Probabilistic Tracking of the Soccer Ball

Kyuhyoung Choi and Yongdeuk Seo

Dept. of Media Technology, Sogang University,
Shinsu-dong 1, Mapo-gu, Seoul 121-742, Korea
{Kyu, Yndk}@sogang.ac.kr

Abstract. This paper proposes an algorithm for tracking the ball in a soccer video sequence. Two major issues in ball tracking are 1) the image portion of the ball in a frame is very small, having blurred white color, and 2) the interaction with players causes overlapping or occlusion and makes it almost impossible to detect the ball area in a frame or consecutive frames. The first is solved by accumulating the image measurements in time after removing the players' blobs. The resultant image plays a role of proposal density for generating random particles in particle filtering. The second problem makes the ball invisible for time periods. Our tracker then finds adjacent players, marks them as potential ball holders, and pursues them until a new accumulated measurement sufficient for the ball tracking comes out. The experiment shows a good performance on a pretty long soccer match sequence in spite of the ball being frequently occluded by players.

1 Introduction

Analysis of soccer video sequences has been an interesting application in computer vision and image analysis as can be seen by the abundance of recent papers let alone the fever of the soccer itself. Tracking players and ball must be a necessary step before an higher level analysis. There have been some researches on tracking players [1, 2, 3, 4, 5, 6, 7, 8, 9]. Among them, the papers such as [2, 3, 9] have dealt with the ball tracking problem as well. However ball tracking has not been thoroughly studied yet and that is the focus of this paper. Even though ball tracking belongs to single object tracking while player tracking falls within multi-object tracking, ball tracking is not easier than players tracking because of a few things. Usually ball blobs in images are very small, which makes it difficult to derive features from and to be characterized. Sudden changes in its motion is another factor to make it challenging. In addition, occlusion and overlapping with players causes a severe problem in tracking the ball continuously; The ball becomes invisible and appears at places where a continuous prediction could not reach. Yow *et. al.* [2] used intra-frame at regular intervals to detect the soccer ball without taking into account time continuity in the sequence. Seo *et. al.* [3] tracked the ball by template matching and Kalman filter, when there is no player close enough to the ball. When the ball is occluded and lost by nearby players, they searched for the ball in a bounding box around the players until a good

D. Comaniciu et al. (Eds.): SMVP 2004, LNCS 3247, pp. 50–60, 2004.

detection was obtained. In the work of Yamada *et. al.* [9], the nearest image blob in ground-free image to the predicted position by dynamics was considered as the ball among the candidates. The suggested ball tracking algorithm exploited player tracking results and a given background image to obtain images of the ball blob only and its trajectory.

The ball tracking as well as the players tracking in this paper is done by using particle filters, or equivalently, by SMC (Sequential Monte Carlo) methods [10, 11, 12, 13, 14]. In tracking multiple blobs of the players, we utilized the method proposed in [5] to address the problem of particle migration during occlusion between the same team players by probabilistic weighting of the likelihood of a particle according to the distance to its neighbors. This paper then concentrates on tracking the ball in a soccer video sequence. We utilize the result of players tracking in order to obtain measurement images that do not have players' blobs. Two major problems we consider in this paper are 1) the image portion of the ball in a frame is as small as 3×3 and the color is almost white but blurred due to its motion, and 2) the interaction with players causes overlapping or occlusion and makes it almost impossible to detect and predict the ball area in the sequence by a simple usage of a particle filter. To solve the first problem, we remove the image blobs of the players using the result of the players' tracking, segment out the ground field using a lower threshold, and finally accumulate the image blobs through the sequence. After an image filtering, this procedure results in a ball blobs connected continuously. Based on this accumulation image, particles are randomly generated only from those areas that have some blobs, which could be a noise blob, too, due to incomplete segmentation. Then, the particle filter evaluates each of the random particles to produce a tracking result. However, when occlusion or overlapping happens the accumulation does not provide meaningful ball blobs any more. In this case, our tracker changes the ball tracking mode to *invisible* from *visible*, finds and marks players near the location where the ball have disappeared, and chases the players instead of trying to estimate the ball location. This mode transition is done on the basis of the number of meaningful pixels in the accumulation image. For each player who is suspected (marked) to have the ball, searching for the ball is done in a predetermined area with the player position as the center. When a player comes close enough to the marked, it also becomes enlisted. After a detection of the reappearance of the ball by counting the meaningful pixels, the proposed algorithm resumes ball tracking.

