The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON)
Nov. 5-8, 2007, Taipei, Taiwan

3D Object Tracking Using Mean-Shift and Similarity-Based Aspect-Graph Modeling

Jwu-Sheng Hu, Member, IEEE, Tzung-Min Su, Student Member, IEEE, Chung-Wei Juan, and George Wang
Department of Electrical and Control Engineering, National Chiao-Tung University, Taiwan
jshu@cn.nctu.edu.tw

***Abstract*—The mean shift algorithm is a popular method in the field of 2D object tracking due to its simplicity and robustness over slight variations of lighting condition, scale and view-point over time. However, the appearance of 3D object might have distinctive variations for different viewpoints over time. In this work, a novel method for tracking 3D objects using mean-shift algorithm and a 3D object database is proposed to achieve a more precise tracking. A 3D object database using similarity-based aspect-graph is built from 2D images sampled at random intervals from the viewing sphere. Contour and color features of each 2D image are used for modeling the 3D object database. To conduct tracking, a suitable object model is selected from the database and the mean-shift tracking is applied to find the local minima of a similarity measure between the color histograms of the object model and the target image. The effectiveness of the proposed method is demonstrated by experiments with objects rotating and translating in space.**

I. Introduction

OBJECT tracking is a challenging problem due to the need of real-time computing, complex background, and variations of lighting condition, scaling and viewpoint variations over time. The mean shift algorithm [1] is a popular method in this field because of its simplicity and robustness. The goal of mean-shift algorithm is to find the local minima of a similarity measure between the weighted feature histograms of the object model and target image. In [1], the object location can be found by iteratively finding the local minima of the similarity measure of weighted color histogram using fixed kernel. Many researches are studied in these years to improve the work of [1]. For example, location and scale were both estimated in the work of [2]. In feature space, both color and shape features are adopted in [3] instead of only using color feature. Furthermore, spatiogram is proposed in [4] to replace histogram to improve the robustness. An adaptive kernel model is proposed in [5] to solve rotation and translation, and a tunable representation for tracking using a set of spatial kernels with variable bandwidths is proposed in [6]. Despite these research efforts, the tracking accuracy of mean-shift algorithm on 3D object still suffers from the variations of appearance of the object due to the change of viewpoint over time.

In this paper, recognizing 3D objects with a 3D object database is proposed in this work to provide a suitable template for mean-shift tacking. Existing theorems about the high-level 3D object perception can be classified as object-center and viewer-center representation based on the coordinate system [7]. An alternative classification can be observed as model-based and view-based representation based on the constituent elements [8]. The viewer-center and view-based framework conforms to intuition of human perception that a person can memorize an unknown object with several major views of it and does not need an exhaustive 3D object model. Usually only a single view is needed later for identifying the 3D object based on past experiences. The simplest view-based description of an object is a densely sampled collection of views which are treated independently. Although the object can be described in a greater detail when a large number of 2D views are collected and memorized, the computing time for recognition as well as memory space requirement prohibit its usage in practice. Therefore, some methods have been studied to extract a minimal set of object views. For example, aspect-graph representation [9] focuses on changes in the shape of the projection of the object. The vertices of an aspect-graph are the characteristic views that extracted from some points on a transparent viewing sphere with the object in its center. Those characteristic views are extracted as the aspects to describe the object from the densely sampled collection of object views using visual events. A visual event occurs when the appearance of an object changes between two neighbor views.

The traditional aspect-graph method [10] is based on an assumption that an object belongs to a limited class of shapes and characteristic views can be extracted using prior knowledge of the object. In our previous work [11], a similarity-based aspect-graph, which extended the work of [12], is proposed to extract a minimal set of object views and allowing an incremental update of the aspect-graph database. Training views of an object in [11] are sampled at random intervals from a viewing sphere and the object representation can be updated after gathering a new object view without resorting all the previous collected 2D views.

