Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2010, Article ID 675784, 9 pages
doi:10.1155/2010/675784

Research Article

Localized Detection of Abandoned Luggage

Jing-Ying Chang, Huei-Hung Liao, and Liang-Gee Chen

DSP/IC Design Lab, Graduate Institute of Electronics Engineering and Department of Electrical Engineering, National Taiwan University, Taipei 10617, Taiwan

Correspondence should be addressed to Jing-Ying Chang, jychang@video.ee.ntu.edu.tw

Received 15 December 2009; Revised 19 April 2010; Accepted 2 June 2010

Academic Editor: ChangIck Kim

Abandoned luggage represents a potential threat to public safety. Identifying objects as luggage, identifying the owners of such objects, and identifying whether owners have left luggage behind are the three main problems requiring solution. This paper proposes two techniques which are "foreground-mask sampling" to detect luggage with arbitrary appearance and "selective tracking" to locate and to track owners based solely on looking only at the neighborhood of the luggage. Experimental results demonstrate that once an owner abandons luggage and leaves the scene, the alarm fires within few seconds. The average processing speed of the approach is 17.37 frames per second, which is sufficient for real world applications.

1. Introduction

Intelligent and automatic security surveillance systems have recently become an active research focus due to continuously growing public demand for such systems. Terrorist attacks frequently employ bombs, such as car bombs, suicide bombs, and luggage bombs. Modern technology cannot fully prevent such attacks, and security officers can easily miss their targets. However, compared with the two previous forms, luggage bombs are relatively difficult to hide and there is generally ample time to either deal with the bombs or organize an evacuation. Humans thus have a better chance to prevent destruction arising from luggage bombs. Therefore, to achieve early detection of these threats with the assistance of automatic security systems, the ability to reliably detect suspicious items and identify their owners is urgently necessary in various venues such as airports and train stations.

Previous studies have given several definitions of a luggage abandonment event [1–5]. This study follows three similar but slightly different rules [5]. (1) Contextual rule: luggage is considered unattended after the person who entered the area in possession of that luggage concerned is no longer in close proximity to it. (2) Spatial rule: luggage is considered unattended when its owner is outside of a small neighborhood around the luggage. (3) Temporal rule: If the owner of a luggage leaves the area without the luggage, or if the luggage has been left unattended for more than T consecutive seconds, the luggage is considered abandoned.

1.1. Related Works. The task of abandoned luggage detection in surveillance video generally comprises three stages: The first stage localizes candidate abandoned luggage items in the video. The second stage locates and tracks the luggage owner(s), providing a trajectory for subsequent probabilistic reasoning. The final stage assesses a probability or confidence score for the luggage-abandonment event based on information obtained during previous stages. The three stages all represent distinct research areas with their own rich literature. Various existing algorithms may employ different methods for different stages.

The first stage of locating candidate abandoned luggage items within the video frame is performed using two types of techniques: those that utilize the technique of background subtraction [6–8], and those that do not [9, 10]. As generally acknowledged, object detection and recognition is an instinctive and spontaneous process for human visual system. However, implementing a robust and accurate computer vision system capable of detecting

relevant objects in monitored areas has proved challenging. The main difficulty is the appearance of an object can vary significantly due to viewpoint changes, scene clutter, ambient lighting changes, and in some cases even shape changes (for nonrigid objects such as human body). Consequently, the same object may present enormously different images under various viewing conditions. Background subtraction works reasonably well when the camera is stationary and the change in ambient lighting is gradual. For those approaches without background subtraction, a set of discriminative features of objects must be learned through machine-learning algorithms to enable subsequent detection of these objects.

Most existing event detection methods incorporate some form of tracking algorithm [4–7, 10–13]. In most cases, tracking is performed on all detected moving objects or foreground blobs. However, because of occlusion and fixed camera angle, this comprehensive tracking frequently results in errors such as identity switch (when two objects in close proximity switch identities), which is difficult to avoid and occurs in many PETS 2006 [14] demonstration sequences—such as those in [13].

The final stage of determining whether an alarm is necessary is performed deterministically. In a deterministic system, an event is declared to have occurred if particular criteria are satisfied. A few reports employ a probabilistic framework for event modeling, with an event being deemed to have occurred if its confidence score exceeds a certain threshold [5]. The probabilistic approach gives users increased flexibility to set thresholds, and thus system sensitivity, and a better understanding of how the reality of a situation.

1.2. Contributions. The contribution of this paper is as follows. First, this paper proposes the foreground-mask sampling to localize the candidates of abandoned luggage items by calculating the intersection of a number of background-subtracted frames which are sampled over a period of time. Abandoned luggage items are assumed to be static foreground objects, and thus appear in this intersection. Since this approach requires no prior learning of luggage appearance in any form, luggage of all shapes, sizes, orientations, viewing angles and colors can be successfully localized without the need for training data and associated constraints.

