# Tracking and Recognizing Actions at a Distance

Wei-Lwun Lu and James J. Little

Department of Computer Science,
University of British Columbia,
Vancouver, BC V6T 1Z4, CANADA
{vailen, little}@cs.ubc.ca

**Abstract.** This paper presents a template-based algorithm to track and recognize athlete's actions in an integrated system using only visual information. Usually the two elements of tracking and action recognition are treated separately. In contrast, our algorithm emphasizes that tracking and action recognition can be tightly coupled into a single framework, where tracking assists action recognition and vice versa. Moreover, this paper proposes to represent the athletes by the grids of Histograms of Oriented Gradient (HOG) descriptor. Using of the HOG descriptor not only improves the robustness of the tracker, but also centers the figure in the tracking region. Therefore, no further stabilization techniques are needed. Empirical results on hockey and soccer sequences show the effectiveness of this algorithm.

## 1 Introduction

Vision-based tracking and action recognition systems have gained more and more attention in the past few years because of their potential applications on smart surveillance systems, advanced human-computer interfaces, and sport video analysis. In the past decade, there has been intensive research and giant strides in designing algorithms for tracking humans and recognizing their actions [7].

In this paper, we develop a system that integrates visual tracking and action recognition, where tracking assists action recognition and vice versa. The first novelty of this paper is to represent the target by the HOG descriptor [5]. The HOG descriptor and its variants have been used in human detection [5] and object class recognition [15], and have been shown to be very distinctive and robust. This paper shows that the HOG descriptor can be also exploited in visual tracking and action recognition as well. The second novelty of this paper is an algorithm that solves the tracking and action recognition problems together using learned templates. Given the examples of athletes' appearances of different actions, we learn the templates and the transition between the templates offline. During the runtime, we use the templates and the transition matrix to classify the athlete's actions and compute the most probable template sequence consistent with the visual observation. Moreover, the last template of the most probable sequence can be used for the visual tracking system to search for the next position and size of the athlete, which determines the next visual observation of the athlete.

This paper is organized as follows: In Section 2, we review some related work in visual tracking and action recognition. In Section 3, we introduce the HOG descriptor representation. Section 4 details our tracking and action recognition algorithms. The experimental results in hockey and soccer sequences are shown in Section 5. Section 6 concludes this paper.

## 2 Previous Work

The task of vision-based action recognition can be described as follows: given a sequence of consecutive images containing a single person, our task is to determine the *action* of the person. Therefore, the vision-based action recognition problem is indeed a classification problem of which the input is a set of images, and the output is a finite set of labels.

The input of a vision-based action recognition system is usually a set of *stabilized* images: figure-centric images containing the whole body of a single person (including the limbs). Fig. 3 provides some examples of the stabilized images. In order to obtain the stabilized images, vision-based action recognition systems usually run visual tracking and stabilization algorithms prior to the action recognition [6][22]. Then, the systems will extract relevant features such as pixel intensities, edges, optical flow from the images. These features are fed into a classification algorithm to determine the action of the person. For example, Yamato *et al.* [24] transform a sequence of stabilized images to mesh features, and use a Hidden Markov Model classifier to recognize the actions. Efros *et al.* [6] transform the images to novel motion descriptors computed by decomposing the optical flow of images into four channels. Then, a nearest-neighbor algorithm is performed to determine the person's actions. In order to tackle the stabilization problem, Wu [22] develops an algorithm to automatically extract figure-centric stabilized images from the tracking system. He also proposes to use Decomposed Image Gradients (DIG), which can be computed by decomposing the image gradients into four channels, to classify the person's actions.

The task of visual tracking can be defined as follows: given the initial state (usually the position and size) of a person in the first frame of a video sequence, the tracking systems will continuously update the person's state given the successive frames. During the tracking algorithm, relevant features should be extracted from the images. Some systems use intensities or color information [2][20], some use shape information [8][10], and some use both [1][23]. The tracking problem can be solved either deterministically [2][4][11] or probabilistically [1][10][23]. In order to improve the performance and robustness of the tracker, many systems also combine the tracking system with other systems such as object detection [1][19] and object recognition [12].