The object database described above is used as a pool of templates when performing object tracking. The contour obtained from the current tracking result is utilized to find out the view angle of the object and the template for mean-shift algorithm is updated when necessary. It can be shown that the dynamic template (e.g., template generated from each image) leads to an accumulated error. The proposed method with absolute template indexing can prevent the error from drifting. Fig. 1 illustrates the block diagram of the overall scheme.

The rest of the paper is organized as follows: Section II describes the procedure of extracting contour and color features that are used to measure the similarity between two 2D views. Next, Section III describes the novelty of this work; similarity-based aspect-graph representations of 3D objects are

1-4244-0783-4/07/$20.00 ©2007 IEEE

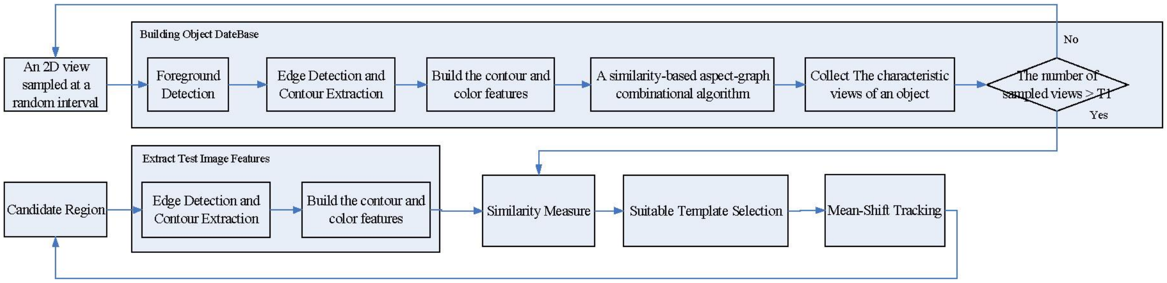
An 2D view sampled at a random interval
Building Object DateBase
Foreground Detection
Edge Detection and Contour Extraction
Build the contour and color features
A similarity-based aspect-graph combinational algorithm
Collect The characteristic views of an object
The number of sampled views > T1
No
Yes
Candidate Region
Extract Test Image Features
Edge Detection and Contour Extraction
Build the contour and color features
Similarity Measure
Suitable Template Selection
Mean-Shift Tracking

Fig. 1. Basic workflow of the proposed framework; T1 denotes the sufficient number of sampled views for building the aspect-graph representation of an object.

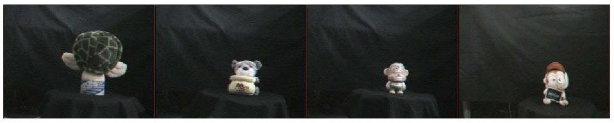

Fig. 2. The image database that contains 4 3D rigid objects, where object 1, object 2…object 4 are listed from left to right.

built from 2D objects views and then be used for selecting the suitable template for mean-shift tracking. Subsequently, some experimental results are presented to demonstrate the performance of the proposed method in rigid object tracking. Conclusions are finally made in Section V.

II. Feature Extraction

A. *Foreground Detection and Contour Extraction*

The first image database contains 4 real rigid objects and is listed in Fig. 2. According to the effects of lighting, shadows and highlights need to be removed before extracting the object features. Therefore, a robust background subtraction framework in our previous work [13] is applied to extract the foreground regions with the consideration to shadows and highlights. The usage of foreground detection provides the flexibility to build the object database even in an out-of-control environment.

In order to extract the shape information from the foreground object, Canny edge detection [14] is applied to extract the shape edge and Gradient Vector Flow Snake (GVF) [15] is then applied to extract the contour information. The contour information is included in a set Z, which is composed of N points z_i, where z_i can be described as a complex form as (1).

$$Z = \{z_i\} = \{x_i + jy_i\},\ 0 \le i < N \tag{1}$$

B. *Contour Feature: Fourier descriptor*

In order to avoid the variation in shift and scale, the edge points inside the set Z are re-sampling by (2),

$$\tilde{Z} = \{\tilde{z}_i\} = \{(\tilde{x}_i + j\,\tilde{y}_i\} = \{L_c[(x_i - x_c) + j(y_i - y_c)]/L\} \tag{2}$$

where $0 \le i < N$, (x_c, y_c) is the mean of (x_i, y_i). L means the contour length and L_c means the expected contour length.