Second, selective tracking is applied following identification and localization of a suspicious luggage item. This approach seeks the owner of the luggage in a neighborhood around the detected item. If the owner is found within this neighborhood, the luggage is assumed to be being attended by its owner and thus to require no further processing. However, if no owner is found, the tracking algorithm returns to the frame in which the owner was still attending the luggage, and starts tracking the owner from that point. Selective tracking only tracks the owner, and ignores other irrelevant moving objects in the foreground. Accordingly, the computational requirements of selective tracking are less than in previous works.

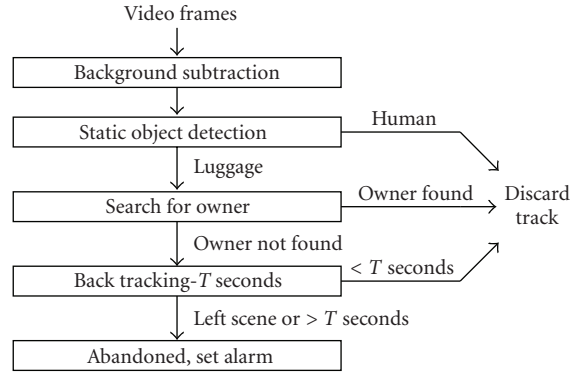

Figure 1: System work flow.

Third, the reliability of the tracked trajectory of the owner is used in evaluating the overall confidence score of the luggage-abandonment event. An alarm is triggered if the overall confidence score exceeds a given threshold, which is adjustable by the user to achieve varying degrees of system sensitivity. Figure 1 presents the system work flow.

The remainder of this paper is organized as follows. Section 2 details how the foreground-mask sampling approach localizes the suspicious luggage. Section 3 then elucidates the selective tracking module. Section 3 then presents the experimental results, indicating the tracked owner and alarm time. Finally, Section 4 draws conclusions.

2. Foreground-Mask Sampling

During the first stage of the system, the foreground-mask sampling attempts to localize static and possibly abandoned luggage items within the camera view. This technique imitates the natural human ability to focus attention exclusively on objects of interest. The algorithm identifies the objects (in this case abandoned luggage items) via logical foreground-background reasoning, while ignoring all irrelevant objects within the same scene. The appearance-based model is not used in locating suspicious luggage items, and thus can deal with luggage of any color and shape and is not affected by different viewing angles.

Since abandoned luggage is assumed to remain static for more than T consecutive seconds, a number of video frames are collected from the past T seconds; the number of frames is set to n, and is evenly distributed across the T second sample. In the subsequent experiment, detection performance is not significantly influenced with changing n.

The background model is constructed using selected clean frames from the standard test sequences in which foreground clutter is minimized. In situations in which clean background frames are unavailable, frames with minimal foreground clutter are used. The background model comprises the average of the selected frames, with a standard deviation calculated on each background pixel to consider the pixel variation. This study does not employ dynamic update of the background model, since the tested video sequences contain minimal ambient lighting change,

and for such sequences a one-time construction of a static background model provides reasonable performance. The background-subtraction-based object detection is not constrained to the following method. It can be replaced by other state-of-the-art approaches [15, 16] for complex environments.

Background subtraction is then performed on n sample frames to produce n corresponding foreground images. Specifically, $M_k(i, j)$ represents the k foreground masks, F_k denotes the k sample frames, B represents the background image, and std is the standard deviation image, with (i, j) denoting pixel position within the image

$$M_k(i,j) = \begin{cases} 1, & |F_k(i,j) - B(i,j)| \geq w(i,j) \cdot \mathrm{std}(i,j) \\ 0, & |F_k(i,j) - B(i,j)| < w(i,j) \cdot \mathrm{std}(i,j), \end{cases} \tag{1}$$

where $k = 1$ to n, and $w(i, j)$ denotes a weight on the standard deviation at pixel (i, j), which is smaller when i is small (upper part of the image) and larger when i is large (lower part). The weight $w(i, j)$ is implemented as a function of image row i to consider the gradual change in image resolution in the row-wise, vertical direction

$$w(i,j) = \left\lceil \frac{c}{h} \cdot i \cdot W \right\rceil, \tag{2}$$

where h denotes image height; c represents the number of quantization steps; W denotes the weight of top-most pixels.

This gradual variation in image resolution results from using a camera angled to look down on the scene from above, a characteristic shared by a majority of surveillance cameras. Images produced by these cameras have a higher resolution in the lower part of the screen in which objects appear larger and the camera is closer to the scene; but as the objects move away from the camera, the foreground instances of the objects move upward in the image plane and become smaller, resulting in lower resolution and decreased image quality, see Figure 2. The use of the $w(i, j)$ weighting raises the foreground threshold for the lower part of the image where the resolution is better and lowers the threshold where the resolution is poor, compensating for the change in resolution resulting from a tilting camera angle. The modification using the variable weight $w(i, j)$ is reasonably effective in the absence of specific camera parameters. Although this method may lack the precision offered by meticulous calibration using camera parameters, for the purpose of this part (namely distinguishing foreground and background), it represents an adequate substitute. It is important to note that the model of the gradual change in weight $w(i, j)$ is not constrained to the method presented here. Any function monotonically increasing with i can replace $w(i, j)$. We use quantized steps here in $w(i, j)$ to make sure the ratio of the weights on the upper frame and those on lower frame can be kept at a reasonable value.