In order to simplify the tracking problem, many trackers use a fixed target appearance [4][19][20]. However, having a fixed target appearance is optimistic in the real world because the view point and the illumination conditions may change, and the person constantly changes his poses. In order to tackle this problem, [9][13] project the images of the person to a linear subspace and in-

crementally update the subspace based on the new images. These systems are efficient; however, they have difficulties recovering from drift because the linear subspace also accumulates the information obtained from the images containing only the background. Jepson *et al.* [11] propose the WSL tracker of which the appearance model is dominated by either the stable (S), wandering (W), or lost (L) components. They use an online EM algorithm to update the parameters of the stable component, and therefore the tracker is robust under smooth appearance changes.

The tracking systems that most resemble ours are Giebel *et al.* [8] and Lee *et al.* [12]. Giebel *et al.* [8] learn the templates of the targets and the transition matrix between the templates from examples. However, they do not divide the templates into different actions. During tracking, they use particle filtering [20] to infer both the next template, and the position and size of the target. Lee *et al.* [12] introduce a system that combines face tracking and recognition. They also learn templates of faces and the transition matrix between the templates from examples, and partition the templates into different groups according to the *identity* of the face. During the runtime, they first recognize the identity of the face based on the history of the tracking results. Knowing the identity of the face, the target template used by the tracker can be more accurately estimated, and thus improve the robustness of the tracker.

## 3 The HOG Descriptor Representation

In this paper we propose to use the grids of Histograms of Oriented Gradient (HOG) descriptor [5] to represent the athletes. The HOG representation is inspired by the SIFT descriptor proposed by Lowe [14], and it can be computed by first dividing the tracking regions into non-overlapping grids, and then computing the orientation histograms of the image gradient of each grid (Fig. 1).

The HOG descriptor was originally designed for human detection [5]. In this paper, we will show that the HOG/SIFT representation can be also used in object tracking and action recognition as well. Using the HOG/SIFT representation has several advantages. Firstly, since the HOG/SIFT representation is based on *edge*, the rectangular tracking region can contain the entire body of the athlete (including the limbs) without sacrificing the discrimination between the foreground and background. This is especially the case in tracking athletes in sports such as hockey and soccer because the background is usually homogeneous. Another attractive property of the HOG/SIFT representation is that it is insensitive to the changes of athlete's uniform. This enables the tracker to focus on the *shape* of the athletes but not the colors or textures of the uniform. Secondly, the HOG/SIFT representation improves the robustness of the tracker because it is robust to small misalignments and illumination changes [5][16]. Thirdly, the HOG/SIFT representation implicitly centers the figure in the tracking region because it preserves some spatial arrangement by dividing the tracking region into non-overlapping grids. In other words, no further stabilization techniques [22] need to be used to center the figure in the tracking

region. This helps integrate tracking and action recognition into a single framework. Fig. 1 gives an example of the grids of HOG descriptor with a $2 \times 2$ grid and 8 orientation bins.

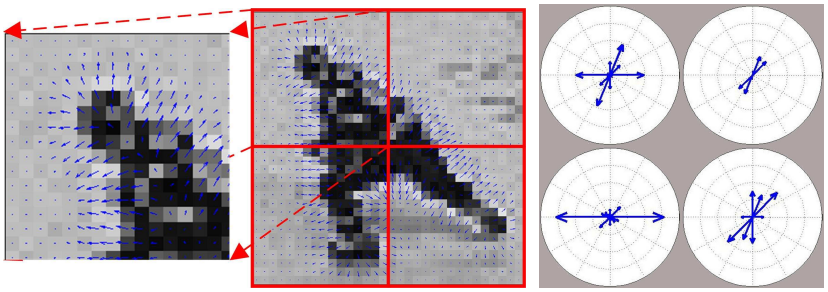


**Fig. 1. Examples of the HOG descriptor:** (a) The image gradient. (b) The HOG descriptor with a $2 \times 2$ grid and 8 orientation bins.

