REAL-TIME VIDEO INTELLIGENT SURVEILLANCE SYSTEM

Weidong Zhang, Feng Chen, Wenli Xu, Enwei Zhang

Department of Automation, Tsinghua University. Beijing, China. 100084

ABSTRACT

With the rapid development of hardware equipments, it is now economically and technically feasible to build a video surveillance system. This paper presents the system architecture of VISS, a Video Intelligent Surveillance System deployed in parking lots. In VISS we adopt robust moving object detecting and tracking algorithm, and we present a novel activity recognition framework based on Layer Hidden Semi-Markov Model (LHSMM) which is used for modeling activities. The experimental results on real-time video show that the system is effective and robust in complex activity recognition.

1. INTRODUCTION

Video surveillance is becoming increasingly important for those security-sensitive areas such as airports, banks, casinos, and parking lots. Many efforts have been made in this field, CMU's Video Surveillance and Monitoring (VSAM) project [1] and W4 real-time system [2] detect and track the human in scene and recognize simple activities such as walking and running. Recent systems [3][4] adopt HMMs to recognize activities, but neither of them can gain a good performance for complex activities.

In this paper we present a vision-based surveillance system framework for recognizing complex activities in parking lots. If the knowledge of the ground plane is available VISS can be easily retrained for other scenes. In our system we modify Codebook model [5] to detect moving objects in the scene, and use Kalman filter-based tracking algorithm to record trajectories for further analysis. Moreover we present a novel method named Layer Hidden Semi-Markov Model (LHSMM) to recognize complex activities such as *stealing car* in parking lots. Activities are modeled in the LHSMM in two ways: with HMMs, the bottom layer represents subactions such as *get_close_to_car, surround _car, stay_by_car, get_away_from_car* in stealing car; the top layer represents the complex activities and their states durations using HSMMs. We adjust atomic segment length in subactions by feedback information from the bottom layer's recognition results. Using this mechanism VISS can understand the activities with large variable-durations.

This work was supported by the key project of National Natural Science Foundation of China(project no.60432030)

The rest of the paper is organized as follows: In Section 2, the hardware and software architecture are provided. A detail description of the algorithms is presented in Section 3. Section 4 demonstrates the system results, and Section 5 contains our conclusions and future works.

2. SYSTEM ARCHITECTURE

VISS obtains video data of the scene using static cameras mounted at high places. The data from cameras is sampled and compressed by the NETMEs (network video server, our group has participated in its development) connected directly with the cameras. NETMEs push the MPEG4 stream into network. The store server and client, located in local or wide network, stores and processes the stream coming from network respectively. Figure 1 depicts a typical hardware architecture of VISS.

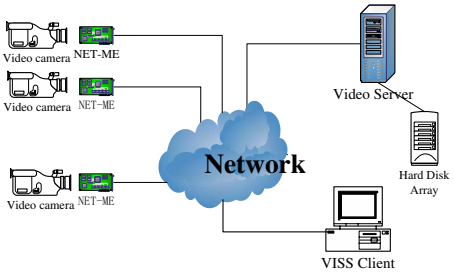

Fig. 1. The hardware architecture of VISS.

The whole software runs on the client. Firstly, it decodes the MPEG4 stream into frame-based image sequence. And then analyze the decoded image sequence to recognize activities. To recognize the activities, moving human detecting and tracking must be firstly executed to obtain necessary information such as trajectories. LHSMM is used to recognize the activities. Detail algorithms will be discussed in Section 3.

3. ALGORITHM COMPONENTS

3.1. Foreground Detection

The capability of segmenting foreground regions from background is crucial for visual surveillance, since its accuracy and robustness affect all the sequential processes. We modify Codebook (CB) model background subtraction algorithm proposed in [5] mainly based on the following four aspects:

- CB allows moving foreground objects in the scene during the initial training period, which is necessary in our real-time surveillance system.
- CB shows better performance in modeling fast variations in background, which often occurs when the person is surrounding the car.
- VISS works with low-bandwidth compressed videos because of the limited network bandwidth, CB can eliminate most compression block artifacts.
- CB is efficient in both memory and speed for real-time video surveillance system.

