# Toshiba at TRECVID 2009: Surveillance Event Detection Task

Kentaro Yokoi, Tomoki Watanabe, and Satoshi Ito
Corporate Research and Development Center, TOSHIBA Corporation,
1, Komukai-Toshiba-Cho, Saiwai-Ku, Kawasaki, 212–8582, Japan
E-mail: {kentaro.yokoi, tomoki8.watanabe, satoshi13.ito}@toshiba.co.jp

## Abstract

*In this paper, we describe the Toshiba event detection system for TRECVID surveillance event detection task [1] that detects three TRECVID required events (E05:PersonRuns, E19:ElevatorNoEntry, and E20:OpposingFlow). Our system ("Toshiba_1 p-cohog_1") consists of four components: (1) robust change detection based on the combination of pixel intensity histogram, PTESC (Peripheral TErnary Sign Correlation), and PrBPRRC (Probabilistic Bi-polar Radial Reach Correlation) that is robust against illumination changes and background movements, (2) human detection using the CoHOG (Co-occurrence Histograms of Oriented Gradients) feature that outperforms the one using the HOG (Histogram of Oriented Gradient) feature, (3) human tracking using linear estimation and color histogram matching, and (4) event detection based on change detection and human tracking. We briefly describe the four components.*

## 1 Introduction

We developed a basic event detection system for TRECVID surveillance task [1] that detects three TRECVID required events (E05:PersonRuns, E19:ElevatorNoEntry, and E20:OpposingFlow). Our system consists of four components: (1) change detection, (2) human detection, (3) human tracking, and (4) event detection (Fig.1).

First, change detector detects change region from the input image. We adopted change detection that is robust against illumination changes and background movements because TRECVID data include small illumination changes and foreground invasions in the training images such as walking people. Our change detection is the combination of pixel intensity histogram [2], PTESC (Peripheral TErnary Sign Correlation) [3], and PrBPRRC (Probabilistic Bi-polar Radial Reach Correlation) [4].

Next, human detector detects human from the change region. We adopted CoHOG (Co-occurrence Histograms of Oriented Gradients)-based human detection [5]. CoHOG is a high-dimensional feature that extends the HOG (Histogram of Oriented Gradient) feature and CoHOG-based human detection outperforms the one using HOG.