Sequential Monte-Carlo method is explained in Section 2. Section 3 deals with pre-image processing and Section 3.1 covers player tracking. The method of ball tracking is discussed in 4. Section 5 provides experimental results and finally Section 6 concludes this paper.

2 Sequential Monte-Carlo Algorithm

Particle filtering or sequential Monte-Carlo (SMC) algorithm estimates the posterior distribution $p(x_t|z_t)$ sequentially, where x_t is the state and z_t is the measurement at time t, given a sequential dynamic equation with Gauss-Markov process. The posterior is represented by random particles or samples from the posterior distribution. When it is not possible to sample directly from the posterior distribution, a proposal distribution q of known random sampler can be adopted to compute the posterior, and in this case the posterior at time t is represented by the set of pairs of particle s and its weight w updated sequentially:

$$w_t = w_{t-1} \frac{p(z_t|x_t)p(x_t|x_{t-1})}{q(x_t|x_{0:t-1}, z_{1:t})} \tag{1}$$

After computation of w_t's for the particles generated from q and normalization $\sum_1^N w_t^i = 1$, where N is the number of particles, the set of particles comes to represent the posterior distribution. Particles have the same weight $1/N$ after re-sampling based on the weights or the posterior distribution.

Taking the proposal distribution as $q = p(x_t|z_{t-1})$ results in $w_t = w_{t-1}$ $p(z_t|x_t)$, saying that the posterior can be estimated by evaluating the likelihoods at each time using the particles generated from the prediction process of system dynamics. Incorporated with resampling, the weight update equation can be further reduced to $w_t = p(x_t|z_t)$, where weight normalization is implied afterwards. This is the method of *condensation* algorithm [10, 11]. To solve the problem at hand by the condensation algorithm, one needs design appropriately the likelihood model $p(z|x)$ and state dynamic model $p(x|x_{t-1})$.

In this paper, the random proposal particles are not generated from $p(x|x_{t-1})$ in the ball tracking, but from a novel proposal distribution taking account of the accumulated measurements. Therefore, we use Equation 1 for updating the weights for the posterior density.

(a) Left image (b) Right image (c) The stitched of (a) and (b)

Fig. 1. Stitched image used as input to the tracking. The two video cameras are looking divergently, but placed so that their focal positions may be as close as possible

3 Pre-image Processing and Player Tracking

The source image to which we applied our tracking algorithm is a stitched version of two images. They are taken from the left and right camera mounted at the stadium respectively and formed into the one stitched image through the homography as shown in Figure 1. The two cameras are placed so that their focal positions may be as close as possible. The homography from one of the two to the stitched is computed by using points matches supplied manually.

(a) I^{bgd} (b) I_k^{ogn}

Fig. 2. Pre-processed images for players tracking. Left shows the constructed background image and right shows in input frame

The background image I^{bgd} is then obtained based on the methods [15, 16, 17]. The field part of original soccer image, I_k^{ogn} at frame k is subtracted to yield field-free image I_k^{sub} exploiting the background image I^{bgd}. Figure 2 shows the corresponding example images. In I_k^{sub}, the pixels of field parts are marked as black. Via CCL (connected component labeling) I_k^{ccl} is obtained. Size filtering deletes colored blobs that have either bigger or smaller enough size not to be considered as those of people.

3.1 Player Tracking

Player tracking is done in a similar way to [5] but using a different likelihood evaluation function. For the image I_k^{sub} of k th frame, state estimates of players ($\boldsymbol{p}_k$) is done by the particle filter assigned respectively. The state vector of a player $\boldsymbol{p}$ is $(\boldsymbol{r}^T, w, h)^T$, where $\boldsymbol{r}$ is $(r_x, r_y)^T$ and represents the center position of a rectangle which a player is considered as. w and h mean the half width and height of the rectangle. Constant velocity is assumed for the dynamics of position and no velocity for the width and height.

