Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

A Robust Visual Servo System for Tracking an Arbitrary -Shaped Object by a New Active Contour Method

Pei-Ying Chen[1] *Cheng-Ming Huang*[1] *Li-Chen Fu*[1,2]

Department of Electrical Engineering [1]
Department of Computer Science and Information Engineering [2]
National Taiwan University, Taipei, Taiwan, R.O.C.

E-mail: lichen@ccms.ntu.edu.tw

Abstract

This paper presents a real-time, highly reliable, open field visual tracking system, which can automatically detect an arbitrary-shaped object in 3-D space and find out its location so that the camera platform can be controlled to keep the target centered in the monitor image. Even if the object goes through highly cluttered environment or is occluded by other objects, the system should not fail to work properly. And the total processing period is less than 34 ms. The overall system consists of a motion detector, a snake-based outline extraction, a hybrid tracking methodology, and a VPDA filter which evolves from Probabilistic Data Association filter (PDA filter). At last, the effective functionality of the visual servo system are confirmed by a series of experiments.

1 Introduction

Visual tracking has been an important topic in computer vision and robotics fields. For practical visual tracking systems, there are some basic functionalities required: real-time, automation, and robustness to nonideal situations, such as occlusion and cluttered environment. Surveys show that individual problems mentioned above have been solved. For example, CONDENSATION algorithm [5] is highly regarded since it solved both the real-time and the robustness problems, but not the automation one. Up to now, however, no literature seems to be found able to cope with all the problems completely. Therefore, an integrated algorithm is presented in the paper to overcome all three problems mentioned above as well as to deal with the occlusion problem by combining template matching and Snake. The proposed algorithm also overcomes the disadvantages of classical Snake.

2 Tracking System

The architecture of our visual tracking system can be easily shown by the block diagram in Fig.1. It consists of four subsystems, including motion detection, Snake -based outline extraction, hybrid tracking algorithm and visual probability data association (VPDA) filter. The on-line image sequences are grabbed by a camera, which is mounted on a pan-tilt servo platform.

2.1 Motion Detection

There are several advantages of using motion detection before tracking an object. First, it provides two important clues to an automatically tracking system. Second, we can determine a smaller search range to reduce computation time due to the position

0-7803-8335-4/04/$17.00 ©2004 AACC

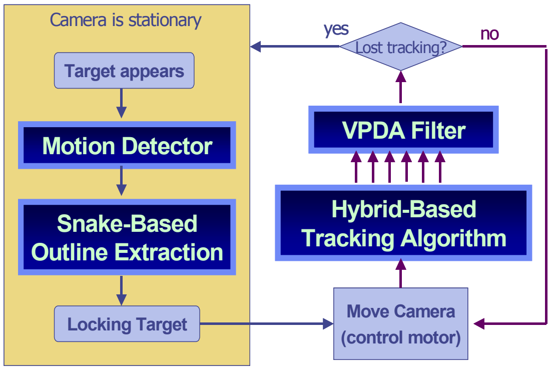


Fig.1 The overview of overall system architecture

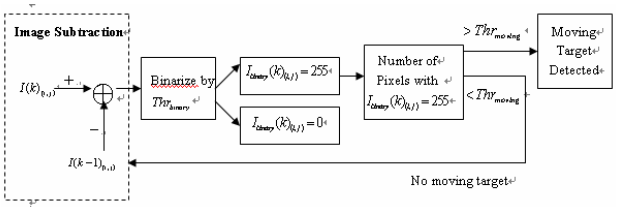


Image Subtraction
$I(k)_{(i,j)}$
+
−
$I(k-1)_{(i,j)}$
Binarize by Thr_{binary}
$I_{binary}(k)_{(i,j)} = 255$
$I_{binary}(k)_{(i,j)} = 0$
Number of Pixels with $I_{binary}(k)_{(i,j)} = 255$
$> Thr_{moving}$
$< Thr_{moving}$
Moving Target Detected
No moving target

Fig. 2 Block diagram of the motion detection

of the moving object in the image, obtained by motion detection with a stationary camera. Third, the segmentation of the moving object can be the basis for automatic initialization of Snake.