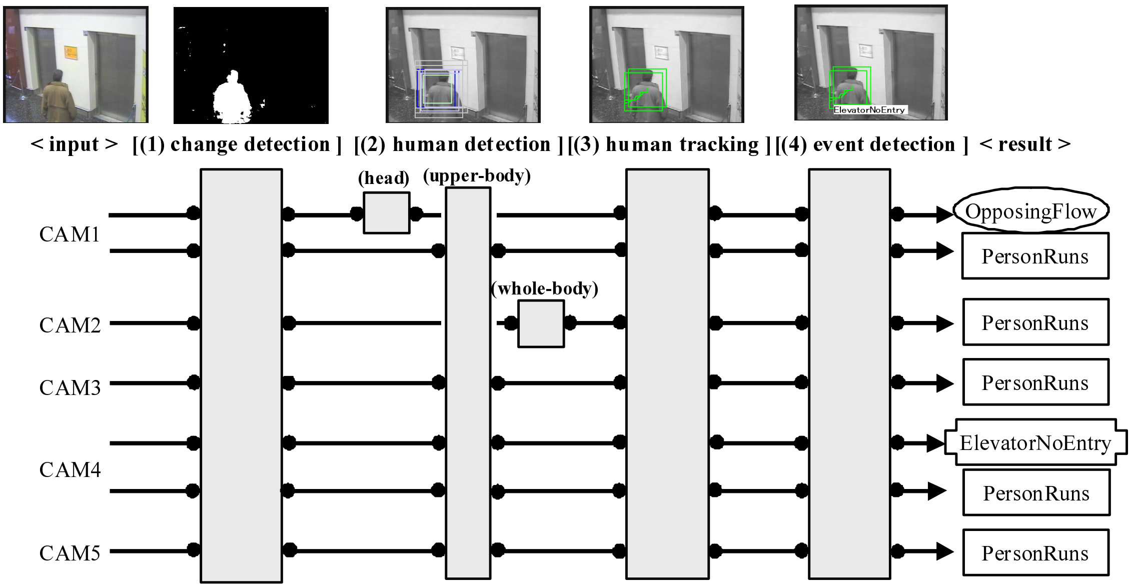


Figure 1: Process flow of our surveillance system

Then, human tracker tracks the detected humans using linear estimation of human position and color histogram matching.

Finally, event detector recognizes events according to the results of change detection and human tracking.

In the following sections, we explain each of the four components.

## 2 Change Detection

We adopted change detection that is robust against illumination changes and background movements. It combines three change detection methods: pixel intensity histogram [2] using color information that is robust against background movements, PTESC (Peripheral TErnary Sign Correlation) [3] using local texture that is robust against illumination changes, and PrBPRRC (Probabilistic Bi-polar Radial Reach Correlation) [4] using local/global texture that is robust against illumination changes and background movements. We explain several change detection methods including ours and show some results of the application of our change detection to TRECVID data.

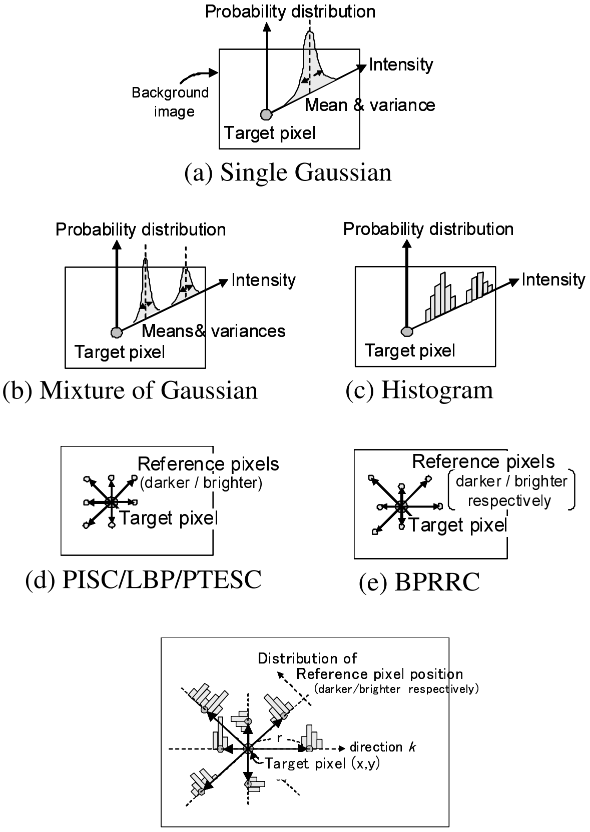
### 2.1 Change Detection Method

Many change detection methods have been proposed. Generally, the systems calculate the probability distribution of the input pattern from training images with the background model, and then detect changes from the test image according to the posterior probability. Fig.2 and Tab.1 show the schematics of the background models and a comparison of them, respectively.

One of the simplest background models is the single Gaussian model that models each pixel intensity with a single Gaussian distribution (Fig.2(a)). The Gaussian distribution can model intensity fluctuation of each pixel caused by sensing devices but the model is too simple to model real environmental changes such as illumination changes and background movements. MoG (Mixture of Gaussian) [6] uses multiple Gaussian distributions to model multiple background intensity distributions caused by tree swaying and door movement (Fig.2(b)). MoG is used in many applications but requires a decision on the number of Gaussian distributions. The non-parametric pixel intensity model with pixel intensity histogram [2] (Fig.2(c)) uses histogram that can model arbitrary intensity distributions and is free from the decision on the number of Gaussian distributions.

