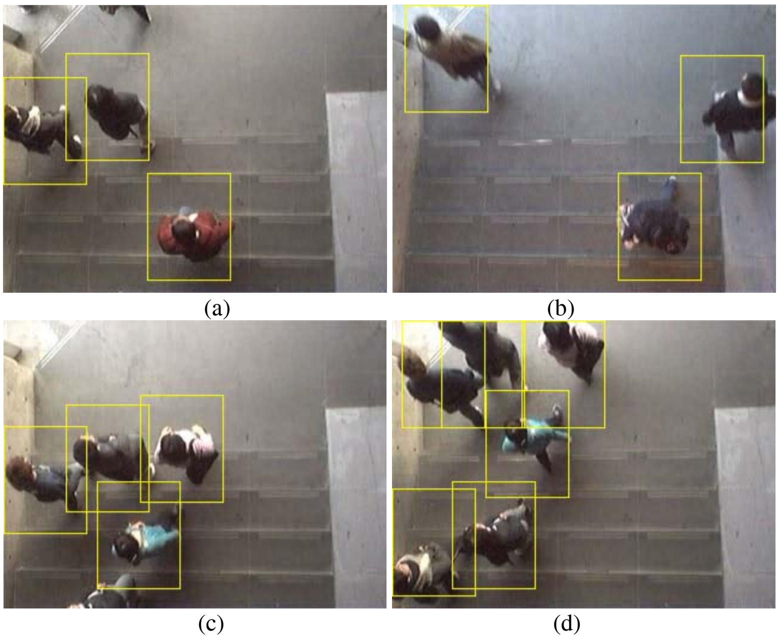


Fig. 17. Result of pedestrian detection when multiple persons appeared (a) and (b) Persons having no occlusion (c) and (d) Persons having different occlusions

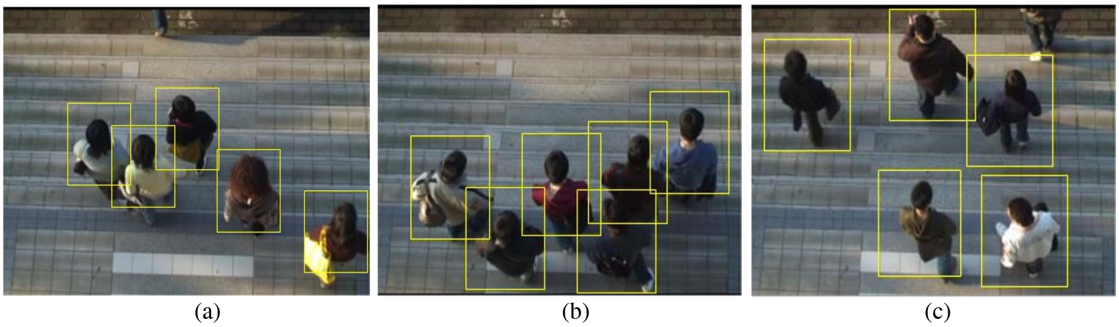


Fig. 18. Result of pedestrian detection when multiple persons appeared (It is noticed that different occlusions were included in the analyzed video)