Then the Fourier transform is applied on $\tilde{Z}$ to calculate the Fourier descriptors. The first T_2 magnitude parts are extracted as the main feature to describe the object shape without the variations on the high-frequency noises. The equation to extract the main feature is described as (3).

$$F_m = \{|f_{t'}|, |f_{N-t'}|\},\ 0 \le t' < T_2 \tag{3}$$

where $|f_{t'}|$ means the magnitude part of Fourier descriptor at the $2\pi t'/N$ frequency.

C. *Color feature: background-weighted histogram*

In the work of [1], background-weighted histogram is proposed to improve the robustness of color histogram by incorporating with background information. Background-weighted histogram reduces the variations caused from similar target features with background and inaccurate description of the target.

Let $\{\hat{o}_u\}_{u=0\ldots M-1}$ (with $\sum_{u=0}^{M-1} \hat{o}_u = 1$) be the discrete histogram of the background in the feature space and $\hat{o}^*$ be its smallest nonzero entry. This representation is computed in a region around the target. The extent of region is application dependent and we use an area equal to 1.5 times the target area. The weights are then computed as (4)

$$\left\{v_u = \min\left(\hat{o}^* / \hat{o}_u, 1\right)\right\}_{u=0\ldots M-1} \tag{4}$$

The background-weighted histogram is then defined as (5)

$$q_u = Cv_u \sum_{i=0}^{N-1} k(\| x_i^* \|^2)\delta\left[b(x_i^*) - u\right] \tag{5}$$

with the normalization constant C expressed as (6)

$$C = 1/(\sum_{i=0}^{N-1} k(\| x_i^* \|^2) \sum_{u=0}^{M-1} v_u \delta\left[b(x_i^*) - u\right]) \tag{6}$$

where $\{x_i^*\}_{i=0\ldots N-1}$ be the normalized pixel location in the region defined as the target model. The region is centered at 0. $k(x)$ is a kernel function that assigns smaller weights to pixels farther from the center. The function $b: \mathrm{R}^2 \to \{0\ldots M-1\}$ associates to the pixel at location x_i^* and the index $b(x_i^*)$ denotes its bin in the quantized feature space. Function δ is the delta function.

III. Similarity-Based Aspect-Graph and Mean-Shift Tracking

A. *Similarity Measure*

In order to calculate the similarity between 2D views, a similarity measure metric is necessary to applied on the extracted contour and color features. Suppose U and V

denote one kind of feature extracted from two 2D views and L means the feature length, which are described as (7).

$$U = \{u_0, \cdots, u_i, \cdots, u_{L-1}\} \qquad (7)$$
$$V = \{v_0, \cdots, v_i, \cdots, v_{L-1}\}$$

The similarity measures between contour features are then calculated using 1-norm distance, which is defined as (8).

$$d_F(u,v) = \sum_{i=0}^{L-1} |u_i - v_i|,\ L = N \qquad (8)$$

In the work of [1], Bhattacharyya Coefficient is proposed to measure the similarity among target model and candidates. We apply it on the similarity measure between two weighted-color histograms, which is defined as (9).

$$d_B(u,v) = \sum_{i=0}^{L-1} \sqrt{q_u(i) \cdot q_v(i)},\ L = M \qquad (9)$$

B. Generation of Aspects and Characteristic Views

In our previous work [12], a similarity-based aspect-graph is proposed to present the 3D object using a minimal set of 2D views. The aspects of 3D objects are extracted using 2D views sampled at random intervals. Moreover, object representations become more and more detailed using new 2D views by only calculating the similarity measures among the new view and characteristic views.