$$\boldsymbol{r}_k = 2\boldsymbol{r}_{k-1} - \boldsymbol{r}_{k-2} \tag{2}$$

For particle filtering, each player has N samples or particles, the weight is determined by likelihood evaluation, that is, histogram comparison. A class is assigned to each player and has its model color histogram. When the color histogram of the region for the i-th ($i \in N$) sample s_A^i of player A is h_A^i and the model color histogram of the corresponding class is h_A, the likelihood L_i of s_A^i is expressed with total divergence D [18].

$$L_i = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-D(h_A, h_A^i)^2}{2\sigma^2} \right), \tag{3}$$

where

$$D(h_i, h_j) = \sum_{y \in B} \left\{ h_i(y) \log \frac{h_i(y)}{h_i(y) + h_j(y)} + h_j(y) \log \frac{h_j(y)}{h_i(y) + h_j(y)} \right\}. \tag{4}$$

Here y is an index of bin and B is the set of y given by:

$$B = \{y : h_i(y) > 0, h_j(y) > 0\} \tag{5}$$

The weights are obtained by normalizing the likelihoods L_i and the weighted sum of particles leads to $\hat{\boldsymbol{p}}$, which is the estimate of $\boldsymbol{p}$ at this frame. The problem of particle migration during occlusion between the same team players is resolved by probabilistic weighting of the likelihood of particles according to the distance to its neighbors. See [5]. Some results of tracking multiple players are shown in 7 together with the results of ball tracking.

4 Ball Tracking

The basic idea in ball tracking is that the image consists of the players, ball and static background. So we may get I_t^{ball}, the image of ball only at the frame number t if we remove the portions of the background and players from the image. While player tracking is done at every single frame, ball tracking is batch processed at every m-th frame, where the interval of ball tracking is to produce a long enough accumulated area of the ball blobs. Examples of the accumulation are shown in Figure 3 and in our experiments the ball tracking interval m was 50 frames. If the blobs of players are deleted completely from the background-free image, we can get an accumulation image of I^{ball}s that is supposed to contain white pixels only from the ball area. However, notice that it contains noise pixels, too, due to incomplete background removal and players' blob detection. One could see that the ball has been in *visible* mode through the sequence since there are white accumulated areas (the linear structure in the accumulation image). The discontinuity means that the ball has been *invisible* during a period due to some reasons such as occlusion and overlapping.

The initial location of the ball is automatically detected by the proposed algorithm as following. If the interval m is 60, the accumulation image after the first interval is Figure 3(a) and it gives a filtered image of Figure 4(b) through CCL (See Figure 4(a)) and size filtering. If we have a filtered accumulation image the same as Figure 4(b) except that the non-black pixel value is the frame number instead of RGB color values, we can initialize the ball location, which is likely to be either end of the long blob in Figure 4(b), by finding pixels of minimum frame number. So during the first interval, accumulation images of ball blobs for both RGB color values and the frame number are made.

During the visible mode, we use a first order dynamic model for the ball motion perturbed by Gaussian random noise η:

$$\boldsymbol{x}_t = 2\boldsymbol{x}_{t-1} - \boldsymbol{x}_{t-2} + \eta, \tag{6}$$

where $\boldsymbol{x} = (x, y)$ is the location of the ball. The shape of the ball is modelled simply to be 3×3 rectangular. We measure the color values on the pixels in the 3×3 rectangle whose center is given by $\boldsymbol{x}$ - the state of the ball motion. Hence, our observation model for a ball particle is defined to be:

$$p(\boldsymbol{z}_t|\boldsymbol{x}_t) = \prod_i \prod_c \exp\left(-\frac{(c_i - \mu_c)^2}{\sigma_c^2}\right), \tag{7}$$