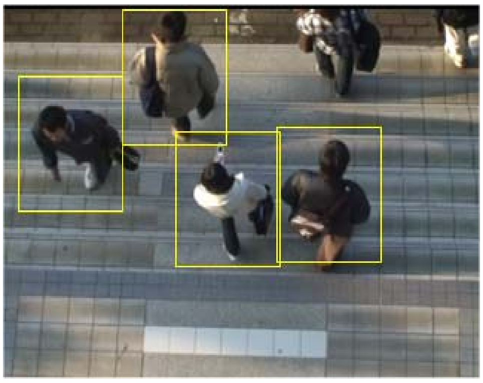


Fig. 19. Detection result when walking persons have different directions

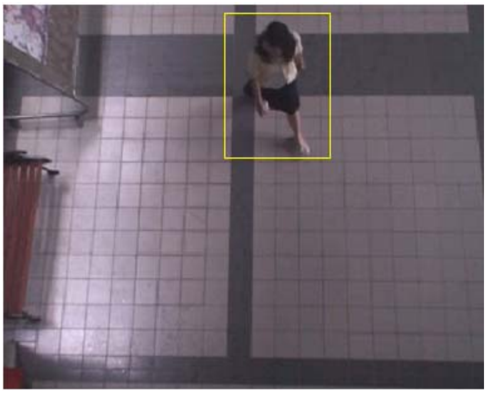


Fig. 20. Detection result when a female wore a skirt

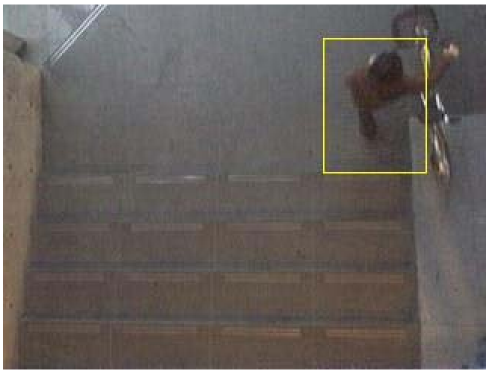


Fig. 21. Detection result when a person took a bicycle

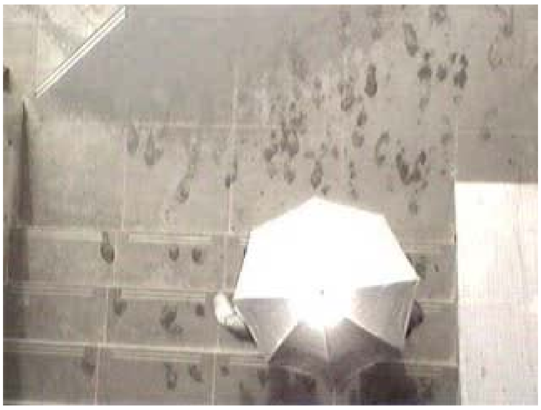
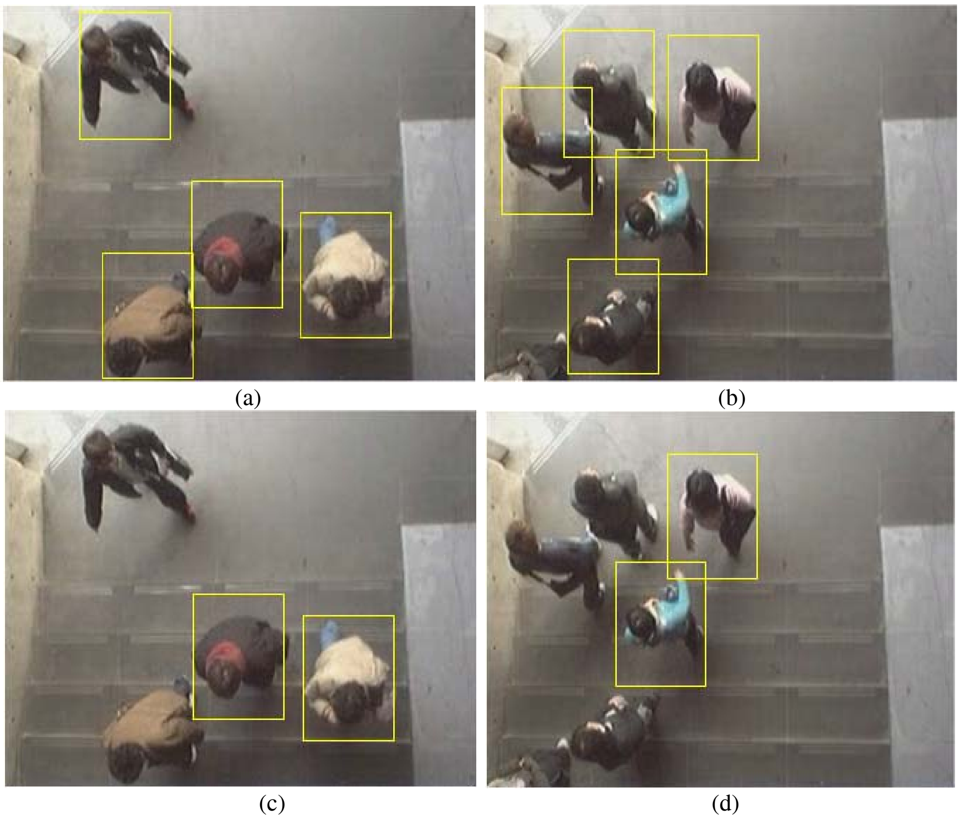