Suppose V_{new}^n means the new sampled view of the n_{th} object, $C_m^n(i)$ means the i_{th} characteristic view of the m_{th} aspects of the n_{th} object, $C_{m^{\min}-1}^n$ and $C_{m^{\min}+1}^n$ means the neighbor views of $C_{m^{\min}}^n$ that has the minimum distance with V_{new}^n, $A_{m^{\min}}$ means the aspects that has the minimum distance with V_{new}^n, where $m^{\min}$ means the index of $A_{m^{\min}}$. Then four steps are imposed to form aspects and characteristic views as Step A-1 to A-4 and the flowchart of the modified aspect-graph representation is illustrated as Fig. 2.

Step A-1:

When the number of existed aspects of the n_{th} object equals zero, V_{new}^n is regarded as a characteristic view of a new aspect.

Step A-2:

When the number of existed aspects of the n_{th} object equals one or two:

(A-2.1) If (10) and (11) both meet, V_{new}^n is combined into the $m^{\min}$ aspect and the characteristic view of the aspect keeps the same;

(A-2.2) Otherwise, if (10) is satisfied but (11) is not, V_{new}^n is combined into the $m^{\min}$ aspect and is regarded as a new characteristic view of the $m^{\min}$ aspect;

(A-2.3) Otherwise, if (10) and (11) are both violated, a new aspect of the n_{th} object is built, and V_{new}^n is regarded as the new characteristic view of the new aspect.

$$\min_{all\ \mathrm{C}_\mathrm{m}^\mathrm{n} \in A_{m^{\min}}} d_F\left(V_{new}^n, C_m^n\right) < T_3 \qquad (10)$$

$$\min_{all\ \mathrm{C}_\mathrm{m}^\mathrm{n} \in A_{m^{\min}}} d_B\left(V_{new}^n, C_m^n\right) < T_5 \qquad (11)$$

where T_3 and T_5 are both predefined threshold value.

Step A-3:

When the number of existed aspects of the n_{th} object is equal to or greater than three,

(A-3.1) If (12) or (13) meet and (11) conflicts, a new aspect is built up and V_{new}^n is regarded as the characteristic view of the new aspect.

(A-3.2) Otherwise, if (12) and (13) both conflict and (11) meets, V_{new}^n is combined into the $m^{\min}$ aspect and the characteristic view of the $m^{\min}$ aspect keeps the same.

(A-3.3) Otherwise, if (11), (12) and (13) are all violated, V_{new}^n is combined into the $m^{\min}$ aspect and is regarded as a new characteristic view of the $m^{\min}$ aspect.

$$\min_{all\ \mathrm{C}_\mathrm{m}^\mathrm{n} \in A_{m^{\min}}} d_F\left(V_{new}^n, C_m^n\right) > T_4 \qquad (12)$$

$$T_3 < \min_{all\ \mathrm{C}_\mathrm{m}^\mathrm{n} \in A_{m^{\min}}} d_F\left(V_{new}^n, C_m^n\right) < T_4 \ \ and \ \ d_F\left(V_{new}^n, C_{m^{\min}\pm1}^n\right) > T_3 \qquad (13)$$

Moreover, if a new aspect is built, the aspect order can be decided using (14). If the similarity distance between V_{new}^n and $C_{m^{\min}+1}^n$ is larger than the similarity distance between V_{new}^n and $C_{m^{\min}-1}^n$, the new aspect is inserted between aspect $m^{\min}$ and aspect $m^{\min-1}$. Otherwise, the new aspect is inserted between aspect $m^{\min}$ and aspect $m^{\min+1}$. Therefore, the similar aspects are close to each other.

$$d_F\left(V_{new}^n, C_{m^{\min}+1}^n\right) > d_F\left(V_{new}^n, C_{m^{\min}-1}^n\right) \qquad (14)$$

C. Object Recognition using 2D Characteristic Views

After building the aspects-graph representation of each 3D object, a test view of an unknown object can be recognized using the similarity measure with the contour and color features. Two steps are imposed as follows:

Step B-1:

The test view of an unknown object is compared with the characteristic views of the database via contour features. Then, the first T_6 2D characteristic views in the database having the smallest similarity distance with the test 2D view via contour features are preserved to be further recognized.