## 4 Tracking and Action Recognition

The probabilistic graphical model of our system (Fig. 2) is a hybrid Hidden Markov Model with two first-order Markov processes. The first Markov process, $\{E_t; t \in \mathbb{N}\}$, contains discrete random variable $E_t$ denoting the template at time $t$. The second Markov process, $\{\boldsymbol{X}_t; t \in \mathbb{N}\}$, contains continuous random variable $\boldsymbol{X}_t$ denoting the position, velocity, and size of a single athlete at time $t$. The random variable $\{\boldsymbol{I}_t; t \in \mathbb{N}\}$ denote the frame of the video at time $t$, and the parameter $\alpha_t$ denote the action of the athlete at time $t$. The joint distribution of the entire system is given by:

$$\begin{aligned} p(\boldsymbol{X}, \boldsymbol{E}, \boldsymbol{I}|\boldsymbol{\alpha}) = p(\boldsymbol{X}_0)p(E_0) \prod_t p(\boldsymbol{I}_t|\boldsymbol{X}_t, E_t, \alpha_t) \cdot \\ \prod_t p(E_t|E_{t-1}, \alpha_t) \prod_t p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) \end{aligned} \tag{1}$$

The transition distribution $p(E_t|E_{t-1}, \alpha_t)$ is defined as:

$$p(E_t = j \mid E_{t-1} = i,\ \alpha_t = a) \ = \ A^a_{ij} \tag{2}$$

where $A^a_{ij}$ is the transition distribution between templates $i$ and $j$ of action $a$.

The continuous random variable $\boldsymbol{X}_t$ is defined as $\boldsymbol{X}_t = \{x_t, y_t, v^x_t, v^y_t, w_t\}^T$, where $(x_t, y_t)$ denotes the center of the athlete, $w_t$ denotes the width of the rectangular tracking region (we currently fix the aspect ratio of the tracking region),

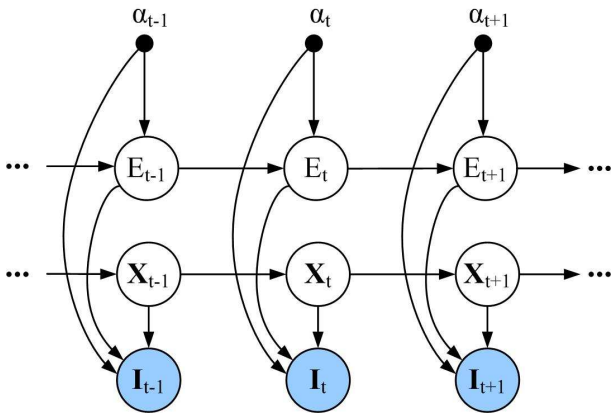


**Fig. 2. Probabilistic Graphical Model** of our system.

and $v_t^x$ and $v_t^y$ denote the velocity of the athlete along the $x$ and $y$ direction, respectively. The transition distribution $p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1})$ is a linear Gaussian:

$$p(\boldsymbol{X}_t \mid \boldsymbol{X}_{t-1}) \;=\; \mathcal{N}(\boldsymbol{X}_t \mid \boldsymbol{B}\boldsymbol{X}_{t-1},\; \boldsymbol{\Sigma_X}) \tag{3}$$

where $\boldsymbol{\Sigma_X}$ is a $5 \times 5$ covariance matrix and $\boldsymbol{B}$ is the dynamics matrix.

In order to track and recognize the athlete's actions simultaneously, we perform the following three procedures at time $t$:

1. **Tracking**: Under the assumption that the the appearance of the athlete changes smoothly, we use the template at time $t-1$ as the target appearance to update the current state of the tracker using particle filtering.
2. **Action Recognition**: To estimate the action $\alpha_t$, we use a Hidden Markov Model classifier [21][24] to determine the athlete's action based on the previous $T$ observations.
3. **Template Updating**: Having the optimal action, we update the current template $E_t$ based on the most probable sequence in the Hidden Markov Model [21][24].

By repeatedly performing these three procedures at time $t$, the system can approximately update the athlete's position, velocity, size, and determine the athlete's action in practice though we do not prove the convergence of the system.

### 4.1 Tracking

The posterior distribution $p(\boldsymbol{X}_t|\boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t})$ can be computed by the following recursion:

$$\begin{aligned} p(\boldsymbol{X}_t|\boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t}) \;&\propto\; p(\boldsymbol{I}_t|\boldsymbol{X}_t, E_t, \alpha_t)\cdot \\ &\int p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1})\; p(\boldsymbol{X}_{t-1}|\boldsymbol{I}_{1:(t-1)}, E_{0:(t-1)}, \alpha_{1:(t-1)})\; d\boldsymbol{X}_{t-1} \end{aligned} \tag{4}$$