As shown in Fig.2, two consecutive image frames $I(k)$ and $I(k-1)$ are subtracted pixel by pixel, and the results are then binarized. If the total number of white pixels is less than the moving threshold $\mathrm{Thr}_{\mathrm{moving}}$, the motion detection unit continues to examine the next two consecutive frames. On the contrary, if the number exceeds the moving threshold, it means that there is a target to be detected. The concept of "moving edge" is included by doing a logic AND operation between the subtracted image and the edge image of the current frame. This highlights the edges within the moving pixel region to obtain the moving edges in the latest image, which denotes the information of target's outline. Finally, the moving edge image is submitted to generate a proper-sized initialization of Snake for outline extraction followed by a tracking algorithm.

2.2 Snake-Based Outline Extraction

The objective of the section is to extract the outline of the target by active contour models (also called Snakes) [1]. A modified Snake is presented here on the basis of the Greedy algorithm [2]. The proposed Snake has an external constraint force to speed up the convergence to the desired feature of the object, and provides an easy way to determine if there is a noise point in order to avoid wrong convergence. If $v_i = (x_i, y_i)$ for $i = 0,1,\ldots,N$ represents the $N-$length discrete contour, the modified Snake energy is

$$\begin{aligned} E^{*}_{snake} &= \sum_{i=1}^{N} E_{snake}(v_i) \\ &= \sum_{i=1}^{N} \{\alpha_i E_{cont}(v_i) + \beta_i E_{curv}(v_i) + \gamma_i E_I(v_i) + \eta_i E_{dis}\} \end{aligned} \tag{1}$$

where E_{cont} is the continuity energy, E_{curv} is curvature energy, E_I is the image energy and E_{dis} is the constraint energy, called distance energy. The parameters α_i, β_i, γ_i and η_i are used to balance the relative influence of the fourth terms. The mathematical formula of each energy term is

$$E_{cont} = \left| \frac{\bar{d} - |v_i - v_{i-1}|}{MAX\left\{\bar{d} - |v_i(j) - v_{i-1}|\right\}} \right|,$$

$$E_{curv} = \frac{|v_{i-1} - 2v_i + v_{i+1}|^2}{MAX\left\{|v_{i-1} - 2v_i(j) + v_{i+1}|^2\right\}},$$

$$E_I = \frac{\nabla I_{\max} - \nabla I(v_i)}{\nabla I_{\max} - \nabla I_{\min}},$$

$$E_{dis} = \frac{D_{\max} - D(v_i)}{D_{\max} - D_{\min}}, \tag{2}$$

where $\bar{d}$ is the average distance between contour points v_i and $v_i(j)$ represents the eight neighbors of a point v_i for $j = 0,1,...,8$. $\nabla I(v_i)$ denotes the image intensity of the edge image at current position and $D(v_i)$ denotes the absolute value of the distance between the current position and the center of the object; $\nabla I_{\max}$ ($D_{\max}$) and $\nabla I_{\min}$ ($D_{\min}$) denote the maximum and minimum image intensity (distance from the center of the object) in the neighborhood, respectively. The moving direction of the Snake can be decided by adding the distance energy to or subtracting it from the Snake energy. Hence, the automatic initialization of Snake can be solved by combining the modified Snake and motion detector, which will be described in next section.

The concept of Greedy algorithm [2] is to take minimized E^*_{snake} as sum of each single minimized $E_{snake}(v_i)$, that is

$$\min E^*_{snake} \approx \sum_{i=1}^{N} \min E_{snake}(v_i) . \qquad (3)$$

Figure 3(a) demonstrates how the iterative Greedy algorithm works. The energy function is computed for the current position v_i and each of its neighbors. Then the location having the smallest value is chosen as the new position $v_i^{'}$. As a result of repeating the aforementioned process point by point, all points of the contour keep moving forward to their corresponding new positions and then form a new contour. Throughout the processes, the Snake completes an iteration of deformation loop. The Snake will repeat the deformation loop again and again until it converges to the desired feature of the object, as shown in Fig. 3(b).

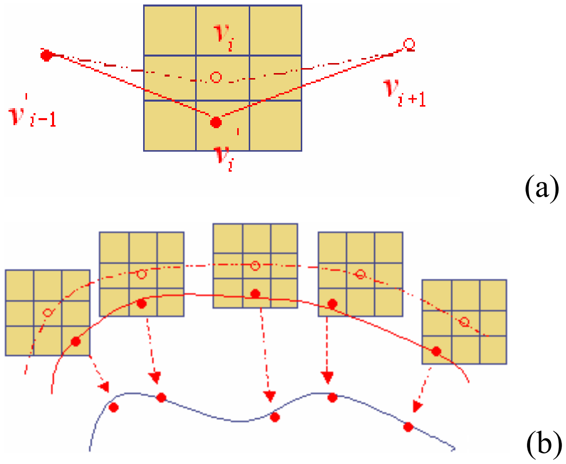


Fig. 3 (a) The demonstration of Greedy algorithm. (b) The deformable of Snake.