The pixel-intensity-based models mentioned above are not robust against illumination changes (Tab.1(a)-(c)) because illumination changes cause large intensity changes deviating from the past intensity history. For example, background models trained from images in the sun cannot cover inputs in the shade.

To increase robustness against illumination changes, some methods introduced texture information. Texture information based on the intensity differences among local pixels is stable against illumination changes because all the local pixels change their intensities by almost the same



(a) Single Gaussian

(b) Mixture of Gaussian

(c) Histogram

(d) PISC/LBP/PTESC

(e) BPRRC

(f) PrBPRRC

Figure 2: Schematics of the background models for change detection

Table 1: Comparison of background models for change detection

| background model | | robust against illumination changes | robust against background movements |
|---|---|---|---|
| pixel intensity based | (a) Single Gaussian (average background) | × | × |
| | (b) Mixture of Gaussian (MoG, GMM) | × | ○ |
| | (c) Pixel-intensity Histogram | × | ○ |
| texture based | (d) PISC/LBP/PTESC | ○ | × |
| | (e) BPRRC | ○ | × |
| | (f) PrBPRRC | ○ | ○ |

amount and the intensity differences among them don't change. PISC (Peripheral Increment Sign Correlation) [7], LBP (Local Binary Pattern) [8], and PTESC (Peripheral TErnary Sign Correlation) [3] model background using local texture information (Fig.2(d)) and BPRRC (Bi-polar Radial Reach Correlation) [9] models it using local/global texture information (Fig.2(e)). These texture-based methods are robust against illumination changes but not robust against background movements because of the static texture model (Tab.1(d)(e)). To solve this problem, PrBPRRC, which introduces non-parametric histogram model into BPRRC (Fig.2(f)), has been proposed [4]. PrBPRRC is robust against both illumination changes and background movements (Tab.1(f)) because it models dynamic texture distribution caused by background movements such as tree swaying and walking people in the training images by histogram model.

We adopted pixel intensity histogram using color information, PTESC, and PrBPRRC, and combined them to make the detection more robust. Robust change detection can reduce false positives of human detection as well as reduce the search area for human detection [10] (Fig.3).

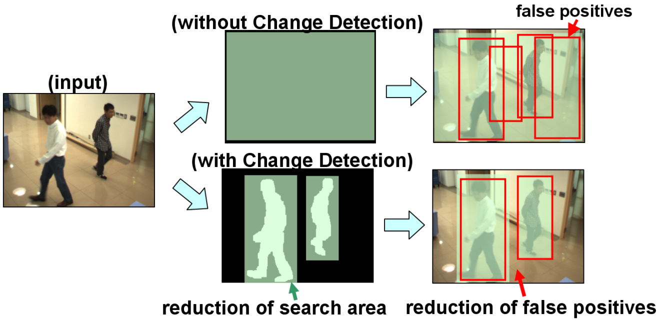


Figure 3: Schematics of the effectiveness of change detection for human detection. Change detection reduces search area and false positives of human detection

## 2.2 Change Detection Result

Fig.4 shows some results of change detection. In Fig.4, (a) shows an input image and (b) shows the result of change detection, the combination of all the components (c)-(e). Three components (c)-(e) compensate one another and make the result (b) more stable under illumination changes and background movements.

# 3 Human Detection

## 3.1 CoHOG Human Detection

Our human detector detects humans from the change region detected by the change detection described in Sec.2. We adopted a human detector based on CoHOG (Co-occurrence Histograms of Oriented Gradients) feature descriptor [5] and Support Vector Machine (SVM).