CB algorithm samples values for each pixel over long times, it builds a codebook consisting of one or more codewords. Samples at each pixel are clustered in the set of codewords based on a color distortion metric together with brightness [5]. In VISS, we build a background codebook and a codebook cache for each pixel and define more reasonable brightness range for long-time steady detecting.

Let $l = \{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_L\}$ represents the background codebook for the pixel consisting of L codewords, $\mathbf{c}_i, i = 1...L$ consists of an RGB vector $\mathbf{v}_i = (\bar{R}_i, \bar{G}_i, \bar{B}_i)$ and a 5-tuple $\mathbf{aux}_i = \langle \check{I}_i, \hat{I}_i, f_i, \lambda_i, q_i \rangle$ whose item means the min and max brightness of all pixels assigned to this codeword, codeword occurring frequency, the longest interval that the codeword has not recurred and the last access time respectively.

Given background codebook l and an incoming pixel $\mathbf{x}_t = (R, G, B)$, $I = \sqrt{R^2 + G^2 + B^2}$, CB finds the matched codeword based on two conditions:

$$colordist(\mathbf{x}_t, \mathbf{v}_m) \leq \varepsilon \tag{1}$$

$$brightness(I, \langle I_{low}, I_{hi} \rangle) = true \tag{2}$$

We calculate the color distortion by

$$colordist(\mathbf{x}_t, \mathbf{v}_m) = \sqrt{\|\mathbf{x}_t\|^2 - \langle \mathbf{x}_t, \mathbf{v}_m \rangle^2 / \|\mathbf{v}_m\|^2} \tag{3}$$

and define

$$I_{low} = \min\{\alpha \hat{I}, \check{I}\}, I_{hi} = \max\{\hat{I}, \check{I}/\beta\} \tag{4}$$

where $\alpha < 1$ and $\beta < 1$, which can handle the shadow and highlight instances.

We also build a codebook cache to store codewords added recently, and increase their f once a matched instance occurs. We add the codeword to the background codebook when f is larger than the predefined threshold.

3.2. Human Tracking

After foreground detection, we obtain separated regions. Because of noises, shadows, reflections and occlusions in image sequence, a person may be mis-split into several smaller close regions, some of them even merge with other unrelated regions. Since the later seems infrequent in parking lots, we handle the former instance by assuming that the human appearance model satisfies some constraint. We construct an appearance model H_{avg} which is adjusted when people walk towards or away from camera, and normalize $p(H_{avg}) = 1$. VISS merges close enough regions R_1 and R_2 based on the following condition:

$$p(R_1) < \varepsilon_1, p(R_2) < \varepsilon_1 \ and \ p(R_1 \cup R_2) > \varepsilon_2 \tag{5}$$

where ε_1,ε_2 are merging thresholds and

$$p(R) = p_H(h/h_{avg}) \cdot p_W(w/w_{avg}) \cdot p_S(\sum_{x \in R_f} 1/(h \cdot w)) \tag{6}$$

in which all the component probabilities follow Gaussian distribution, R_f represents the detected foreground region, h and w represent the height and width of the region. After regions merging, persons' position and shape features are recalculated. Tracking over time involves matching persons in consecutive frames using features such as intensity and shape template. Firstly persons' positions in next frame are predicted by:

$$\begin{cases} x_{predict}(n+1) = x(n) + v_x(n)\Delta t \\ y_{predict}(n+1) = y(n) + v_y(n)\Delta t \end{cases} \tag{7}$$