Fig. 22. Failure case when a person took an umbrella

Table 1. Accuracy analysis of people counting among different video sequences

Analysis / Types	No. of people	People detected	Accuracy
Moving up	268	259	96.64%
Moving down	213	204	95.77%
Total	481	463	96.26%



(a) (b)

(c) (d)

Fig. 23. Comparisons of people counting between our grid-based approach and the non-grid-based template matching scheme (a) and (b): Results of pedestrian detection using our grid-based scheme (c) and (d): Results of pedestrian detection using the non-grid-based template matching scheme

Table 2. Accuracy comparisons of people counting between the non-grid-based scheme and our grid-based one

Analysis / Types	Template matching without grids	Grid-based
Moving up	75.87%	96.64%
Moving down	73.59%	95.77%
Total	74.73%	96.26%

Table 3. Efficiency comparisons between our scheme with a ring structure, our scheme without a ring structure, and the template matching scheme without integral image

Methods	Grid-based scheme with ring structure	Grid-based scheme without ring structure	Template matching without integral image
Speed	0.0536 seconds	0.0719 seconds	1.1354 seconds

Fig. 19 shows the result of detection when pedestrians had different moving directions. Fig. 20 shows the case when a female wore a skirt. Fig. 21 shows another case when a pedestrian took a bicycle. In all the above cases, our method still worked successfully to detect and count them. Fig. 22 shows the failure case of our method to detect walking pedestrians. Since the walking person took an umbrella, our method failed to detect him. Table 1 lists the accuracy analysis of people counting from 34 video sequences. The average accuracy of our proposed method is 96.26%. Fig. 23 shows the comparison results between our grid-based method and the non-grid-based template matching scheme. (a) and (b) are the results obtained from our scheme. All the pedestrians were correctly detected. (c) and (d) are the results using the non-grid-based template matching scheme. Since the pedestrians have different appearance changes, five of them were missed. Table 2 lists the accuracy comparisons between the two methods. The average accuracy of the non-grid-based template matching is 74.73%. Clearly, our method performs very well to detect various pedestrians even if their appearances change very significantly. Table 3 shows the efficiency comparisons of pedestrian detection among our ring-based scheme, our grid-based scheme without the ring structure, and the matching scheme without integral image. Clearly, the ring-based scheme has the best efficiency. In addition to the efficiency comparison, two other methods were also implemented in this paper for accuracy comparison. They are, respectively, the head-based scheme [9] and the ratio-based scheme [8] which uses the ratio of moving areas to count people. Table 4 lists the accuracy comparisons among our scheme and the two methods. The head-based scheme was easily disturbed by clothes' colors and object shadows. The ratio-based scheme often failed to work if large background subtraction errors happed or people passed the door with a cart, bicycle, or trolley. All the above experiments have proved the superiority of our grid-based method in people counting.

Table 4. Accuracy comparisons among the head-based scheme, the ratio-based scheme, and our grid-based scheme

Methods	Head-based	Ratio-based	Grid-based Template Matching
Accuracy	78.59%	83.23%	96.26%

7. Conclusions

This paper has presented a grid-based method to effectively verify pedestrians so that different passing persons can be counted more accurately. To effectively detect foreground objects from videos, a novel background subtraction scheme using a minimum filter was proposed to significantly eliminate subtraction errors. Then, a grid-based approach was proposed for verifying each foreground object. Contributions of this paper are summarized as follows:

1. A subtraction scheme using a minimum filter was proposed. Thus, the problem of camera vibrations can be tackled and subtraction errors can be reduced into a minimum.
2. A grid-based method was proposed for effectively verifying pedestrians even though they have different appearance changes at different positions. This technique can significantly improve the accuracy of template matching in people counting.
3. A ring structure based on integral image was proposed for speeding up the process of candidate verification. Thus, different pedestrians can be verified and counted in real time.