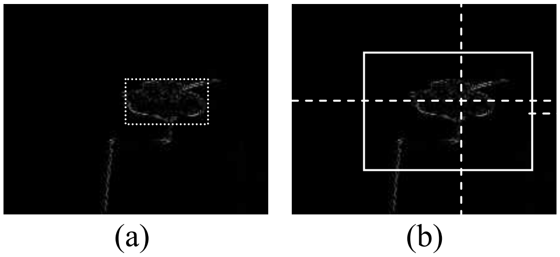


Fig.4 In moving edge image, (a)shows the bounding box of the object, and (b) is the searching area and its center.

From the result of motion detector, a moving object is detected. Assume the whole body of the object is completely inside a 140×140 area, which is the searching area. We choose a proper-sized ellipse (including circles), which can enclose the whole moving object inside, as an initialization of Snakes. And then the center of ellipse is located at the center of the bounding box that contains the object inside. Then, axes length of the initial contour are determined according to the width and height of the bounding box. At the same time, we also use the center of the minimized rectangle as the center of the searching area, as illustrated in Fig. 4.

2.3 Hybrid Tracking Algorithm

Different from the traditional Snake-based tracking algorithms [3]~[10], Snake is only used to extract the outline of the target rather than to perform tracking in our system because that Snake-based tracking algorithms should be under the assumption of slowly moving object, which unfortunately imposes serious constraints on the general use of tracking. The paper presents a contour matching method to make use of the extracted contour model for the purpose of tracking an arbitrary-shaped object. But the method also has a restriction against low-contract environment, just like what Snake-based tracking algorithms encountered. Therefore, we integrate the most commonly used method in visual tracking, called template matching, with contour matching.

The details of contour matching are described as follows. To highlight the contour of an object in an image, the edge image is used for object tracking (see Fig.5). Just like template matching, we need to sum the total gradient values pixel by pixel in the edge image along a pre-extracted contour model. After summing over a contour model, we will shift the center of contour model to the next pixel and compute the total sum along the perimeter of the contour model again. After going through all the search area, we can get the largest value corresponding to the distribution of edge pixels which resembles the object's contour, whereby the true object's contour is successfully located. The numerical process in each searching loop as mentioned above can be summarized by the following normalized sum equation

$$\phi_g(s) = \frac{1}{N_\sigma}\sum_{i=1}^{N_\sigma}|g_i|, \qquad (4)$$

where $s=(x,y)$ represents the position of the center of contour model, N_σ is the total numbers of all pixels along the perimeter of contour model, and g_i denotes the gradient value of each pixel along the perimeter of contour model. Combined with sum of absolute difference (SAD) template matching, The best object position is

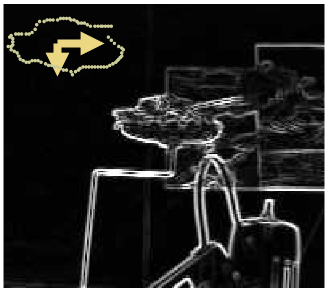


Fig.5 Search the object by matching contour model in the edge image.

$$\mathbf{s}^* = \arg\max_{\mathbf{s}_i\in S}\left\{\overline{\phi}_g(\mathbf{s}_i)+\overline{\phi}_{SAD}(\mathbf{s}_i)\right\}, \qquad (5)$$

where

$$\overline{\phi_g}(\mathbf{s}) = \frac{\phi_g(\mathbf{s}) - \min_{\mathbf{s}_i\in S}\phi_g(\mathbf{s}_i)}{\max_{\mathbf{s}_i\in S}\phi_g(\mathbf{s}_i) - \min_{\mathbf{s}_i\in S}\phi_g(\mathbf{s}_i)} \qquad \text{and}$$

$$\overline{\phi}_{SAD}(\mathbf{s}) = \frac{\max_{\mathbf{s}_i\in S}\phi_{SAD}(\mathbf{s}_i) - \phi_{SAD}(\mathbf{s})}{\max_{\mathbf{s}_i\in S}\phi_{SAD}(\mathbf{s}_i) - \min_{\mathbf{s}_i\in S}\phi_{SAD}(\mathbf{s}_i)}$$

are the normalized form, $\mathbf{s}$ is the center position of the contour model as well as the template, and S is the search area.