Here we assume that persons move at a reasonable speed, and persons in next frame definitely fall into a near large region of the predicted position. Then VISS searches the most matched region in the large region for each person using intensity template and shape template. We adopt intensity template correlation function defined in [1] and a shape template cost function defined as follows:

$$D_I(d) = \sum_{x \in R} \frac{W(i,j)|I_n(x) - I_{n+1}(x+d)|}{||W||} \tag{8}$$

$$D_S(d) = \sum_{x \in R_f} 1/h \times w \tag{9}$$

where $W(\cdot)$ represents the distance of current pixel with the central pixel. The best matched position is given by

$$\begin{aligned} &\hat{d} = \arg\min_d D_I(d) \\ &if\ D_I(\hat{d}) < T_I \ and \ T_{SL} < D_S(\hat{d}) < T_{SH} \end{aligned} \tag{10}$$

where T_I and T_{SL},T_{SH} are predefined thresholds. The new position is $p_{n+1} = p_n + \hat{d}$ and the new velocity estimation is given by $\hat{v}_{n+1} = d/\Delta t$, and then we use $\alpha - \beta$ filter to refine the values.

The regions merging mechanism eliminates the effect of one region matching with multi-objects as well as multi-regions matching with one object. If there is no match occurs, a new object model is constructed.

3.3. Event Recognition

Event recognition is the most complex and challenging task in video surveillance. Traditionally, event recognition focused on learning the temporal characteristics in sequence using dynamic models such as the hidden Markov model (HMM).

VISS aims to analyze the real-time video in which events corresponding to the same semantic content maybe differ greatly in appearance. While the HMM is a simple and efficient model for learning sequential data, its performance tends to degrade when the range of activities becomes more complex, or the activities exhibit long-time temporal dependency that is difficult to be deal with under the Markov assumption [6]. Although the hidden Semi-Markov model and Layer HMM [7] can solve the long-time case to some extent, they can not cover the case that the duration varies greatly. We propose a LHSMM framework to handle this problem: the bottom layer recognizes the atomic actions and adjusts the segment length by the ground plane knowledge and recognition results of bottom layer; the top layer takes the output of the bottom layer as observation sequence to recognize the complex activities.

When the object is being tracked, the system obtains the feature vector $\mathbf{x}_e = \{x, y, v_x, v_y, a_x, a_y\}$ representing the object's position, velocity and acceleration. To recognize the *stealing car* for example, interaction information of human and cars is essential. With the ground-truth about the cars, feature vector is transformed into polar coordinates using the cars' center as the origins as shown in Figure 2. Let $\mathbf{x}^i = (\rho^i, \theta^i, v^i_\rho, v^i_\theta), i = 1, ..., M$ be the interaction feature vector of human and the ith car of M cars in the scene. Considering the case that there is one car in the scene, $\mathbf{x} = (\rho, \theta, v_\rho, v_\theta)$ is the interaction feature vector and $\mathbf{x}^i_f = (\rho^i_f, \theta^i_f, v^i_{\rho f}, v^i_{\theta f}), i = 1, ..., N$ is reference feature vector where $v^i_{\rho f} \sim N(v^i_\rho, \sigma^i_{v_\rho})$ and $v^i_{\theta f} \sim N(v^i_\theta, \sigma^i_{v_\theta})$.

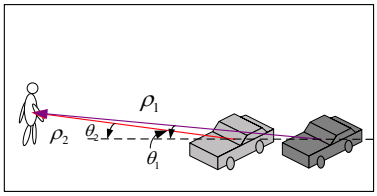

Fig. 2. Polar coordinates.

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{T_L}\}$ be the observation sequence of the bottom layer in which $\mathbf{x}_t$ represents t time feature and $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{T_H}\}$ be the observation sequence of top layer, which is also the output sequence of bottom layer.

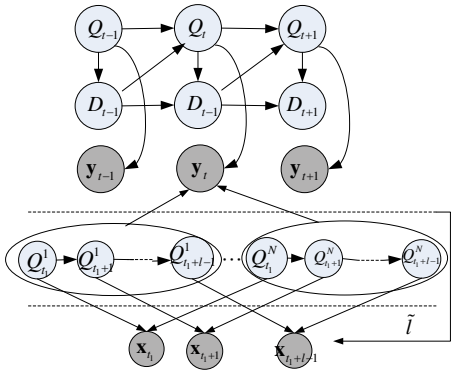

Fig. 3. Graphical representation of LHSMM.