The CoHOG feature extends HOG (Histogram of Oriented Gradient) [11] by considering the co-occurrence of

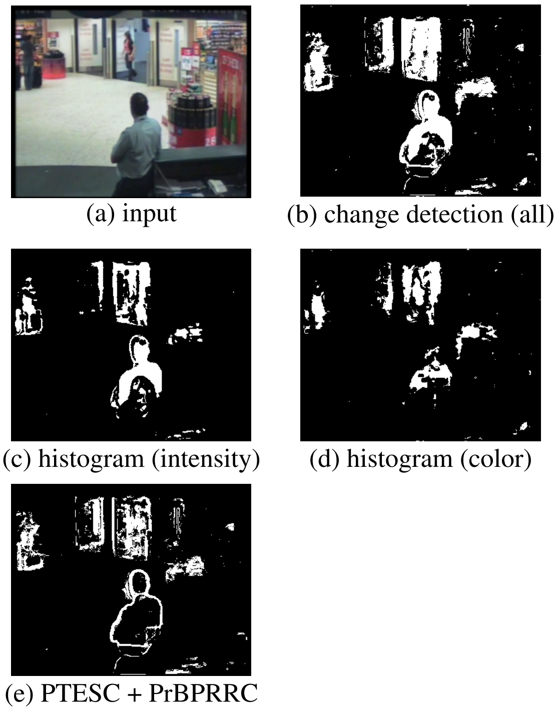

(a) input
(b) change detection (all)
(c) histogram (intensity)
(d) histogram (color)
(e) PTESC + PrBPRRC

Figure 4: Typical results of change detection

oriented gradients and forms higher-dimensional feature vector than HOG. CoHOG inherits the robustness against deformation and illumination changes from HOG and extends its description power to describe the complex shape of human in detail.

Fig.5 shows the process flow of CoHOG human detection. The system calculates oriented gradients in an input image (Fig.5(a)) and then makes histograms of the co-occurrence of the oriented gradients (Fig.5(b)). By considering co-occurrences between the oriented gradients at various offsets, CoHOG has an extensive vocabulary and description power. Then the co-occurrence histograms are concatenated into one vector (Fig.5(c)) and a Support Vector Machine (SVM) classifies the vector into human or non-human (Fig.5(e)).

We use a linear SVM for classifier because the CoHOG feature is so powerful that non-linear SVM requiring much computation is unnecessary. Though the dimension of CoHOG is tens of thousands, the simplicity of CoHOG calculation and the speed of linear SVM realize real-time detection. The performance of the human detection using CoHOG outperforms the one using HOG and is better than or at least comparable to other state-of-the-art methods [5]. Fig.6 shows the comparison of the performance on the DaimlerChrysler Pedestrian Classification Benchmark Dataset [12] and the INRIA Person dataset [11]. Upper left plot on ROC curve in Fig.6(a) and lower left plot on DET curve in Fig.6(b) indicate better performance. Our CoHOG detector with red plot shows performance better than or at least comparable to that of other state-of-the-art methods (refer to [5] for detail).