Since computing the exact posterior distribution of Eq. (4) is intractable, we use *particle filtering* [10][19][20] to approximate Eq. (4). Assume that we have a set of $N$ particles $\{\boldsymbol{X}_t^{(i)}\}_{i=1\ldots N}$. In each time step, we sample candidate particles from an proposal distribution:

$$\tilde{\boldsymbol{X}}_t^{(i)} \sim q(\boldsymbol{X}_t \mid \boldsymbol{X}_{0:t-1}, \boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t}) \tag{5}$$

and weight these particles according to the following importance ratio

$$\omega_t^{(i)} = \omega_{t-1}^{(i)} \frac{p(\boldsymbol{I}_t|\tilde{\boldsymbol{X}}_t^{(i)}, E_t, \alpha_t)\ p(\tilde{\boldsymbol{X}}_t^{(i)}|\boldsymbol{X}_{t-1}^{(i)})}{q(\tilde{\boldsymbol{X}}_t^{(i)}|\boldsymbol{X}_{0:t-1}^{(i)}, \boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t})} \tag{6}$$

In this paper, we set the proposal distribution by:

$$q(\boldsymbol{X}_t \mid \boldsymbol{X}_{0:t-1}, \boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t}) = p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) \tag{7}$$

and thus Eq. 6 becomes $\omega_t^{(i)} = \omega_{t-1}^{(i)}\ p(\boldsymbol{I}_t|\tilde{\boldsymbol{X}}_t^{(i)}, E_t, \alpha_t)$. We *re-sample* the particles using their importance weights to generate an unweighted approximation $p(\boldsymbol{X}_t|\boldsymbol{I}_{1:t}, E_{0:t}, \alpha_{1:t})$.

The problem of Eq. (6) is that $E_t$ and $\alpha_t$ are unknown. Assuming that the appearance of the athletes changes smoothly, we approximate the current template and action by the previous ones, i.e., $\tilde{E}_t = E_{t-1}, \tilde{\alpha}_t = \alpha_{t-1}$. Following [4][20][19], we define the sensor distribution $p(\boldsymbol{I}_t|\boldsymbol{X}_t, \tilde{E}_t, \tilde{\alpha}_t)$ as:

$$p(\boldsymbol{I}_t|\boldsymbol{X}_t,\ \tilde{E}_t = i,\ \tilde{\alpha}_t = a) \propto exp(-\lambda\ \xi^2(\boldsymbol{H}_t, \boldsymbol{\Pi}_i^a)) \tag{8}$$

where $\boldsymbol{H}_t$ is the HOG descriptor of the image $\boldsymbol{I}_t$ given the state $\boldsymbol{X}_t$, $\boldsymbol{\Pi}_i^a$ is the HOG descriptor of the template $i$ of the action $a$, and $\lambda$ is a constant. The similarity measure $\xi^2(\cdot,\cdot)$ is the Bhattacharyya similarity coefficient [4][20][19] defined as:

$$\xi(\boldsymbol{H}, \boldsymbol{\Pi}) = \left[1 - \sum_{k=1}^{N_k} \sqrt{\boldsymbol{h}_k \boldsymbol{\pi}_k}\right]^{1/2} \tag{9}$$

where $\boldsymbol{H} = \{\boldsymbol{h}_1 \ldots \boldsymbol{h}_{N_k}\}$, $\boldsymbol{\Pi} = \{\boldsymbol{\pi}_1 \ldots \boldsymbol{\pi}_{N_k}\}$, and $N_k$ denotes the dimensionality of the HOG descriptor.

### 4.2 Action Recognition

Knowing the position and size of the athlete, the current HOG descriptor $\boldsymbol{H}_t$ of the tracking region can be computed. Assuming that the previous $T$ observations are generated by the same action, we use a Hidden Markov Model classifier [21] [24] to determine the athlete's current action based on the previous $T$ observations. Let $s = t - T + 1$ denote the time of first observation we use to classify the athlete's action, the likelihood of the previous $T$ observations can be defined as:

$$p(\boldsymbol{H}_{s:t}|\alpha_t) = \sum_{E_t} p(\boldsymbol{H}_{s:t}, E_t|\alpha_t) \tag{10}$$

$$p(\boldsymbol{H}_{s:t}, E_t|\alpha_t) = p(\boldsymbol{H}_t|E_t, \alpha_t) \sum_{E_{t-1}} p(\boldsymbol{H}_{s:(t-1)}, E_{t-1}|\alpha_t)\ p(E_t|E_{t-1}, \alpha_t) \tag{11}$$

