Real-Time Apartment Building Detection and Tracking with AdaBoost Procedure and Motion-Adjusted Tracker

Yi Hu, Dae Sik Jang, Jeong Ho Park, Seong Ik Cho, and Chang Woo Lee

ABSTRACT—In this letter, we propose a novel approach to detecting and tracking apartment buildings for the development of a video-based navigation system that provides augmented reality representation of guidance information on live video sequences. For this, we propose a building detector and tracker. The detector is based on the AdaBoost classifier followed by hierarchical clustering. The classifier uses modified Haar-like features as the primitives. The tracker is a motion-adjusted tracker based on pyramid implementation of the Lukas-Kanade tracker, which periodically confirms and consistently adjusts the tracking region. Experiments show that the proposed approach yields robust and reliable results and is far superior to conventional approaches.

Keywords—Video-based navigation systems, building detection and tracking, AdaBoost classifier, motion-adjusted tracker.

I. Introduction

A new concept of car navigation system has recently been proposed, which uses real-time video instead of the digital map to provide more realistic guidance information [1]. Recognition of buildings in video sequences acquired from a 4S-Van vehicle [2] or similar one is essential for replacing points of interest in a digital map with an augmented reality representation. The names of buildings are superimposed on building images, as shown in Fig. 1, where real-time video is acquired by a forward looking camera mounted on the vehicle. Many studies have focused on the recognition of buildings by applying linear or corner features [3], [4]. However, those are inadequate for detecting and consistently tracking buildings in real-time video sequences. In this letter, we propose a novel approach for detecting and tracking apartments as important landmark information. We propose a robust algorithm for the detection of apartments based on the AdaBoost procedure [5], [6] followed by hierarchical clustering. Then, we propose a motion-adjusted tracker based on pyramid implementation of the Lukas-Kanade (LK) tracker [7]. The proposed method is described in section II. Experimental results and conclusions are given in sections III and IV, respectively.

II. Apartment Building Detection and Tracking

The proposed approach consists of two main modules:

Fig. 1. Representation concept of landmark information.

Manuscript received Aug. 22, 2007; revised Jan. 3, 2008.

This work was supported by the IT R&D program of MIC/IITA, Rep. of Korea [2005-S114-02, Technology Development for Construction and Management of Tangible Content for Telematics Services].

Yi Hu (email: yihu@dali.korea.ac.kr) was with the Department of Computer Information Engineering, Kunsan National University, Gunsan, Rep. of Korea and is with the School of Electrical Engineering, Korea University, Seoul, Rep. of Korea

Dae Sik Jang (email: dsjang@kunsan.ac.kr) and Chang Woo Lee (phone: + 82 63 469 4861, email: leecw@kunsan.ac.kr) are with the Department of Computer Information Engineering, Kunsan National University, Gunsan, Rep. of Korea.

Jeong Ho Park (email: parkjh@etri.re.kr) and Seong Ik Cho (email: chosi@etri.re.kr) are with the IT Convergence Technology Research Laboratory, ETRI, Daejeon, Rep. of Korea.

building detection and motion-adjusted tracking. Once apartment buildings are detected successfully, the detection process is suspended and only the tracking process is activated.

1. Apartment Building Detection

For the detection of apartment buildings, we select a primitive pattern from the analysis of its appearance. Then, we train the strong classifiers as in (1), a weighted combination of several weak classifiers using the AdaBoost algorithm given in [5] as

$$H(x)=\begin{cases}1 & \sum_{t=1}^{T}\alpha_t h_t(x)\geq \frac{1}{2}\sum_{t=1}^{T}\alpha_t, \\ 0 & \text{otherwise,}\end{cases} \qquad (1)$$

where $h(x)$ denotes a weak classifier, and α_t denotes the weight of the corresponding weak classifier. A strong classifier is composed of several weak classifiers, each of which is trained to detect one specific primitive. Finally, we construct a cascade of the strong classifiers to detect patterns representing apartments. To train our detector, the Haar-like features shown in Fig. 2 are used. We added four new types of features as shown in Figs. 2(h)-(k). The tilted features proposed in [6] are not considered because the features of the apartments are almost upright.