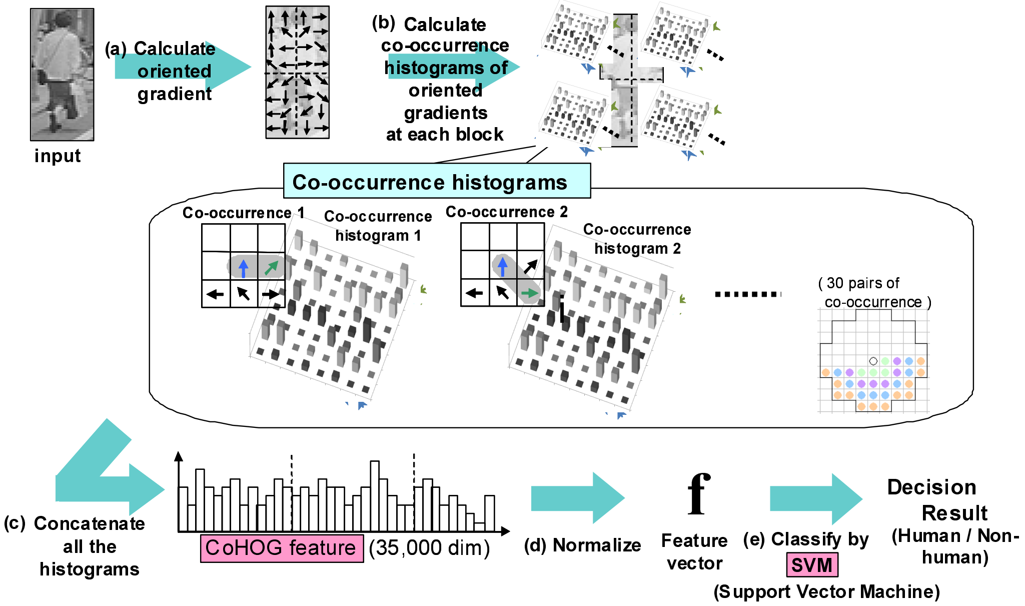
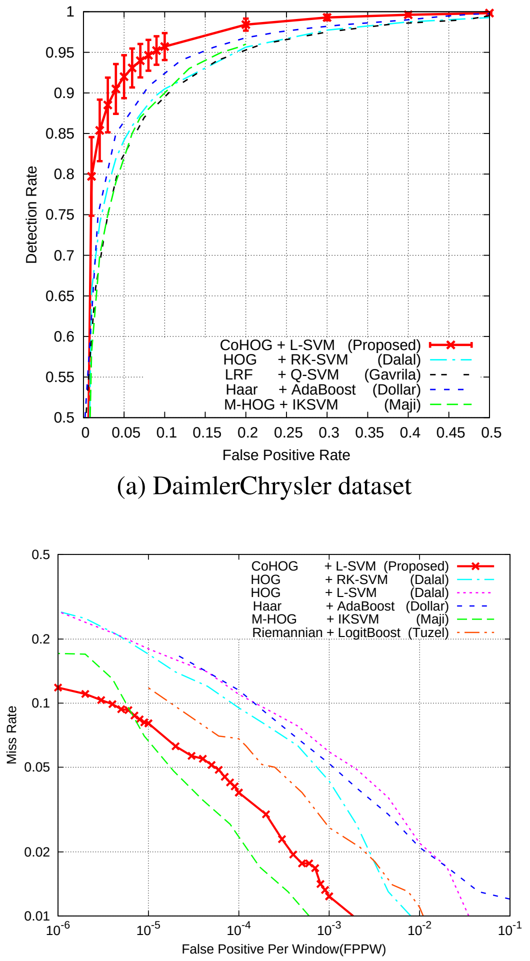

input
(a) Calculate oriented gradient
(b) Calculate co-occurrence histograms of oriented gradients at each block
Co-occurrence histograms
Co-occurrence 1
Co-occurrence histogram 1
Co-occurrence 2
Co-occurrence histogram 2
( 30 pairs of co-occurrence )
(c) Concatenate all the histograms
CoHOG feature (35,000 dim)
(d) Normalize
f
Feature vector
(e) Classify by SVM
(Support Vector Machine)
Decision Result (Human / Non-human)

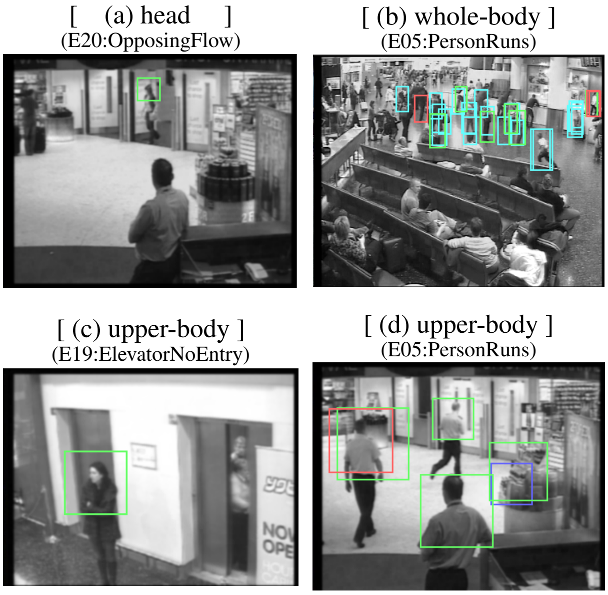
Figure 5: Process flow of CoHOG human detection