n foreground masks $M_k(i, j)$ are merged and their intersection taken as the static foreground object mask $S = \bigcap_{k=1}^{n} M_k$. Filtering is then conducted on S to remove irrelevant and sporadic noisy pixels, connected component analysis is subsequently performed. A white (valued 1) block on S indicates a region that has remained in the foreground of all the n sample frames over the previous T-second period, and therefore this region should likely correspond to either a static abandoned luggage item or a stationary human. The tracking module, which is detailed in the next section, then analyzes the region and further localizes it if it is determined to be a static luggage item. Figure 3 shows an example. S presents candidate abandoned luggage items. The localized targets provide search regions for subsequent tracking and higher level event reasoning.

Figure 2: AVSS 2007 video dataset. Images captured via a typical surveillance camera are looking down, causing the lower part of objects to appear larger and the upper part to appear smaller.

3. Selective Tracking Module

The system presents information on the locations of suspicious items after obtaining S. All static foreground objects are assumed to be either humans or luggage items. Each foreground region in S is checked to determine whether it is a human via a combination of skin color information and body contours. If the region is identified as a human, it is discarded because the object of the search is abandoned luggage items. If the region is identified as not a human, it is assumed to be a luggage item. A local search region is constructed around the detected luggage to see whether its owner is in close proximity in the present frame at time t. If the owner is found, the region is again discarded because the owner exhibits no intention of abandoning the luggage. However, if the owner is not located near the luggage, the algorithm goes back in time for a predefined Δt seconds to the frame at time $(t - \Delta t)$ when the owner was still attending the luggage and begins tracking the owner from that point (at time $(t - \Delta t)$). The tracking module also employs skin color information and human body contour to track the owner.

Δt is set to 30 seconds based on the assumption that when an isolated luggage item is first detected in a scene, its owner must have been in close proximity to the item until shortly before detection. This assumption is valid because if

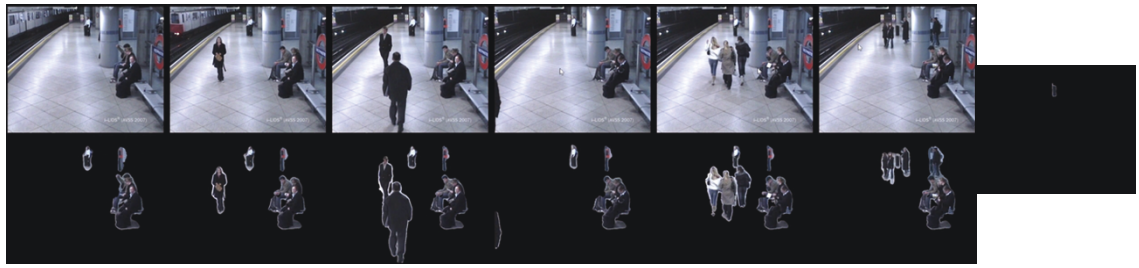

Figure 3: Foreground-mask sampling. The first row shows input frames while the second row shows corresponding foreground images. After obtaining the intersection result, two filters are applied to remove certain static regions which are not the luggage items. One filter is the human classifier introduced in Section 3, while the other removes unreasonably large regions. Image on the right represents the intersection of the n foreground images sampled over T seconds, and it was obtained after removing nonluggage regions.

the owner has been absent for some time, the isolated luggage item will be detected faster using the foreground-mask sampling technique. Furthermore, owners who abandon their luggage with criminal intention would generally want to avoid attention and thus are unlikely to loiter; instead they will remain constantly with their luggage prior to abandonment. Therefore, in the case in which multiple people surround the abandoned luggage, the person closest to the luggage is assumed to be the owner.

The actual implementation uses a cache mechanism to store the information from the previous backtracking. When the system needs to repeat back-tracking around a single abandoned luggage item, the system needs only to update some of the stored information. For example, if two 30-second back-tracks overlap by 20 seconds, the information regarding the first 20 seconds of the second back-tracking can be directly obtained from the last 20 seconds of the first back-tracking, which is cached in the system. Overheads associated with recollecting thus are eliminated. This mechanism provides sufficient computational reduction in the back-tracking procedure and guarantees real-time performance on live streaming surveillance videos.

Because suspicious luggage items have been identified, tracking can be performed solely and selectively on their owners. This mechanism closely mimics the human ability to notice and track only objects of interest even under a highly cluttered background; for example, humans have a natural ability to identify familiar faces even in such crowded environments as an airport pick-up area.

The implementation of detection and tracking using skin color information and human body contour is detailed below, and its integration into the motion prediction of the tracking module.