Step B-2:

Suppose A_{T_6} is defined as the set that contains the T_6 2D characteristic views described at the Step B-1 , then the final similarity distance can be calculated with the color features

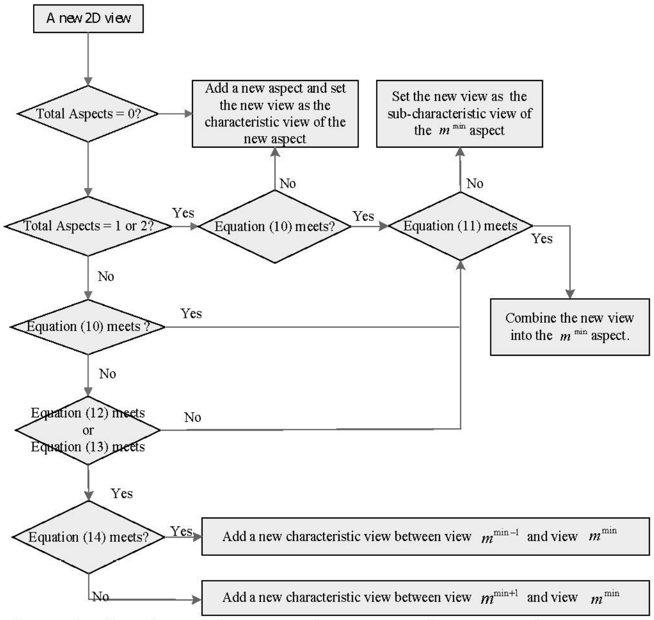
A new 2D view
Total Aspects = 0?
Add a new aspect and set the new view as the characteristic view of the new aspect
Set the new view as the sub-characteristic view of the m^{min} aspect
No
No
Total Aspects = 1 or 2?
Yes
Equation (10) meets?
Yes
Equation (11) meets
Yes
No
Equation (10) meets ?
Yes
Combine the new view into the m^{min} aspect.
No
Equation (12) meets or Equation (13) meets
No
Yes
Equation (14) meets?
Yes
Add a new characteristic view between view m^{min-1} and view m^{min}
No
Add a new characteristic view between view m^{min+1} and view m^{min}

Fig. 3. The flowchart of the proposed aspect-graph representation

by (15).

$$d(V_j^i, C_m^n) = \sqrt{(d_B((V_j^i, C_m^n))^2 + (w_d \cdot (L_{main} / L_{Max})^2} \qquad (15)$$

where w_d is a weight, V_j^i means the 2D view of an unknown object, C_m^n denotes the m_{th} characteristic view of the n_{th} object in the database , L_{main} denotes the similarity distance calculated using contour features between the unknown object and the m_{th} characteristic view of the n_{th} object, which is defined as (16) and (17).

$$L_{main} = d_F(V_j^i, C_m^n)), \text{ where } C_m^n \in A_{T_6} \qquad (16)$$

$$L_{Max} = \arg \max_{all\ C_m^n \in A_{T_6}} (d_F(V_j^i, C_m^n)) \cdot \qquad (17)$$

D. 3D Object Mean-Shift Tracking

Let $\{x_i^*\}_{i=1...n_h}$ be the normalized pixel locations of the target candidates, centered at y in the current frame. The probability of the feature $u = 1...m$ in the target candidate is given by (18)

$$\hat{p}_u(y) = C_h v_u \sum_{i=1}^{n_h} k(\| (y - x_i)/h \|^2) \delta[b(x_i) - u] \qquad (18)$$

with the normalization constant C_h expressed as (19)

$$C_h = 1/(\sum_{i=1}^{n} k(\| (y - x_i)/h \|^2) \sum_{u=1}^{m} v_u \delta\left[b(x_i) - u\right]) \qquad (19)$$