(a) DaimlerChrysler dataset

(b) INRIA datset

Figure 6: Comparison of human detection performance


[ (a) head ]
(E20:OpposingFlow)

[ (b) whole-body ]
(E05:PersonRuns)

[ (c) upper-body ]
(E19:ElevatorNoEntry)

[ (d) upper-body ]
(E05:PersonRuns)

Figure 7: Sample results of human detection

## 3.2 Human Detection Result

We trained head, upper-body, and whole-body human detectors and used the proper one for each camera data (Fig.1(2)).

Fig.7 shows some results of human detection. Head detector is used for E20:OpposingFlow detection of CAM1 because the people exiting from the gate are sometimes heavily occluded by other people and only the head region can be seen (Fig.7(a)). Whole-body detector is used for E05:PersonRuns detection of CAM2 because the people running in the CAM2 movie are very small and the whole-body can usually be seen (Fig.7(b)). Upper-body detector is used for other events (E19:ElevatorNoEntry of CAM4 and E05:PersonRuns of CAM1, CAM3, CAM4, and CAM5) because the lower half of the body sometimes cannot be seen (Fig.7(c)(d)).

# 4 Human Tracking

We adopted a human tracker using linear estimation of human position and color histogram matching. The tracker maintains the history of trajectories of each tracked human and estimates the human position at the current frame from the history. Then, it finds the correspondence between the estimated human and the detected human based on the similarity of the color histogram between them.

## 4.1 Linear Estimation of Human Position

Human position is estimated with a linear model using the previous position, velocity, and acceleration. Let the human position, velocity, and acceleration at frame $t$ be $p_t$, $v_t$, and $a_t$, respectively. The estimation of the human position at frame $t+1$, $\tilde{p}_{t+1}$, is calculated as

$$\tilde{p}_{t+1} = p_t + v_t.$$

If the human position at frame $t+1$, $p_{t+1}$, is confirmed by the correspondence matching described below, the system updates $v_{t+1}$ and $a_{t+1}$ as

$$v_{t+1} = w_v * \hat{v}_{t+1} + (1 - w_v) * v_t$$

and

$$a_{t+1} = w_a * \hat{a}_{t+1} + (1 - w_a) * a_t,$$

where $\hat{v}_{t+1} = p_{t+1} - p_t$, $\hat{a}_{t+1} = v_{t+1} - v_t$, and $w_v$ and $w_a$ are the update weights for the newest velocity and acceleration respectively. Larger weights make the system follow the change of velocity and acceleration quickly but be sensitive to the detection errors.

## 4.2 Correspondence Matching

The correspondence between the estimated human and the detected human is evaluated from four measures: (1) the ratio of region overlapped, (2) the ratio of change region, (3) detection score, and (4) color histogram similarity. The correspondence measure is given by

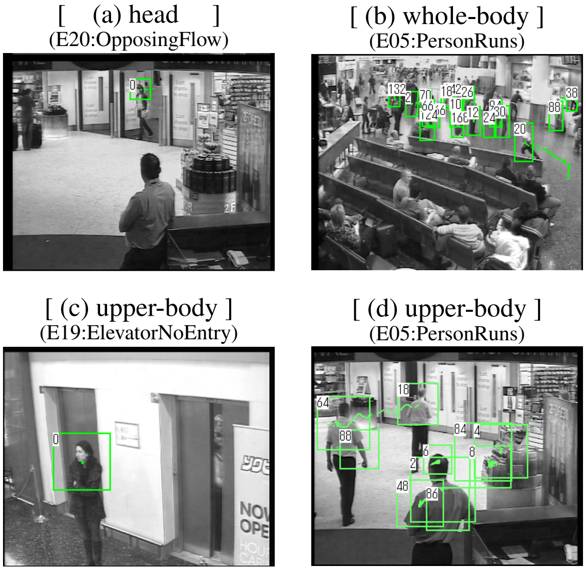
$$\begin{aligned} & M(estimation, detection) \\ &= w_{region} * M_{region} + w_{change} * M_{change} \\ &\quad + w_{score} * M_{score} + w_{color} * M_{color}, \qquad (1) \end{aligned}$$