3.1. Cr Color Channel with Human Skin. Human skin signal response is significantly larger in the YCbCr color space than the commonly used RGB color space. Due to significant blood flow, human skin responds strongly to the Cr channel in the YCbCr space, irrespective of skin color [17]. Accordingly, the Cr channel of skin color is used for human face localization because in situations involving severe occlusion (crowded scenes with people overlapping one another), human face is the most visible body part when viewed with a typical surveillance camera positioned looking downwards from a height.

To find the face of the owner, the method proposed by Chai and Ngan [18] is adopted. A search region is first constructed around the suspicious static luggage item. Background subtraction is then performed on RGB color space within the region. An RGB foreground of the region is obtained and then converted to the YCbCr color space, and the Cr channel is retained, as illustrated in Figure 4. Background subtraction is performed within the search region prior to conversion to YCbCr color space because Cr is a channel representing the difference of red color, and thus the face signal is stronger when background clutter is removed. The Cr channel response is then used to locate the face of the luggage owner, while simultaneously locating human body contour information as explained below.

3.2. Improved Hough Transform on Body Contour. Cr channel responds to red within the search region, which in some cases may include other reddish objects besides the face of the owner. Therefore, a new mechanism for reliably detecting the presence of the luggage owner is employed. Most surveillance cameras are mounted to the ceiling. The human upper-body contour, which comprises the head-shoulder silhouette, is visible in most cases. Hence, it is a useful feature to detect a human. To use this feature, we have to assume people in scenes do not wear hats, or only wear caps that can keep the head-shoulder contours. The head-shoulder contour, as inspired by [9], is used under the Hough transform (HT) to detect human upper-body within the search region. Figure 5 depicts the contour and the used notations. HT is an edge-based detection method. Readers may wonder most cameras nowadays produce blurry images in real-world surveillance systems, which edge-based methods will fail to detect contours. This situation will not last for a long time since many companies (such as AXIS communications [19] and Arecont Vision [20]) have already provided high-end surveillance IP cameras which can produce stunning image quality.

HT is a morphological tool which, in its simplest form, maps a straight line in normal space to a point in parameter space [21]. A generalized version of HT is utilized to localize

(a) (b)

Figure 4: (a) input video frame with localized search region indicated by red circle. (b) the Cr detection result within the search region.

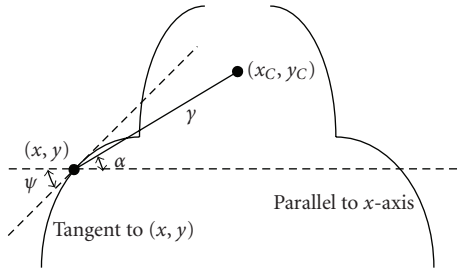

Figure 5: Head-shoulder contour under HT. In template generation, the relative position of (x_C, y_C) to (x, y) is recorded in (r, α); in contour matching, the pixel value on the assumed position of (x_C, y_C) is incremented by 1 on the detection map.

contour of an arbitrary shape. The algorithm comprises two stages: template generation and contour matching.

During template generation, given a predefined head-shoulder contour, as in Figure 5, the HT algorithm first establishes a center point (x_C, y_C) of the face for the contour template. The algorithm then runs through all edge points (x, y) on the contour template and for each point records the ψ (angle with respect to the horizontal direction), r (distance with respect to the center point) and α (angle with respect to the center point). The ψ lies between 0 and 180 degrees, and thus serves as the bin-index with which the (r, α) pair is recorded into a 180-bin reference table. Multiple pairs of (r, α) can be recorded under the same ψ-angle bin in the reference table. After traversing all the points on the contour template, the template generation and reference table are completed.

The next stage performs contour matching on the edge image of the input video frame. The edge image is obtained by filtering the original image using a 3×3 Sobel kernel. First a detection map with equal size to input video frame is created and initialized with zeros. The HT algorithm again traverses all edge points on the edge image, calculating the ψ angle of each. For an edge point $E(x, y)$ on the input edge image with ψ angle of m degree, all (r, α) pairs under that specific mth bin are accessed. Furthermore, for each of the (r, α) pairs under this bin, $E(x, y)$ represents the start point and the associated coordinate pair is calculated as

$$(x_{C'}, y_{C'}) = (x + r\cos\alpha, y + r\sin\alpha). \quad (3)$$

The pixel value at location $(x_{C'}, y_{C'})$ on the detection map is increased by 1. Once all (r, α) pairs in this bin have been processed, the algorithm proceeds to the next edge point on the input edge image.

An improved means of implementing the HT technique is proposed based on [22]. Besides accessing all (r, α) pairs under the mth bin of the reference table, a Gaussian-weighting system is employed and centered on the mth bin—which has a width of Δm. In the experiments, Δm is set to 5 and $(2\Delta m + 1)$ bins are accessed, with their respective weights given by a Gaussian distribution g, where $g = 1$ for the center mth bin and $0 < g < 1$ for neighboring $2\Delta m$ bins. g decreases with respect to distance from the center bin. For these $2\Delta m$ neighboring bins, $(x_{C'}, y_{C'})$ is also calculated for each (r, α) pair under these bins, and the pixel value at the corresponding location on the detection map is incremented by the given weight g of the bin. Figure 6 presents an example.