(a) After 60 frames

(b) After 200 frames

(c) After 350 frames

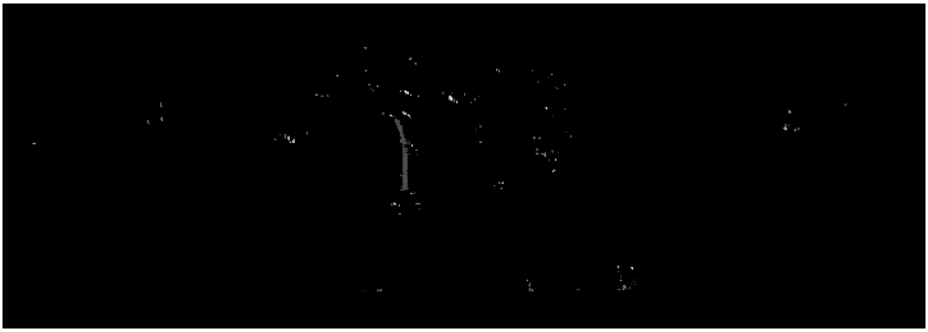
(d) After 500 frames

(e) After 650 frames

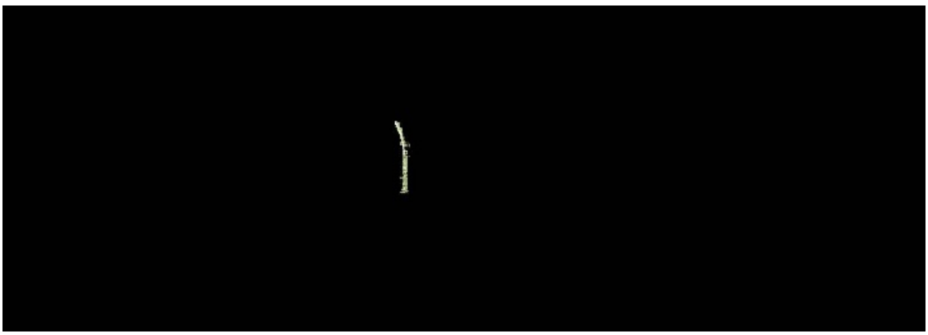
Fig. 3. Accumulation images for the ball blobs

where i denotes a pixel location i in the 3×3 rectangle, c_i the value in RGB color space at the pixel location, and μ_c and σ_c the mean and standard deviation calculated based on the pixel values around the ball area in a few video frames. Particles for the tracking is generated in the image region detected as the ball area after removing the players' blob and the background. Those pixels are designed to have equal probability and hence a uniform random sampler is utilized. The likelihood is evaluated using Equation 1, and the ball location is given by the weighted average of the particles.

When the ball is in the mode of *invisible*,we stop tracking the ball. In this case, the ball is assumed to be possessed by players near the place where the ball has become invisible. As shown in Figure 5, for each of the players who are suspected to have the ball, ball searching is done in the circled area with the player position as the center. Any player who comes close enough to the

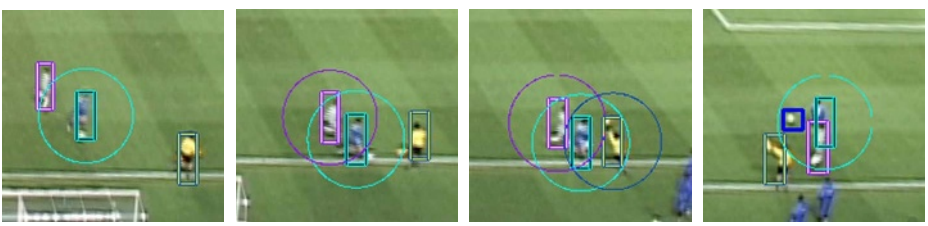


(a) The connected component labelled image of Figure 3(a)



(b) The size-filtered image of Figure 3(a)

Fig. 4. Images in the steps of initializing the ball location



(a) frame 156 (b) frame 191 (c) frame 206 (d) frame 241

Fig. 5. Sub-images of some frames of interest

suspects also becomes enlisted. After the ball reappears and is detected through the accumulation, that is, one end of another ball blob trajectory (e.g. Figure 3) is found, the proposed algorithm resumes normal ball tracking as in the early part of this section.

In order to determine the ball tracking mode, we observe the number of pixels of the ball area in the accumulation image. At the frame number t ($t \neq 0$ and $(k-1)m \leq t < km$ for a natural number k), this value is given as the sum:

$$S_t = \sum_{j \in \{t-1, t, t+1\}} \sum_{l \in W_t} C_j(\boldsymbol{x}_l), \tag{8}$$

where $\boldsymbol{x}_l$ denotes an l-th pixel location in the search window W_t whose center is given by the estimated ball position at the frame number t, and C_j is an indication function:

$$C_j(\boldsymbol{x}) = \begin{cases} 0 & \text{if the color at } I_j^{ball}(\boldsymbol{x}) \text{ is black} \\ 1 & \text{otherwise} \end{cases} \tag{9}$$