The sensor distribution $p(\boldsymbol{H}_t|E_t, \alpha_t)$ is defined as a Gaussian distribution:

$$p(\boldsymbol{H}_t \mid E_t = i, \alpha_t = a) \;=\; \mathcal{N}(\boldsymbol{H}_t \mid \boldsymbol{\Pi}_i^a,\; \boldsymbol{\Sigma}_i^a) \tag{12}$$

where $\boldsymbol{\Pi}_i^a$ and $\boldsymbol{\Sigma}_i^a$ are the mean and covariance of the HOG descriptor of the template $i$ in action $a$.

The optimal action of the athlete at time $t$ can be computed by

$$\alpha_t^* \;=\; \underset{\alpha_t}{argmax}\;\; p(\boldsymbol{H}_{s:t} \mid \alpha_t) \tag{13}$$

Note that Eq. (10), (11), and (13) can be efficiently computed using the forward-backward algorithm [21]. The parameters of the Hidden Markov Model, i.e. $A_{ij}^a$, $\boldsymbol{\Pi}_i^a$, $\boldsymbol{\Sigma}_i^a$, and the initial distribution, can be learned using the Baum-Welch (EM) algorithm [21].

### 4.3 Template Updating

Knowing the current action $\alpha_t^*$, we compute the most probable template sequence from time $s$ to $t$ given the observations represented by the HOG descriptors:

$$E_{s:t}^* \;=\; \underset{E_{s:t}}{argmax}\;\; p(E_{s:t} \mid \boldsymbol{H}_{s:t}, \alpha_t^*) \tag{14}$$

We can use the Viterbi algorithm [21] to compute the most probable template sequence $E_{s:t}^*$.

To update the current template $E_t$, we simply set $E_t = E_t^*$. In other words, we use the last template of the most probable sequence as the template of time $t$.

## 5 Experimental Results

We tested our algorithm in soccer sequences [6] and hockey sequences [19]. For both sequences, we first manually crop a set of stabilized images and partition them into different groups according to the action of the player. These images will be used to learn the parameters of the Hidden Markov Model of each action. Fig. 3 shows some of the training images of both the hockey and soccer sequences.

For the hockey sequences, we partition the training images into 6 actions: skating left, skating right, skating in, skating out, skating left 45, and skating right 45. Note that the action of the player can not be determined by only the position and velocity of the tracking region because the camera is not stationary; often the player of interest remains centered in the image by the tracking camera. Then, we transform the training images to the HOG descriptors. To compute the HOG descriptors, we first convolve the images by a $5 \times 5$ low-pass Gaussian filter with $\sigma = 5.0$, and then divide the image into $5 \times 5$ grids. For each grid, we compute the 8 bins orientation histograms of the image gradient. Next, we perform the Baum-Welch algorithm to learn the parameters of the Hidden Markov Model of

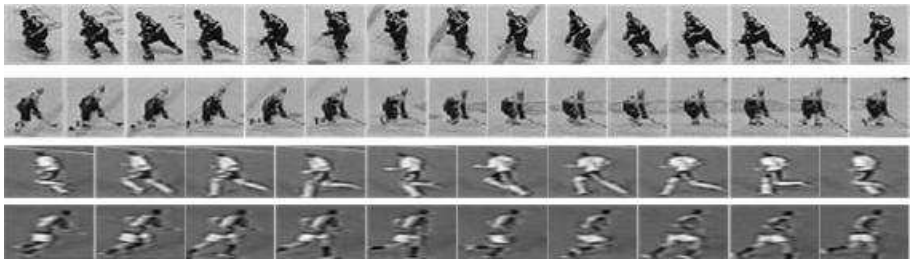

**Fig. 3. Examples of the training images**.

each actions, i.e., $A^a_{ij}$, $\boldsymbol{\Pi}^a_{ij}$, $\boldsymbol{\Sigma}^a_{ij}$, and the initial distribution. For each action, we assume that there are 10 possible templates, and we classify the action of player based on the previous 7 observations ($T = 7$). Tracking will start with a manually initialized tracking region and with 60 particles in our experiments.