This modification is made because for a ψ angle computed from an edge point on the input edge image, an inherent error arises from pixel quantization and angle quantization, and thus the ψ angle obtained at best indicates only a small range of neighboring angles. This small range is modeled by applying a Gaussian-weighting system to a range of ψ-angle bins. Besides, by allowing the ψ angle to vary within a limited range, the system can handle human head-shoulder contours that are slightly out of alignment given a perfect frontal image, and then allowing some variance in pose.

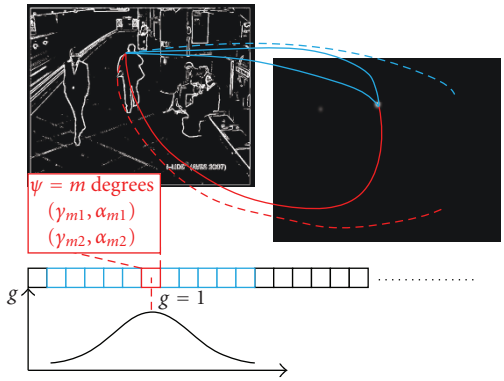

Figure 6: The origin of the two red lines has a ψ angle with degree m, corresponding to the red, mth, bin which contains two (r, α) pairs. Solid lines denote correct matches, while dashed lines represent noise. Different points on the head-shoulder contour in the edge image converge to a local maximum on the detection map on the right. At the bottom is the 180-bin reference table; a Gaussian weighting g is shown below the table, with the center bin labeled in red and the neighboring ten bins in blue.

Once all edge points on the input edge image have been traversed, the supposed center point of the contour of interest in the edge image corresponds to a local maximum in the detection map. Figure 7 displays the detection results provided by the generalized version of HT and compares them with the original implementation of HT. Since the contour detection is performed within the search region, interference from irrelevant contours is minimized. In the implementation, because people at different locations may have different-sized silhouettes, a few different-sized templates are applied simultaneously to locate head-shoulder contours.

With the assistance of multiple cues to detect owners, even the color of abandoned object is close to skin color, the object will not be recognized as a human since it has no head-shoulder contour.

3.3. Integration into Motion Prediction. For identifying the head-shoulder position of the luggage owner in a single frame, the color information from the Cr channel and the upper-body contour information from the generalized HT algorithm are combined. Motion prediction is employed to further exploit the temporal relationship between successive frames. Prediction of owner location in the next frame is based on their location in the current frame, and in previous frames with exponentially decaying weights. Specifically, $r(t)$ denotes the position vector of the owner at time t, and the prediction for time $t+1$ can be formulated as

$$r(t+1) = r(t) + \Delta r, \tag{4}$$

where Δr is generated recursively via motion prediction and given by

$$\Delta r_t \longleftarrow \alpha \cdot \Delta r_{t-1} + \beta \cdot (r(t) - r(t-1)), \tag{5}$$

where $\alpha + \beta = 1$, $\alpha > 0$, $\beta \geq 0$. The fact that Δr is calculated recursively ensures that past information is considered and past influences decay exponentially with time. In the implementation, the exponential smoothing coefficients α and β are empirically determined to be 0.4 and 0.6, respectively.

Three measures are used to calculate the probability score for the trajectory of the luggage owner, which is then used to obtain a confidence score for the luggage-abandonment event. The three measures include the differences between the prediction from the last frame and the detection on the present frame, in location, size, and color histogram of the luggage owner [10]. The probability score increases with closeness of prediction and detection. Let P_{TOTAL} denote the probability score combining the measures; let P_{POS}, P_{SIZE}, and P_{CH} denote the scores of the position measure, size measure and color-histogram measure, respectively; finally, let r represent the position vector, s the size (in pixel area), and c the color histogram. The three scores are defined as follows, with subscript P corresponding to prediction and D to detection

$$P_{\text{POS}}(r_{\text{P}}, r_{\text{D}}) = \exp\left(-\frac{(x_{\text{P}} - x_{\text{D}})^2}{\sigma_x^2}\right) \cdot \exp\left(-\frac{(y_{\text{P}} - y_{\text{D}})^2}{\sigma_y^2}\right),$$

$$P_{\text{SIZE}}(s_{\text{P}}, s_{\text{D}}) = \exp\left(-\frac{(s_{\text{P}} - s_{\text{D}})^2}{\sigma_s^2}\right),$$

$$P_{\text{CH}}(c_{\text{P}}, c_{\text{D}}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{D^2}{2\sigma^2}\right), \tag{6}$$

where D is a histogram distance by any distance measuring method. We use the χ^2 distance [23], as in $D^2 = \sum_{i=1}^{256} (c_{\text{P}}(i) - c_{\text{D}}(i))^2 / c_{\text{P}}(i) + c_{\text{D}}(i)$ to get the best result. The total probability score is calculated by combining the above three measures, each with a scale factor λ, so they total 1, as follows:

$$P_{\text{TOTAL}} = \lambda_{\text{POS}} P_{\text{POS}} + \lambda_{\text{SIZE}} P_{\text{SIZE}} + \lambda_{\text{CH}} P_{\text{CH}}. \tag{7}$$