Note that we incorporate the three consecutive image measurements in Equation 8 for a robust computation. Mode change is done simply by thresholding. When S_t is smaller than a threshold Th then the tracking mode changes to *invisible*, and as we explained before, the players are kept traced until our tracker finds the re-appearance of the ball pixels, that is, $S_t \geq Th$. Figure 6 shows a graph of $S_t - t$ for the input video of our experiments and S_t shows obvious changes at transitions between the two modes. At the most frames of *ivisible* mode S_t is zero and more than 50 for the *visible*. When $S_t < 15$ in the real

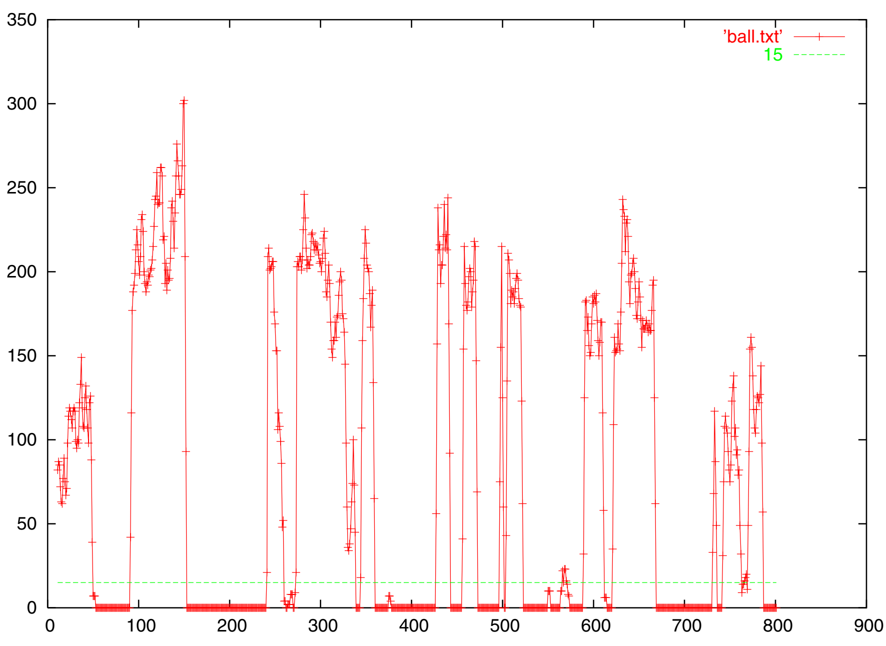


Fig. 6. Graph of $S_t - t$. Counting the number of meaningful pixels to determine the mode of the tracker

experiment, the mode changed to *invisible* and nearby players were traced to find the initiation of the ball blobs.

5 Experiments

Experiments were carried out on a video sequence of 800 images which are stitched as in Section 3. The image size was 1361×480 pixels. Figure 7 shows some frames of the results of which the detail is contained in accompanying video clip. The rectangle around each of the players is colored to show its identity. The inner color of its line stands for his class: ordinary players of each team, goal keeper of each team, and referee. A black circle around the ball means that the ball is not occupied by any player and thus the tracking mode is *visible*, and a colored circle shows the search area whose center is given by the location of the player, who is marked as a candidate having the ball. Notice that the color of the circle and the rectangle of the player are the same.

All the tracking process was done automatically without a manual intervention as well as the estimation of the initial ball location. The interval m was 60 and the threshold Th for the mode transition was set to $Th = 15$ which appears to be reasonable according to Figure 6. Note that even though players frequently repeat to keep, pass or kick the ball, the suggested algorithm showed a very robust performance and did not lose it, which is not guaranteed in other works such as [3, 9].

6 Conclusion

The algorithm presented in this paper have focused on an effective way of tracking the ball in a soccer match video sequence. The result of multiple player tracking was made use of in order to obtain a robust measurement for the ball tracking. By removing the blobs of players, we could obtain an accumulation image of the ball blobs. This accumulation image provided us not only a proposal density for the particle filtering but also a clue to deciding whether the ball was visible or invisible in the video frames. Basically, the ball tracking was done by particle filtering. However, the performance was highly improved by two ingredients: first, taking the accumulation image as the proposal density, and second, mode change by counting the meaningful ball pixels. When the ball was invisible, we pursued every nearby players until the ball pixel came out again. Since the ball pixels were accumulated in time, the tracking algorithm showed in the real experiment a very robust ball tracking results, that was not shown by other studies.