We implement the entire system using Matlab. The processing time of the hockey sequence is about 2 seconds for each frame ($320 \times 240$ gray image). Fig. 4 and Fig. 5 show the experimental results of two hockey sequences. The upper part of these images shows the entire frame while the lower part of the images shows the previous tracking regions used to determine the player's action. In Fig. 4, the player first moves inward, turns left, moves outward, and finally turn a right. Since the camera is following the tracked player, the velocity of the tracking region is not consistent with the action of the player. However, our system can still track the entire body of the player and recognize his action based on the visual information. Fig. 5 gives an example when there is significant illumination changes (flashes) and partial occlusion. We can observe that our system is not influenced by the illumination changes because we rely on the shape information. Moreover, the system also works well under partial occlusion because it maintains an accurate appearance model of the tracked player. However, when two players cross over and one completely occludes the other, further techniques such as [3] are needed to solve the data association problem. In Fig. 4 frame 198, we can observe that the action recognizer make a mistake when the player moves toward the camera. This is because the contours of a person moving toward and away from the camera are very difficult to distinguish (even by human beings).

For the soccer sequences, we partition the training images into 8 actions: run in/out, run left, run right, run left 45, run right 45, walk in/out, walk right, and walk left (the categories are the same as [6]). We use the same way to compute the HOG descriptors. We also assume that there are 10 possible templates, and classify the player's action based on the previous 7 observations ($T = 7$).

The processing time of the soccer sequence is about 3 seconds for each frame ($720 \times 480$ gray image) due to the larger image size (we pre-compute the image gradients of the entire image and quantize them into orientation bins). Fig. 6 and Fig. 7 show the experimental results of two soccer sequences. Fig. 6 shows a soccer player running across a line on the field, and Fig. 7 shows a soccer player of very low resolution. In both cases, our tracker and action recognizer work well.

## 6 Conclusions and Future Work

This paper presents a system that tightly couples tracking and action recognition into an integrated system. In addition, the HOG descriptor representation not only provides robust and distinctive input features, but also implicitly centers the figure in the tracking region. Therefore, no further stabilization techniques need to be used. Experimental results in hockey and soccer sequences show that this system can track a single athlete and recognize his/her actions effectively.

### 6.1 Future Work

In the future, we plan to extend the current system to the following three directions. Firstly, the feature size of the HOG descriptor can be reduced using dimensionality reduction techniques such as Principal Component Analysis (PCA). The advantages of reducing the feature size are two-folds: (1) it can possibly increase the processing speed without loss of accuracy. (2) the number of training examples can be reduced because of the small feature vector. Secondly, it is possible to use a more sophisticated graphical model such as a hybrid Hierarchical Hidden Markov Model [17] instead of the current one. For example, we can treat the action as a random variable instead of a parameter. We can also introduce dependencies between the current action and the previous action, and dependencies between the action and the state of the tracker. Then, the action of the player will not only determined by the appearance/pose of the player, but also by the velocity and position of the player. Furthermore, after registering the player onto the hockey rink [18] or the soccer field, the action of the player can help to predict the velocity and position of the player. Finally, the action can be used as an additional cue in multi-target tracking when players occlude each other. Imagine that one player is moving left and another is moving right and they cross over. Since we know the action and the appearance change of the players, we can possibly accurately predict the next appearance of the player. Thus, the probability that two tracking windows get stuck into a single player can be reduced and the data association problem could be solved.

## 7 Acknowledgments

This work has been supported by grants from NSERC, the GEOIDE Network of Centres of Excellence and Honeywell Video Systems.

## References

1. S. Avidan. Ensemble tracking. In *CVPR*, volume 2, pages 494–501, 2005.
2. M.J. Black and A.D. Jepson. EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation. *IJCV*, 26(1):63–84, 1998.
3. Y. Cai, N. de Freitas, and J.J Little. Robust visual tracking for multiple targets. In *ECCV, to appear*, 2006.