From the work of [1], the new location of object $\hat{y}_1$ can be calculated by (20) and (21) and then the object location is moved from the current location $\hat{y}_0$ to the new location $\hat{y}_1$ till (22) is satisfied, otherwise $\hat{y}_0 = \hat{y}_1$ and repeats (20) to (22).

$$\hat{y}_1 = \frac{\sum_{i=1}^{n_h} x_i w_i g(\| (\hat{y}_0 - x_i)/h \|)}{\sum_{i=1}^{n_h} w_i g(\| (\hat{y}_0 - x_i)/h \|)} \qquad (20)$$

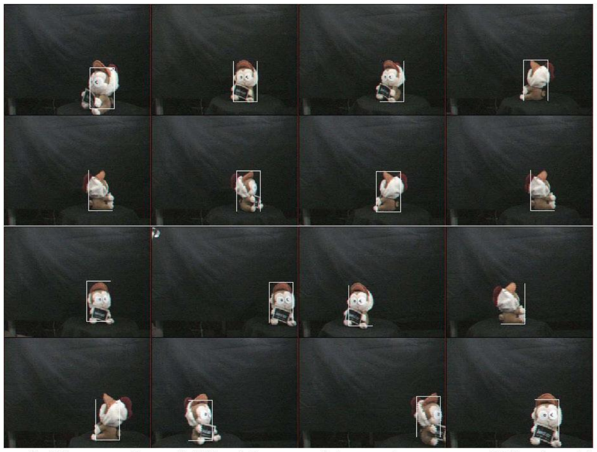

Fig. 4. The results of 3D object tracking using mean-shift algorithm without 3D object database, where frame 76, 141, 205, 341, 373, 437, 469, 509, 733, 805, 885, 965, 990, 1095, 1135, 1261 are listed from left to right and from top to down.

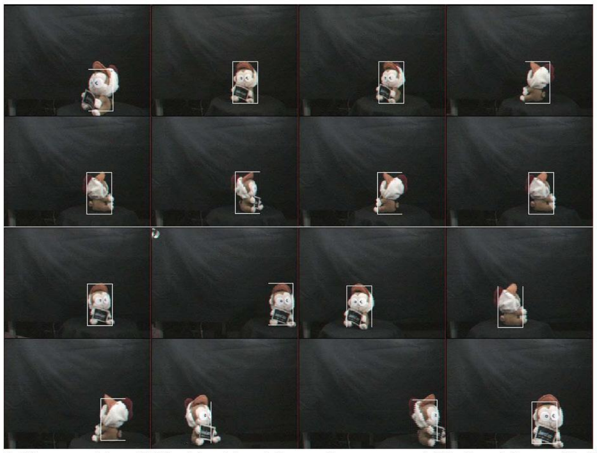

Fig. 5. The results of 3D object tracking using mean-shift algorithm with 3D object database. The frame index is the same as Fig. 4.

where $g(x) = -k'(x)$ and w_i can be described as (21)

$$w_i = \sum_{u=1}^{m} \sqrt{q_u \big/ \hat{p}_u(\hat{y}_0)} \, \delta[b(x) - u] \qquad (21)$$

$$\| \hat{y}_1 - \hat{y}_0 \| < \varepsilon \qquad (22)$$

After calculating the new location, a new region is defined using the new location and then the object inside the region is recognized using the 3D object database. A suitable template is selected as the new target model q_u for next mean-shift tracking.