As a future work, we are going to develop a real time system for tracking through the whole game. Another interesting area is extending this work to the case of multiple camera sequences so that a full 3D reconstruction of the soccer game is possible.

(a) full image of frame 13

(b) full image of frame 92

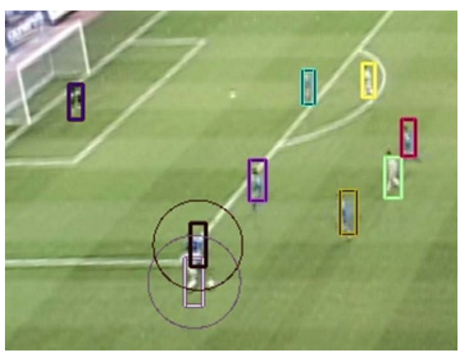
(c) sub-image of frame 152

(d) sub-image of frame 234

(e) sub-image of frame 336

(f) sub-image of frame 498

(g) sub-image of frame 588



(h) sub-image of frame 721

Fig. 7. Examples of result images

References

1. Intille, S., Bobick, A.: Closed-world tracking. In: Proc. Int. Conf. on Computer Vision. (1995)
2. Yow, D., Yeo, B., Yeung, M., Liu, B.: Analysis and presentation of soccer highlights from digital video. In: Proc. Asian Conf. on Computer Vision. (1995)
3. Seo, Y., Choi, S., Kim, H., Hong, K.: Where are the ball and players? soccer game analysis with color-based tracking and image mosaick. In: Proc. Int. Conf. on Image Analysis and Processing, Florence, Italy. (1997)
4. Iwase, S., Saito, H.: Tracking soccer player using multiple views. In: IAPR Workshop on Machine Vision Applications. (2002)
5. OK, H., Seo, Y., Hong, K.: Multiple soccer players tracking by condensation with occlusion alarm probability. In: Int. Workshop on Statistically Motivated Vision Processing, in conjunction with ECCV 2002, Copenhagen, Denmark. (2002)
6. Yoon, H., Bae, Y., Yang, Y.: A soccer image mosaicking and analysis method using line and advertisement board detection. ETRI Journal **24** (2002)
7. Utsumi, O., Miura, K., IDE, I., Sakai, S., Tanaka, H.: An object detection method for describing soccer games from video. In: IEEE International Conference on Multimedia and Expo (ICME). (2002)
8. Kang, J., Cohen, I., Medioni, G.: Soccer player tracking across uncalibrated camera streams. In: Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS). (2003)
9. Yamada, A., Shirai, Y., Miura, J.: Tracking players and a ball in video image sequence and estimating camera parameters for 3d interpretation of soccer games. In: Proc. International Conference on Pattern Recognition. (2002)
10. Kitagawa, G.: Monte-carlo filter and smoother for non-gaussian nonlinear state space model. Journal of Computational and Graphical Statistics (1996)
11. Blake, A., Isard, M.: Active Contours. Springer-Verlag (1997)
12. Liu, J., Chen, R.: Sequential Monte Carlo methods for dynamic systems. (1998)
13. Doucet, A., Godsill, S., Andrieu, C.: On sequential monte-carlo sampling methods for bayesian filtering. (2000)
14. Doucet, A., Freitas, N.D., Gordon, N., eds.: Sequential Monte Carlo Methods in Practice. Springer-Verlag (2001)
15. McKenna, S.J., Raja, Y., Gong, S.: Object tracking using adaptive color mixture models. In: Proc. Asian Conf. on Computer Vision. (1998) 615–622
16. Bevilacqua, A.: A novel background initialization method in visual surveillance. In: IAPR Workshop on Machine Vision Applications. (2002)
17. Elgammal, A., Duraiswami, R., Harwood, D., Davis, L.S.: Background and foreground modeling using nonparametric kernel density for visual surveillance. In: Proceedings of the IEEE. Volume 90. (2002) 1151–1163
18. Lee, L.: Similarity-Based Approaches to Natural Language Processing. PhD thesis, Harvard University, Cambridge, MA (1997)