The three probabilities serve more as comparative than absolute values. A change in the standard deviations of these probability calculations would similarly affect all probabilities thus calculated, with the most probable detection still having the highest probability ranking. Empirical values thus are assigned to the standard deviations, and parameter selections in the experiment produce insignificant effects.

4. Experimental Results

The proposed method is tested using surveillance datasets provided by AVSS 2007 [24] and PETS 2006 [14]. According to the definition of the luggage-abandonment event used in this study, the system should fire an alarm when the luggage owner leaves the scene without their luggage, or when the luggage is left unattended for T consecutive seconds. Table 1 lists the system alarm time. Since this definition differs slightly from that used in AVSS 2007 ($T_L = 60$, owner left scene) or PETS 2006 ($T_L = 30$, owner left luggage), ground truth data from past conferences may not be directly

Table 1: Alarm time (second).

Sequence	Owner break	Left scene	G. T.	Alarm	Diff.
AVSS2007 Easy	114.80	119.76	180.00	179.76	−0.24
AVSS2007 Medium	100.88	102.64	162.00	162.64	+0.64
AVSS2007 Hard	101.08	102.28	162.00	162.28	+0.28
PETS2006 Seq. 1	85.88	90.52	113.72	120.52	+6.80
PETS2006 Seq. 2	61.92	65.04	91.84	95.04	+3.20
PETS2006 Seq. 4	72.88	76.36	104.08	106.36	+2.28
PETS2006 Seq. 5	80.28	83.04	110.56	113.04	+2.48
PETS2006 Seq. 6	68.44	73.96	96.88	103.96	+6.08
PETS2006 Seq. 7	60.68	—	93.96	91.60	−2.36
Average difference					2.71

"G. T." represents "ground truth", which means the alarm time defined in the dataset. "Diff." represents "difference", which indicates the time difference between "alarm time" and "ground truth". The alarm is fired after the owner left the scene or left luggage unattended for T consecutive seconds. T is 60 in AVSS2007 dataset and T is 30 in PETS2006 dataset. The maximum time difference is +6.80, and the average absolute time difference is 2.71.

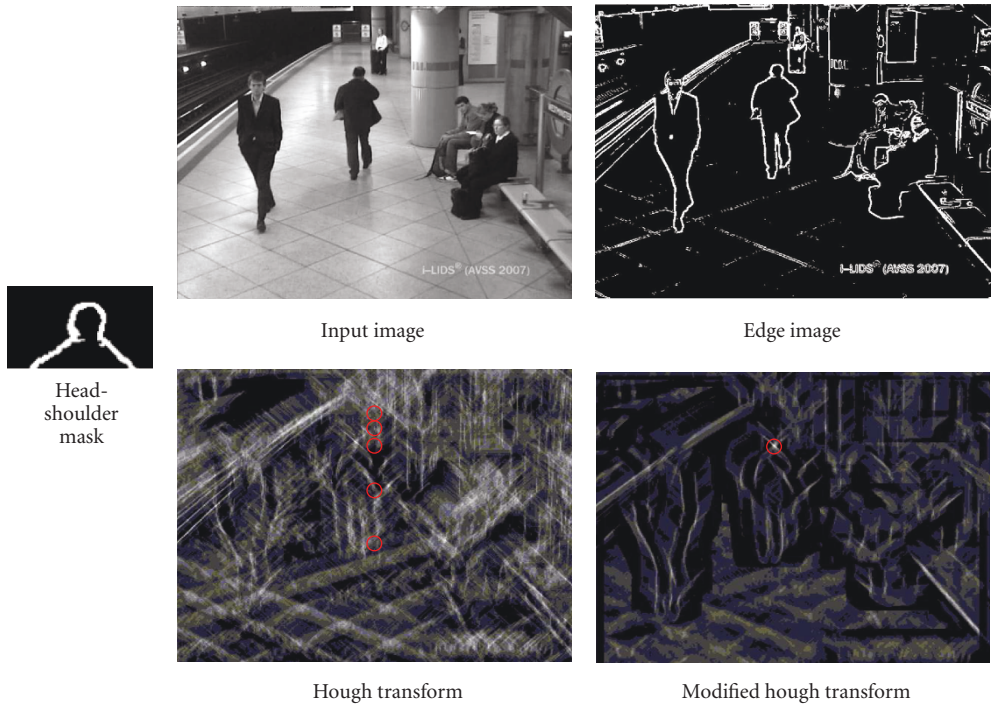

Figure 7: The leftmost image is the queried head-shoulder contour. Given the input image, the edge image is obtained via 3×3 Sobel convolution. Adopting traditional HT will generate a detection result giving more false positives. The improved HT method yields a single large response at the correct location of the most visible head-shoulder contour in the input edge image.

comparable. Therefore, the actual time when the owner breaks contact with the luggage is listed as a reference for performance evaluation.