IV. Experimental Results

This section describes several experiments to demonstrate the effectiveness of the proposed method. SONY EVI-D30 PTZ camera is used to capture object views. The 3D object database is built to test the proposed method. The training views of each object are captured in random intervals and each one contains 72 views. To test the proposed algorithm, motion video of each object is captured which contains about 2500 views for each object. Furthermore, the computing time taken to recognize object and mean-shift tracking was about

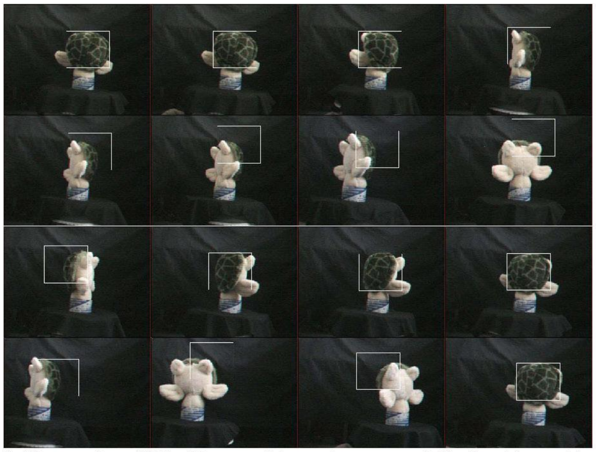

Fig. 6. The results of 3D object tracking using mean-shift algorithm without 3D object database, where frame 13, 36, 80, 127, 135, 144, 194, 213, 229, 242, 271, 958, 1028, 1345, 1394, 1646 are listed from left to right and from top to down.

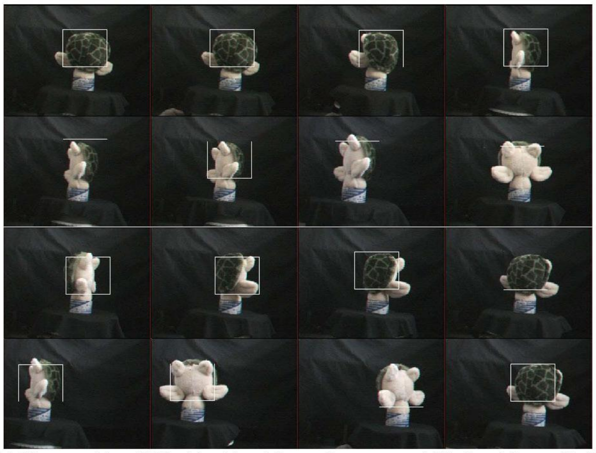

Fig. 7. The results of 3D object tracking using mean-shift algorithm with 3D object database. The frame index is the same as Fig. 6.

one seconds with P4 2.8G CPU and 1GB RAM. The parameters used in the following experiments are: $T_1 = 72$, $T_2 = 25$, $T_3 = 336$, $T_4 = 640$, $T_5 = 0.85$, $T_6 = 3$, $N = 256$, $M = 4096$, and $L_c = 250$.

A. Similar appearances of each view-point over time on simple background

In the first experiment, an object that has similar appearance of each view-point over time is used for testing the robustness of the mean-shift tracking with using 3D object database. Certain representative frames are selected and shown in Fig. 4 and Fig. 5. These frames show that the object has motions with rotation and shift. In Fig. 4, the target model is set as the front of the monkey and then the mean shift tracking is applied without using the 3D object database Because the monkey has the similar color distribution with each side, the mean-shift tracker tracks the candidate in all frames. The result is good due to the similar appearance of each view-point of the object. In Fig. 5, the tracking results are calculated using the proposed method and the results are almost the same as the results in Fig. 4. The tracker with our proposed method tracks the candidate correctly.

B. Different appearances from each view-point over time on simple background

In the second experiment, an object that has different appearance of each view-point over time is used for testing the efficiency of the proposed method. The green turtle shell is chosen as the target model in this experiment. In Fig. 6, the green shell can be tracked in the frames 13, 36, and 80. With the rotation, the green part is vanished gradually with the cream-colored part increasing, and the mean-sift tracker loses the candidate from frames 144 to 229. When the green part is back, the tracker tracks the candidate again. The mean shift tracking is applied without using the 3D object database and the estimated new location becomes inaccurate when the appearance is different from the initial template. In Fig. 7, the template used in each frame is replaced with the suitable one from the 3D object database and the tracker with our proposed method tracks the candidate all the time.