At the bottom layer, VISS holds HMMs with model parameter set $\lambda^i_B = (\pi_i, \mathbf{A}_i, \mathbf{B}_i), i = 1, ..., N$ for subactions such as *get_close_to_car, surround_car, stay_at_car, get_away_from_car*. Firstly, we set up a buffer whose length l is initially chosen as short as possible. At time t_1, the conditional probability densities (CPDs) of segment vector sequence $\mathbf{x}_{t_1:t_1+l-1}$ in buffer are calculated as $P_i = P(\mathbf{x}_{t_1:t_1+l-1}|\lambda^i_B)$, and then we adjust the segment length by

$$\tilde{l} = f(\rho_0, i^*) \times l \times (\frac{v^{i^*}_{pf}}{v_p} + \frac{v^{i^*}_{\theta f}}{v_\theta}) \tag{11}$$

where $i^* = \arg\max_i P_i$, $f(\rho_0, i^*) = \begin{cases} 1 & i^* \neq 1, 3 \\ \rho_0/\rho_f & i^* = 1 \end{cases}$, ρ_0 is the value at time 0, while the length remain unchanged when $i^* = 3$. The adjusted vector sequence is $\mathbf{x}_{t_1:t_1+\tilde{l}-1}$ and the CPDs are recalculated: $\mathbf{y}_t = (\tilde{P}_1, \tilde{P}_2, ..., \tilde{P}_N)$.

The top layer utilizes HSMM to recognize the complex activities consisting of subactions; its observation sequence is the output sequence $\mathbf{Y}$ of the bottom layer. Let Q_t be the state of t and D_t be the remaining duration of state Q_t. The parameter set of the model contains the prior probabilities π_j, conditional observation probabilities $P(\mathbf{y}_{t-d+1:t}|q, d)$, transition probabilities $P(Q_t|Q_{t-1}, D_{t-1})$ and $P(D_t|D_{t-1}, Q_t)$.These parameters are described as follows:

$$\pi_j = P(Q_1 = j) \quad s.t. \sum_j \pi_j = 1 \tag{12}$$

$$P(Q_t = j|Q_{t-1} = i, D_{t-1} = d) = \begin{cases} \delta(i, j) & if\ d > 0 \\ A(i, j) & if\ d = 0 \end{cases} \tag{13}$$

$$P(D_t = d'|D_{t-1} = d, Q_t = i) = \begin{cases} p_i(d') & if\ d = 0 \\ \delta(d', d-1) & if\ d \neq 0 \end{cases} \tag{14}$$

$$P(\mathbf{y}_{t-d+1:t}|q, d) = \prod_{\tau=t-d+1}^{t} P(\mathbf{y}_\tau|Q_\tau = q) \tag{15}$$

where A is transition probabilities matrix, $\delta(\cdot)$ is the Dirac function, $P(Q_1 = j), P(\mathbf{y}_\tau|k), p_i(d')$ are trained probabilities. Given the model parameter λ^i_T , the observation probability is given by

$$P(\mathbf{y}_{1:T}, S|\lambda^i_T) = \pi_j p_j(d_1) P(\mathbf{y}_{1:d_1}|j, d_1) \prod_{n=2}^{N} p_{Q_{t_n+1}}(d_n)$$
$$P(Q_{t_n+1}|Q_{t_n}, d_n) P(\mathbf{y}_{t_n+1:t_n+d_{n+1}}|Q_{t_n+1}, d_{n+1})$$

where $S = (Q_{1:T}, D_{1:T})$is an sample,$t_n = \sum_{i=1}^{n-1} d_i, n = 2, ..., N$. The final label is selected by $c = \arg\max_i P(\mathbf{y}_{1:T}|\lambda^i_T)$.