The AVSS 2007 dataset contains three cases with different difficulty levels: easy, medium and hard. The easy case contains objects with larger appearance, activities closer to the camera and less scene clutter; as the difficulty level raises, objects shrink and clutter increases. The proposed approach successfully detects the abandoned luggage in all three cases. The owner in the easy case is tracked continuously until leaving the scene without his luggage, triggering an alarm event. In the medium and hard cases, the owners pass behind a large pillar before leaving the scene without their luggage, and both are occluded for about 1.5 seconds. The proposed tracking engine is unable to follow the owner through the occlusion, and thus the owner is deemed lost; therefore alarms are also triggered in these two cases.

The PETS 2006 datasets contains seven cases. In videos 1, 2, 4, 5, and 6, the luggage owners leave the scene without their luggage, and the proposed method has successfully issued an alarm in all these five cases while tracking the owner continuously until they are no longer within camera view.

Table 2: Comparison of AVSS2007 dataset.

Approach	Tested events	True detection	False alarm
[25]	3	3	2
[1]	8	8	4
Our method	3	3	0

True detection and false alarm of AVSS2007 dataset. Because i-LIDS did not provide all test cases for free, only [1] tested another two sequences.

Table 3: Comparison of PETS2006 dataset.

Approach	Tested events	True detection	False alarm
[25]	1	1	0
[1]	6	6	0
Our method	6	6	0

In video 3 the owner remains continuously with the luggage, and therefore no alarm is raised. In video 7, the owner wanders before finally leaving the scene without his luggage; the trajectory of the actively moving owner contains too many abrupt changes in speed and direction for the present motion prediction algorithm to successfully follow. The tracker lost the owner 34 seconds after leaving the luggage, while an alarm is triggered at 30 seconds because the owner left the luggage too far for 30 consecutive seconds. Figure 8 shows some labeled scene shots. Using different system parameter settings for all D1 (720×576) size-test videos, the processing speed of the proposed approach running at 2.66 GHz (E6750, Intel) with 4 GB DDR-RAM is 17.37 frames per second on average, which is sufficient for real-world applications.

Detecting abandoned luggage by tracking all objects in a scene is not only computationally extremely costly, but also prone to failure under overmuch occlusion. Hence, both the works [1, 25] adopted background-subtraction-based approaches. However, pure background-subtraction-based approaches without human detector may cause false alarms if a person stops moving for a short time. Our method, adopting head-shoulder contour and skin color detector to identify and track human selectively, can prevent these false alarms, which happen in [1, 25]. The results are shown in Tables 2 and 3. Furthermore, since we do not track all objects, this method can work in real-time.

5. Conclusion and Future Work

This paper presents a localized approach for detecting abandoned luggage in surveillance environments. Through foreground-mask sampling, only the object of interest is localized, while filtering out all irrelevant, interfering agents. Tracking thus can be performed in a more selective and localized manner. An improved implementation of the HT for detecting the contours of the upper-body is also proposed for use in tandem with skin color detection. These techniques make abandoned luggage detection become a real-time system, which can run at 17.37 frames per second on average.

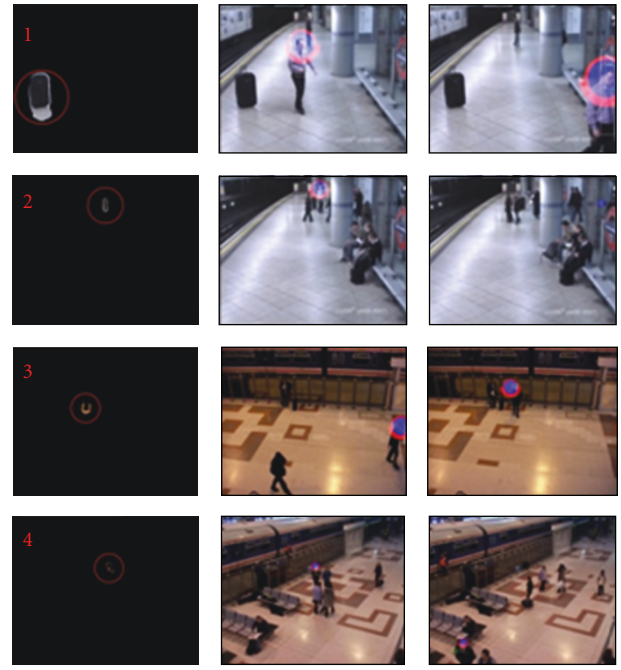

Figure 8: Sequence 1, 3, and 4: (from left) static luggage detected; owner tracking starts; owner leaves the scene, alarm triggered. Sequence 2: static luggage detected; owner tracking starts; owner lost due to occlusion, alarm triggered. Sequence 1 and 2 are from AVSS 2007 dataset. Sequence 3 and 4 are from PETS 2006 dataset.