Experimental results have shown our method is superior in terms of accuracy, robustness, and stability in people counting.

References

[1] S. Harasse, L. Bonnaud, M. Desvignes, “People Counting in Transport Vehicles,” *Transactions on Engineering, Computing, and technology*, Vol. 4, pp. 221-224, 2005.

[2] M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, A. Tibaldi, “Shape-based Pedestrian Detection and Localization,” *Proceedings of IEEE Conference on Intelligent Transportation Systems*, pp. 323-333, 2003.

[3] I. Haritaoglu, D. Harwood, L. S., Davis, “W4: Real-Time Surveillance of People and their Activities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp. 809-830, 2000.

[4] C. J. Pai, H. R. Tyan, Y. M. Liang, H. Y. M. Liao, S. W. Chen, “Pedestrian Detection and Tracking at Crossroads,” *Pattern Recognition*, Vol. 37, No. 5, pp. 1025-1034, 2004.

[5] V. Rabaud and S. Belongie, “Counting Crowded Moving Objects,” *Proceedings of IEEE Conference of Computer Vision and Pattern Recognition*, 2005.

[6] G. Antonini and J. P. Thiran, “Counting Pedestrians in Video Sequences Using Trajectory Clustering,” *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 16, No. 8, pp. 1008-1020, 2006.

[7] O. Masoud and N. P. Papanikolopoulos, “A Novel Method for Tracking and Counting Pedestrians in Real Time Using a Single Camera,” *IEEE Transactions on Vehicular Technology*, Vol. 50, No. 5, pp. 1267-1278, 2001.

[8] L. Snidaro, C. Micheloni, C. Chiavedale, “Video Security for Ambient Intelligence,” *IEEE Transaction on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol. 35, No. 1, pp. 133-144, 2005.

[9] G.P. Adriano, S.I.V. Mendoza, F.N.J. Montinola, P.C. Naval, “APeC: Automated People Counting from Video,” *Proceedings of International Conference of Security and Networking, Philippine*, 2005.

[10] A.J. Schofield, P.A. Metha, T.J. Stonham, “A System for Counting People in Video Images Using Neural Networks to Identify the Background Scene,” *Pattern Recognition*, Vol.29, No.8, pp. 1421-1428, 1996.

[11] L. Zhao and C. E. Thorpe, “Stereo and Neural Network-Based Pedestrian Detection,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 1, No. 3, pp. 148-154, 2000.

[12] T. Darrell, G. Gordon, M. Harville, J. Woodfill, “Integrated Person Tracking Using Stereo, Color, and Pattern Detection,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 601-608, 2002.

[13] P. Kelly, N. E. O. Connor, A. F. Smeaton, “Pedestrian Detection in Uncontrolled Environments Using Stereo and Biometric Information,” *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks*, pp. 161-170, 2006.

[14] K. Terada, D. Yoshida, S. Oe, J. Yamaguchi, “A Method of Counting the Passing People by Using the Stereo Images,” *Proceedings of 1999 International Conference on Image Processing*, Vol.2, pp.338-342, 1999.

[15] D. B. Yang, H. H. G. Banos, L. J. Guibas, “Counting People in Crowds with a Real-Time Network of Simple Image Sensors,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 122- 129, 2003.

[16] A. Prati, I. Mikic, M. Trivedi, R. Cucchiara, “Detecting Moving Shadows: Algorithms and Evaluation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 7, pp. 918-923, 2003.

[17] M. Sonka, V. Hlavac, R. Boyle, “Image Processing, Analysis and Machine vision,” *Brooks/Cole Publishing Company*, 1999.