Generally, tens to hundreds of parts are detected on an image containing apartments. Based on spatial locality of the detected patterns, an agglomerative hierarchical clustering is adopted to combine the parts. Initial clusters are single parts. The clustering method iteratively merges the nearest clusters to form larger ones until the distance between them exceeds a threshold which is empirically set to be the average size of the parts. The final clusters are considered as candidate apartments. To separate apartment clusters from others, we use the number of parts in a cluster, which was set to 6 during experiments.

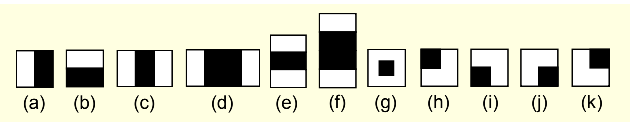


Fig. 2. Prototype simple features: (a)-(g) features proposed in [4] and [5] and (h)-(k) newly added features.

2. Motion-Adjusted Tracking

In an apartment region, there are many clear-cut corners, which are relatively well distributed. Therefore, the pyramid implementation of the LK tracker [7] is applied to track the detected regions along with those corner points. The initial points are detected using the corner detector [7] within the apartment regions. During tracking, some false feature points, such as trees and traffic signs, can be extracted as well as true ones. Usually, these false feature points move faster than true ones, so the tracker may track these obstacles instead of true ones. To solve the false tracking problem, we estimate all the motion vectors with respect to all feature points in the region. We denote the motion vectors as $V=\{v_1, v_2,\cdots, v_k\}$, where k is the number of feature points. Then, false feature points are removed using the following equation:

$$\text{If } \left|v_i - v_\mu\right| > T_v \text{, then reject point } i,\ i=1,2,\cdots,k, \qquad (2)$$

where v_i is the motion vector of the i-th point, v_μ is the mean motion vector in a region, and T_v is a given threshold, which is determined empirically. We calculate mean motion vectors in the half region of each direction and use them to adjust the shifting of the boundaries. When the apartments move out of view, the detection process is activated again. The detection process is periodically activated for more reliable tracking and adjustment of the tracking region.

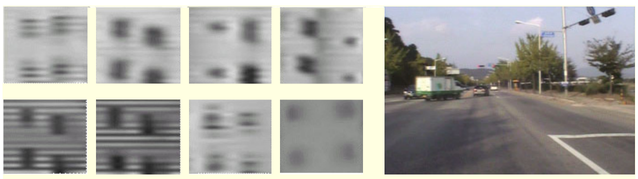
III. Experimental Results

To evaluate our proposed approach, we collected real-world test video sequences at 13 to 24 fps with 640×480 image resolution with vehicle speeds ranging from 60 to 80 km/h in Daejeon, Korea. The test platform was Windows XP running on a PC with Pentium IV, 3 GHz CPU, and 1 GB of memory.

To train the strong classifier, 2,051 regions were manually collected as positive samples and 6,355 images were collected as the negatives. Some training samples are shown in Fig. 3.

We selected seven video clips and captured 10,441 frames. Each clip starts before the appearance of apartments and stops when the vehicle has passed the apartments. The clips are divided into two categories: not-occluded, in which apartments are not occluded or slightly occluded, and occluded, in which apartments are partially occluded.

We trained two detectors using two different feature sets. The



(a) 8 positive samples (b) Negative sample

Fig. 3. Examples of training dataset.

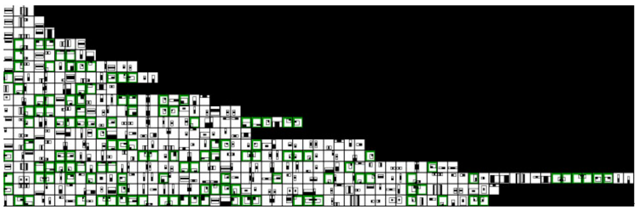


Fig. 4. Trained features used in the detector D2.


(a) (b)

Fig. 5. Results from two detectors: (a) D1 and (b) D2.

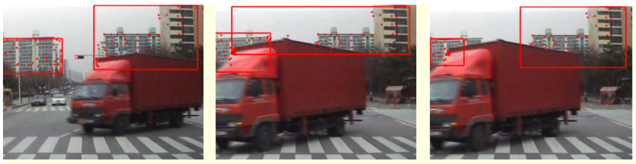

(a) (b) (c)

Fig. 6. Comparison of tracking with and without motion adjustment: (a) result in the *t*-th frame, (b) result in the (*t*+30)th frame without motion adjustment, and (c) result in the (*t*+30)th frame with motion adjustment.