where $M_{region}$ is the ratio of the overlapped region between the estimated human and the detected human, $M_{change}$ is the ratio of the change region detected by change detection in Sec.2 in the human region, $M_{score}$ is the score of human detection, $M_{color}$ is a color histogram similarity between the estimated human and the detected human based on color histogram intersection [13], and the coefficients $w_{region}$, $w_{change}$, $w_{score}$, and $w_{color}$ are the weights for the four measures, respectively. After the correspondence matching, the tracker merges other detected humans similar to the matched human based on the above measure and the remainders are added as new human trajectories.

Kalman filter and particle filter will realize a more stable tracker than the linear estimation tracker. We intend to introduce them in future work.

## 4.3 Human Tracking Result

Fig.8 shows some results of human tracking. Rectangles indicate human detection results and following line segments indicate human tracking results. Fig.7(a) shows head tracking for E20:OpposingFlow detection of CAM1; Fig.7(b) shows whole-body tracking for E05:PersonRuns detection of CAM2; Fig.7(c)(d) show upper-body tracking for E19:ElevatorNoEntry of CAM4 and E05:PersonRuns of CAM1.



[ (a) head ]
(E20:OpposingFlow)

[ (b) whole-body ]
(E05:PersonRuns)

[ (c) upper-body ]
(E19:ElevatorNoEntry)

[ (d) upper-body ]
(E05:PersonRuns)

Figure 8: Sample results of human tracking

# 5 Event Detection

Our event detector detects three required events: (1)E05:PersonRuns, (2)E19:ElevatorNoEntry, and (3)E20:OpposingFlow. In the following subsections, we explain these three event detections.

### 5.1 E05:PersonRuns

Our event detector detects the event E05:PersonRuns based on the velocity of the tracked human. It maintains the average $\mu$ and standard deviation $\sigma$ of the velocity in eight directions at each segmented surveillance area as shown in Fig.9. If the velocity of the tracked human exceeds $\mu + 2.0\sigma$ continuously, the event detector recognizes it as the event E05:PersonRuns.

Though the parameters $\mu$ and $\sigma$ should be learned from the tracked humans in the training data, we used approximate values manually given because of the lack of training time. Since we only checked the part of the training data for the parameter setting, use of all the training data is expected to greatly improve the performance of the event detector.

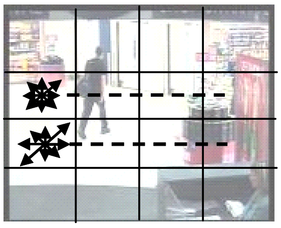


Figure 9: Statistics of human velocity and direction

### 5.2 E19:ElevatorNoEntry

Our event detector detects the event E19:ElevatorNoEntry based on the sequence of the change in the elevator door area and the sequence of human detection. It detects the door is closed by the disappearance of the change in the elevator door area and then detects the event E19:ElevatorNoEntry if a human is detected during the closing of the door. The elevator door area is manually given and the human detector is trained using upper half of the body because only upper half of the body can be seen in CAM4 elevator data.

### 5.3 E20:OpposingFlow

Our event detector detects the event E20:OpposingFlow based on the flow direction of the tracked human. It maintains the occurrence probability of the flow in eight directions at each segmented surveillance area as shown in Fig.9. The ordinariness of the flow direction of the tracked human is given by

$$Ord(flow) \quad = \quad \arg \max_{dir=1..8} w_{dir} * cos(flow, dir), \quad (2)$$

where $w_{dir}$ is the occurrence probability of the flow in the direction of $dir$. If the ordinariness is less than a threshold, the event detector recognizes it as the event E20:OpposingFlow.