In the future, the proposed approach is extended to a multicamera network in which coordination of various cameras enables cues to be gathered from multiple perspectives and information to be relayed from one to another camera. Besides, the approach is generalized to include different viewing angles on the human form. Currently, our approach can detect multiple abandoned objects. The alarm will fire at the first abandoned occurrence. But the system still has room for improvement. It will become inefficient (running at lower frame rate) since multiple abandoned objects existing simultaneously mean the system requires multiple selective tracking modules to locate each owner. High population density is another difficult issue for vision-based methods. Even humans cannot notice abandonment reliably. In this case, foreground-mask sampling method may fail and the systems need an object-recognition-based solution to detect the abandonment.

Acknowledgments

The authors would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Contract no. NSC 97-2221-E-002-173-MY3. Ted Knoy is appreciated for his editorial assistance.

References

[1] Y.-L. Tian, R. Feris, and A. Hampapur, "Real-time detection of abandoned and removed objects in complex environments,"

in *Proceedings of the IEEE International Workshop on Visual Surveillance in Conjunction with European Conference on Computer Vision (ECCV '08)*, 2008.

[2] N. Bird, S. Atev, N. Caramelli, R. Martin, O. Masoud, and N. Papainkolopoulos, "Real time, online detection of abandoned objects in public areas," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '06)*, pp. 3775–3780, May 2006.

[3] S. Ferrando, G. Gera, and C. Regazzoni, "Classification of unattended and stolen objects in video-surveillance system," in *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance (AVSS '06)*, p. 21, IEEE Computer Society, Washington, DC, USA, 2006.

[4] M. Beynon, D. Van Hook, M. Seibert, A. Peacock, and D. Dudgeon, "Detecting abandoned packages in a multi-camera video surveillance system," in *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 221–228, July 2003.

[5] F. Lv, X. Song, B. Wu, V. Singh, and R. Nevatia, "Left-luggage detection using bayesian inference," in *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS '06)*, June 2006.

[6] J. Martinez-del Rincon, J. E. Herrero-Jaraba, J. R. Gomez, and C. Orrite-Urunuela, "Automatic left luggage detection and tracking using multi-camera ukf," in *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS '06)*, pp. 59–66, June 2006.

[7] L. Li, R. Luo, R. Ma, W. Huang, and K. Leman, "Evaluation of an ivs system for abandoned object detection on pets 2006 datasets," in *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS '06)*, pp. 91–98, June 2006.

[8] J. Zhou and J. Hoang, "Real time robust human detection and tracking system," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, p. 149, June 2005.

[9] B. Wu and R. Nevatia, "Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 1, pp. 90–97, IEEE Computer Society, Washington, DC, USA, 2005.

[10] B. Wu and R. Nevatia, "Tracking of multiple, partially occluded humans based on static body part detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 1, pp. 951–958, Washington, DC, USA, 2006.

[11] E. Auvinet, E. Grossmann, C. Rougier, M. Dahmane, and J. Meunier, "Left-luggage detection using homographies and simple heuristics," in *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS '06)*, pp. 51–58, June 2006.

[12] S. Guler, J. A. Silverstein, and I. H. Pushee, "Stationary objects in multiple object tracking," in *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance (AVSS '07)*, pp. 248–253, September 2007.

[13] K. Smith, P. Quelhas, and D. Gatica-Perez, "Detecting abandoned luggage items in a public space," in *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS '06)*, pp. 75–82, June 2006.

[14] "Pets 2006 dataset," http://www.cvg.cs.reading.ac.uk/PETS-2006/data.html.

[15] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Transactions on Image Processing*, vol. 13, no. 11, pp. 1459–1472, 2004.

[16] S. Nadimi and B. Bhanu, "Physical models for moving shadow and object detection in video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1079–1087, 2004.

[17] C. N. R. Kumar and A. Bindu, "An efficient skin illumination compensation model for efficient face detection," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 3444–3449, November 2006.

[18] D. Chai and K. N. Ngan, "Face segmentation using skin-color map in videophone applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551–564, 1999.

[19] "Axis communications," http://www.axis.com/.

[20] "Arecont vision," http://www.arecontvision.com/.

[21] V. Hough and C. Paul, "Method and means for recognizing complex patterns," Patent no. 3 069 654, December 1962.

[22] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.

[23] G. Hetzel, B. Leibe, P. Levi, and B. Schiele, "3d object recognition from range images using local feature histograms," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 2, pp. 394–399, 2001.

[24] "i-LIDS Dataset for AVSS 2007," http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html.

[25] F. Porikli, Y. Ivanov, and T. Haga, "Robust abandoned object detection using dual foregrounds," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 197875, 11 pages, 2008.