V. Conclusions

This work proposes a novel method for tracking 3D objects using mean-shift algorithm and a 3D object database. The 3D object database is built using contour and color features and presents the similarity-based aspect-graph of each 3D object. When the appearance of 3D object changes over the view-point, a suitable object model is provided from the 3D object database and the mean-shift tracking is applied on finding the local minima of a similarity measure between the color histograms of the object model and the target image. When an object has similar appearance of each view-point, both the proposed method and the traditional mean-shift tracks the candidate properly. However, the proposed method tracks the candidate well when an object has different appearance of each view-point, but the traditional mean-shift method fails. The proposed method solves the 3D object tracking problem and the effectiveness of the proposed method is demonstrated by experiments.

Acknowledgments

This work was supported by National Science Council of the R.O.C. under grant no. NSC94 − 2218 − E009064 and DOIT TDPA Program under the project number 95-EC-17-A-04-S1-054.

References

[1] D. Comaniciu and M. Peter, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis And Machine Intelligence*, vol. 24, no. 5, May 2002.

[2] Collins R T., "Mean shift blob tracking through scale space," *in Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 234-240, 2003.

[3] K. She, G. Bebis, H. Gu, and R. Miller, "Vehicle tracking using on-line fusion of color and shape features," *In Proc. Int. Conf. on Intelligent Transportation Sys.*, Washington DC, Oct. 2004.

[4] S.T. Birchfield, S. Rangarajan, "Spatiograms versus histograms for region-based tracking", *in Proc. Computer Vision and Pattern Recognition*, pp. 20-25, June 2005.

[5] H. Zhang, W. Huang, Z. Huang, L. Li, "Kernel-Based Method for Tracking Objects with Rotation and Translation", *International Conference of Pattern Recognition (ICPR)*, pp. 23-26 August, 2004.

[6] V. Parameswaran, V. Ramesh, and I. Zoghlami, “Tunable Kernels for Tracking,” *in Proc. of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[7] Peters, G., “Theories of Three-Dimensional Object Perception - A Survey,” Recent Research Developments in Pattern Recognition, Transworld Research Network, 2000.

[8] I. Weiss and M. Ray, “Model-Based Recognition of 3D Objects from Single Images,” IEEE Trans. On PAMI, Vol.23, No.2, pp.116-128, 2001.

[9] Koenderink, J.J. and van Doorn, A.J. “The singularities of the visual mapping,” Biol. Cyber. 24:51-59, 1976.

[10] Ilan Shimshoni, Jean Ponce, “Finite-Resolution Aspect Graphs of Polyhedral Objects,” IEEE Trans. on Pattern Anal. Mach. Intell. 19(4): 315-327, 1997.

[11] J.S. Hu, T.M. Su, and C.C. Lin, “Shape Memorization and Recognition of 3-D Objects Using a Similarity-Based Aspect-Graph Approach,” *IEEE Int’l Conf. on Systems, Man, and Cybernetics*, Oct. 2006.

[12] C. M. Cyr and B. Kimia, “A Similarity-Based Aspect-Graph Approach to 3D Object Recognition,” in International Journal of Computer Vision, 57(1):5–22, 2004.

[13] J.S. Hu, T.M. Su and S.C. Jeng, “Robust Background Subtraction with Shadow and Highlight Removal for Indoor Environment Surveillance,” *IEEE/RSJ Int’l Conf. on Intelligent Robots and Systems*, Oct. 2006.

[14] J. Canny, “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No.6, 1986.

[15] C. Xu and J. L. Prince, “Gradient Vector Flow: A New External Force for Snakes,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 66-71, 1997.