Though the parameter $w_{dir}$ should be learned from the tracked humans in the training data, we used approximate values manually given because of the lack of training time. Since we only checked the part of the training data for the parameter setting, use of all the training data is expected to greatly improve the performance of the event detector.

# 6 Conclusion

In this paper, we explained our implementation of an event detection system for TRECVID surveillance task. Our system consists of four components: change detection, human detection, human tracking, and event detection.

First, our change detector detects change region from the input image. It combines pixel intensity histogram using color information, PTESC, and PrBPRRC; therefore, it is robust against illumination changes and background movements such as walking people. Next, our human detector detects humans from the change region. It introduces our new powerful CoHOG feature that outperforms the HOG feature. Then, our human tracker tracks the detected humans using linear estimation of human position and color histogram matching. It estimates the next position of the tracked human and compares it with newly detected humans. The correspondence is evaluated based on the ratio of region overlapped, the ratio of change region, detection score, and color histogram similarity. Finally, our event detector detects three required events with the results of change detection and human tracking. The events E05:PersonRuns and E20:OpposingFlow are detected based on the velocity and direction of the human flow. The event E19:ElevatorNoEntry is detected based on the disappearance of changes in the elevator door area and the human detection in the waiting area.

The system requires many parameters for the decisions and they have to be learned from training data. At present, we are using the parameters manually given because of the lack of training time. In future work, we intend to improve the system performance by learning the parameters from the TRECVID training data.

# References

[1] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[2] Hiroaki Nakai. Non-Parameterized Bayes Decision Method for Moving Object Detection. In *Proceedings of 2nd Asian Conference on Computer Vision*, pages III–447–451, 1995.

[3] Kentaro Yokoi. Illumination-robust change detection using texture based features. In *IAPR Conference on Machine Vision Applications (MVA2007)*, number 13-14, pages 487–491, May 2007.

[4] Kentaro Yokoi. Probabilistic BPRRC: Robust Change Detection against Illumination Changes and Background Movements. In *IAPR Conference on Machine Vision Applications (MVA2009)*, number 5-1, pages 148–151, May 2009.

[5] Tomoki Watanabe, Satoshi Ito, and Kentaro Yokoi. Co-occurrence Histograms of Oriented Gradients for

Pedestrian Detection. In *Proceedings of the 3rd Pacific-Rim Symposium on Image and Video Technology (PSIVT2009)*, pages 37–47, January 2009.

[6] Chris Stauffer and W.E.L Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1999)*, volume 2, pages 246–252, June 1999.

[7] Yutaka Satoh, Shun'ichi Kaneko, and Satoru Igarashi. Robust Object Detection and Segmentation by Peripheral Increment Sign Correlation Image. volume 35, pages 70–80, June 2004.

[8] Marko Heikkilä and Matti Pietikäinen. A Texture-Based Method for Modeling the Background and Detecting Moving Objects. volume 28, pages 657–662, April 2006.

[9] Yutaka Satoh and Katsuhiko Sakaue. Robust Background Subtraction based on Bi-polar Radial Reach Correlation. In *Proceedings of the IEEE International Conference on Computers, Communications, Control and Power Engineering (TENCON05)*, pages 998–1003, November 2005.

[10] Kentaro Yokoi, Tomoki Watanabe, and Satoshi Ito. A Demonstration of Human Detection using Co-occurrence Histograms of Oriented Gradients Feature Descriptor. In *Proceedings of the 3rd Pacific-Rim Symposium on Image and Video Technology (PSIVT2009)*, pages D–II–2, January 2009.

[11] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, volume 2, pages 886–893, June 2005.

[12] S. Munder and D. M. Gavrila. An Experimental Study on Pedestrian Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, November 2006.

[13] Michael J. Swain and Dana H. Ballard. Indexing Via Color Histograms. In *Proceedings of the 3rd IEEE International Conference on Computer Vision (ICCV 1990)*, pages 390–393, December 1990.