4. EXPERIMENTAL RESULTS

VISS runs the software on a P4 2.8GHz computer at real-time (25fps) for 352×288 image sequence. We have deployed the

system in a parking lot at Tsinghua Campus. Five actors act as *stealing car* in different directions, velocities and regions. Figure 4 shows an example of VISS detecting and tracking a person in the scene. The red ball line is the trajectory tracked using the method described in Section 3.2. Since we only use Codebook model to eliminate the shadow, the dark shadow cast at noon can not be eliminated completely which will result in blobs and splits in one person. Using the region merging method presented in this paper, we can eliminate most of the mis-splited cases and obtain a smooth trajectory.

Fig. 4. The detection and tracking example.

We segment 53 *stealing car* (ST) samples from the video and segment 23 samples into subactions for training LHSMM. The segments and subactions durations vary as shown in table 1. We also select two segments those durations beyond 1300 frames. Figure 5 shows an example of the bottom layer recognition of the subactions. The x- coordinate value represents the time in which each is equal about 6 frames and the y- coordinate value is the probabilities normalized into 1.

Table 1. Durations of subactions and *stealing car* samples

	Subact.1	Subact.2	Subact.3	Subact.4	ST
Frame	147-350	135-250	80-150	125-300	550-1060

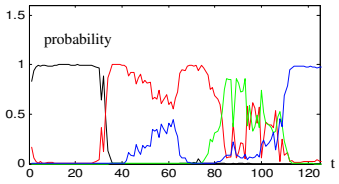

Fig. 5. Subactions recognition results.

Compared with HMMs, LHSMM can recognize activities with large variable-durations. Using LHSMM VISS obtains 90%-96% events recognition accuracy for the test samples with different threshold. While we recognize the same samples using HMMs after segmenting the samples into equal sub-segments, only 73%-76% accuracy is obtained.

5. CONCLUSIONS

This paper has described VISS system that records the trajectories and recognizes the activities of human in the scene. We have implemented VISS in a parking lot. VISS detects the moving human using Codebook model background subtraction method and tracks the human using Kalman filter-based tracking method. In the event recognition module we present a LHSMM framework to recognize the activities.The results demonstrate that our system properly segment moving objects from background and obtain a smooth trajectories; also the results confirm the effectiveness of the presented LHSMM activity recognition framework.

In this paper, we make the following contributions:

- We present a video intelligent surveillance system that can recognize complex activities in real-time video.
- In VISS, we modify the Codebook model for long-time steady detection. We present an effective region merging method to eliminate most of mis-slpit cases to obtain a smooth trajectory.
- We present an event recognition framework for complex activities with large variable-durations, which utilizes the nature of inherent hierarchical structure in activities and typical duration. We also eliminate the effect caused by the large-variation of duration by the feedback information of the bottom layer.

However, VISS can not recognize the activity sequence with overlapped boundaries between two consecutive activities which will result in errors in a long time. How to recognize complex activity sequence is our future research work.

6. REFERENCES

[1] R.T.Collins et al., "A system for video surveillance and monitoring: Vasm final report," Tech. Rep. CMU-RI-TR-00-12, Carnegie Mellon University, 2000.

[2] Haritaoglu I, Harwood D, and Davis L S, "W-4: Real-time surveillance of people and their activities," *IEEE Trans. PAMI*, vol. 22, no. 8, pp. 809–830, 2000.

[3] Raju Rangaswami et al., "The sfinx video surveillance system," *In Proc.ICME*, 2004.

[4] Vinod Nair and James J. Clark, "Automated visual surveillance using hidden markov models," *International Conference on Vision Interface*, pp. 88 – 93, 2002.

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

[6] T.V. Duong, H.H. Bui, D.Q. Phung, and S Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-markov model," *In Proc.CVPR*, pp. 838 – 845, 2005.

[7] N.Oliver, E. Horvitz, and A. Garg, "Layered representations for human activity recognition," *IEEE International Conference on Multimodal Interfaces*, pp. 3–8, 2002.