3 Trajectory Design

Highly cluttered environment may contain many background looks that are quite resembling the target and then result in false alarms, which may further cause error accumulation in the subsequent tracking. Not only a bad template or contour will be involved, but also the trajectory detection will be mislead, which seriously affect the position of the searching window. Thus, it gives rise to a high possibility of loss of tracking once the best match is a false match. For that reason, Visual Probabilistic Data Association (VPDA) is adopted owing to the challenge from cluttered environment in reality. It can provide a more reliable approach to predict next position of the target, and enhance the robustness of the tracking system, even in cluttered environment. Actually, the VPDA filter is the original probabilistic data association (PDA) filter [7] integrated with the visual information, introduced in [8].

The concept of the PDA filter is to take all possible targets into account instead of the best match one that may be mimicked by parts of the cluttered environment, and then produce a weighted-average output from all possible candidates. The method applied for computing the weights is probability. It computes the posterior association probabilities for all current possible candidate measurements and uses them to form a weighted sum of innovations for updating the target's state in a suitably modified version of the Kalman Filter.

4 Experimental Results

The performance of the Snake-based outline extraction is quite appealing and very fast even if the object appears in a highly cluttered environment, as shown in Fig6. Figure 7~9 demonstrate tracking with a hybrid method with contour matching plus SAD template matching. Experimental results show that our tracking system is highly robust against occlusion, low-contrast environment and rapid motion.

5 Conclusion

Different from the traditional video surveillance system, the major contribution of this work is to establish an integrated visual servo system, which can track an arbitrary-shaped object in highly noisy environment, even with some occlusion on the target. The visual servo system can perform the tracking processes with 320×240 image size in nearly real-time constraint (less than 34 ms) and center the target up to 15-pixel square range tolerance. Furthermore, we made improvement for Snake: i) efficient automatic initialization of Snake via cooperation with a motion detection, ii) modification to Snake energy function so that initial Snake contour may not need to be close enough to the desired target outline, iii) a guideline is devised to update the parameters of Snake for the purpose of more robust performance. In order to robustify the hereby proposed visual tracking method, the so-called hybrid tracking method which integrates contour matching and SAD template matching is developed. Such a novel method is shown to be able to track high-speed moving object subject to very economical computation cost. At last, this thesis further develops a strengthened VPDA filter to enhance the robustness of tracking capability.

Reference

[1] M. Kass, A. Witkin, and D. Terzopoulos, "Snake: Active Contour Models," *Int. J. Comput. Vis.*, Vol. 1, pp. 321-331, 1987.

[2] D.J. Williams and M. Shah. "A Fast Algorithm for Active Contours and Curvature Estimation," *CVGIP: Image Understanding*, Vol. 55, No. 1, pp. 14-26, Jan. 1992.

[3] C. Xu and J.L.Prince, "Snakes, Shapes, and Gradient Vector Flow", *IEEE Trans. Image*

Processing, Vol. 7, No. 3, pp. 359-369, 1998.

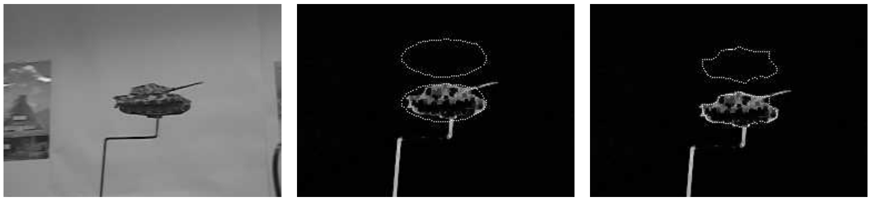
[4] J. Denzeler, and H. Niemann, “A Two Stage Real-Time Object Tracking System,” *In Pavesi'c et al.*

[5] Isard M. and Blake A., “CONDENSATION—Conditional Density Propagation for Visual Tracking,” *Int. J. Computer Vision*, pp. 1-36, 1998.