Table 1. Performance analysis for the proposed approaches (%).

Video clip	Description	D1	D2	D3
Rec1_1	NOC	88.2	91.3	100
Rec1_2	NOC	89.8	97.6	100
Rec2_1	OC	80.4	93.4	98.6
Rec2_2	OC	71.3	89.5	100
Rec2_3	OC	76.1	86.1	100
Rec3_1	OC	85.2	87.1	100
Rec3_2	OC	72.1	84.4	99.9
Average		80.4	89.9	99.8

first detector, D1, used features shown in Figs. 2(a)-(g). The second detector, D2, used all the features including newly added ones. During the training, the final detection rate was 0.91 and the false alarm rate was 1.0E-6 in the training dataset.

Figure 4 shows the trained features selected by detector D2. The newly added features are marked using green rectangles. The detector consists of 18 stages and 435 features, of which 129 features are our newly added features. Detector D1 consists of 21 stages and 596 features.

A comparison of two detectors is shown in Fig. 5. Bar-like objects such as road sign poles are incorrectly detected as part of a building because D1 considers edge- and bar-like features only, while D2 benefits from corner-like ones adopted in our approach. Although D2 is relatively robust to occlusion, it may fail when there is severe occlusion caused by lush trees or bad view angles in which few patterns of apartment is observed.

Accuracy is measured by (3), where N_{FP} and N_{FN} are the number of false positives and negatives respectively, and N_{Total} is the total number of apartments in all frames:

$$\text{Accuracy (\%)} = \left(1 - \frac{N_{FP} + N_{FN}}{N_{Total}}\right) \times 100 . \quad (3)$$

Table 1 shows the experimental results. The D1 and D2 columns represent the accuracy obtained by using a pure LK tracker in combination with D1 and D2, respectively. The D3 column represents the accuracy obtained by the motion-adjusted tracker combined with D2.

The results show that D3 achieves almost perfect performance. This means that the proposed motion-adjusted tracker is far superior to the LK tracker by avoiding inaccurate detection. This can be confirmed by the results shown in Fig. 6, in which the proposed tracker yields a more reliable result than the LK tracker.

IV. Conclusions

In this letter, we proposed a real-time apartment building detection and tracking method. Using the newly added prototype features, we trained a cascade of boosted classifiers so that the reliability and effectiveness of the apartment detection were guaranteed as shown in the experimental results. We also presented a motion-adjusted tracker to make the proposed method more robust and reliable. We expect that the proposed method will be applicable to many applications, such as autonomous vehicles and robotics.

References

[1] W. Narzt, G. Pomberger, A. Ferscha, D. Kolb, R. Müller, J. Wieghardt, "A New Visualization Concept for Navigation Systems," *LNCS*, vol. 3196, 2004, pp. 440-451.

[2] S.Lee, K. Choi, I. Joo, S. Cho, J. Park, "Design and Implementation of 4S-Van: A Mobile Mapping System," *ETRI Journal*, vol. 28, no. 3, 2006, pp. 265-274.

[3] Y. Li and L.G. Shapiro, "Consistent Line Clusters for Building Recognition in CBIR," *Proc. IEEE ICPR*, vol. 3, 2002, pp. 952-956.

[4] Y.K. Kim, K.S. Lee, K.H. Choi and S.I. Cho, "Building Recognition for Augmented Reality Based Navigation System," *Proc. IEEE CIT*, 2006, pp. 131-131.

[5] V. Paul, and J. Michael, "Rapid Object Detection Using a Boosted Cascade of Simple Features," *proc. IEEE CVPR*, 2001, pp. 511-518.

[6] R. Lienhart and J. Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection," *proc. IEEE ICIP*, 2002, pp. 900-903.

[7] J.Y. Bouguet, Pyramidal Implementation of the Lucas Kanade Feature Tracker, Intel Corporation, Microprocessor Research Labs, 2000, http://www.intel.com/research/mrl/research/opencv/.