[6] J. Denzeler and H. Niemann, “A New Energy Term Combining Kalman Filter and Active Contour Models for Object Tracking,” *Machine Graphics Vision*, Vol. 5(1/2), pp. 157-165, 1996.

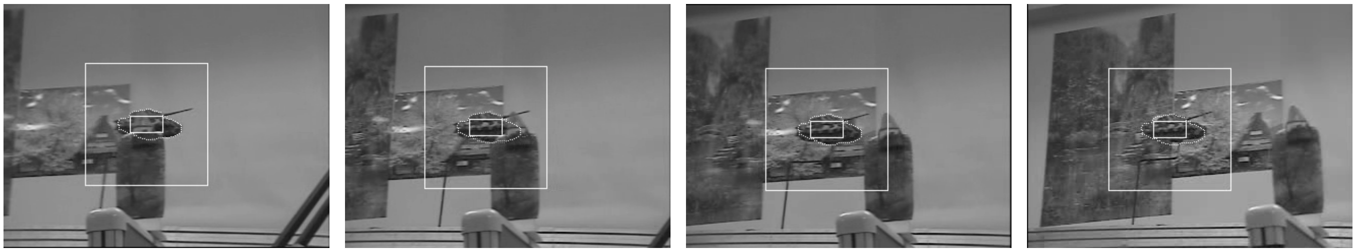
[7] Bar-Shalom, Y., E. Tse, “Tracking in a Cluttered Environment with Probabilistic Data Association,” *Automatica*, Vol. 11, pp. 451-460, 1975.

[8] Liu, D., “Real-Time Visual Tracking in Cluttered Environment with a Pan-Tilt Camera,” *Master Thesis,* Dept. of Electronical Eng., National Taiwan University, 2001.



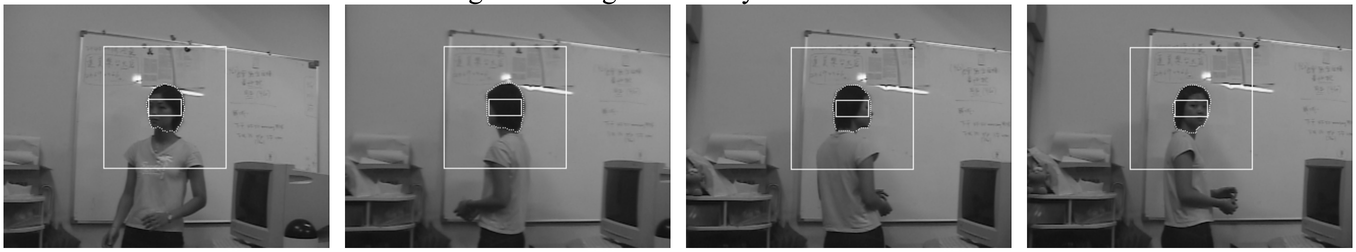
(a) current image (b) initial contour (c) extracted outline

Fig.6 It takes only 9 iterations to extract the tank’s outline with 84 pixel-length of initial contour.



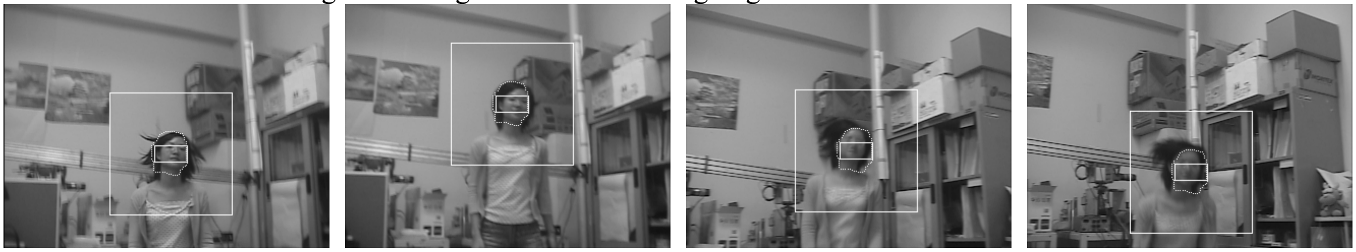
(a) (b) (c) (d)

Fig.7 Tracking with the hybrid method.



(a) (b) (c) (d)

Fig.8 Tracking human head undergoing translation and rotation.



(a) (b) (c) (d)

Fig.9 Tracking quick motion in a cluttered background.