4. D. Comaniciu, V. Ramesh, and P. Meer. Kernel-Based Object Tracking. *PAMI*, 25(5):564–575, 2003.
5. N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, volume 1, pages 886–893, 2005.
6. A.A. Efros, C. Breg, G. Mori, and J. Malik. Recognizing Action at a Distance. In *ICCV*, pages 726–733, 2003.
7. D.M. Gavrila. The Visual Analysis of Human Movement: A Survey. *CVIU*, 73(1):82–98, 1999.
8. J. Giebel, D.M. Gavrila, and C. Schnörr. A Bayesian Framework for Multi-cue 3D Object Tracking. In *ECCV*, pages 241–252, 2004.
9. J. Ho, K.C. Lee, M.H. Yang, and D. Kriegman. Visual Tracking Using Learned Linear Subspaces. In *CVPR*, volume 1, pages 782–789, 2004.
10. M. Isard and A. Blake. CONDENSATION–Conditional Density Propagation for Visual Tracking. *IJCV*, 29(1):5–28, 1998.
11. A.D. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust online appearance models for visual tracking. *PAMI*, 25(10):1296–1311, 2003.
12. K.C. Lee, J. Ho, M.H. Yang, and D. Kriegman. Visual tracking and recognition using probabilistic appearance manifolds. *CVIU*, 99:303–331, 2005.
13. J. Lim, D. Ross, R.S. Lin, and M.H. Yang. Incremental Learning for Visual Tracking. In *NIPS*, 2004.
14. D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2):91–110, 2004.
15. K. Mikolajczyk, B. Leibe, and B. Schiele. Local features for object class recognition. In *ICCV*, pages 1792–1799, 2005.
16. K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *PAMI*, 27(10):1615–1630, 2005.
17. K.P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.
18. K. Okuma, J.J. Little, and D.G. Lowe. Automatic rectification of long image sequences. In *Proc. Asian Conference on Computer Vision (ACCV'04)*, 2004.
19. K. Okuma, A. Taleghani, N. de Freitas, J.J. Little, and D.G. Lowe. A Boosted Particle Filter: Multitarget Detection and Tracking. In *ECCV*, pages 28–39, 2004.
20. P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-Based Probabilistic Tracking. In *ECCV*, pages 661–675, 2002.
21. L.R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.
22. X. Wu. Templated-based Action Recognition: Classifying Hockey Players' Movement. Master's thesis, The University of British Columbia, 2005.
23. Y. Wu and T.S. Huang. Robust visual tracking by integrating multiple cues based on co-inference learning. *IJCV*, 58(1):55–71, 2004.
24. J. Yamato, J. Ohya, and K. Ishii. Recognizing Human Action in Time-Sequential Images using Hidden Markov Model. In *CVPR*, pages 379–385, 1992.

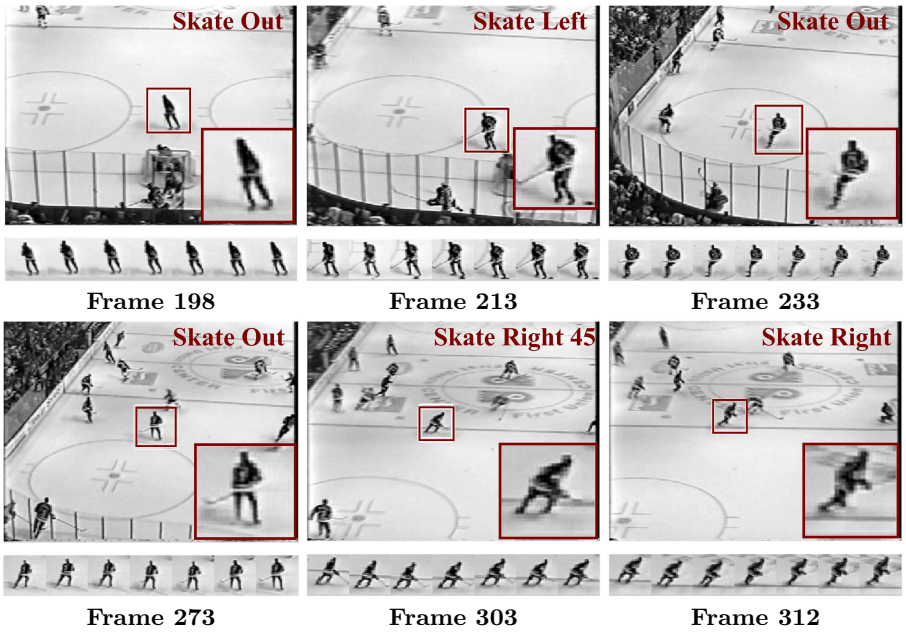


**Fig. 4. Experimental results in hockey sequence 1.**

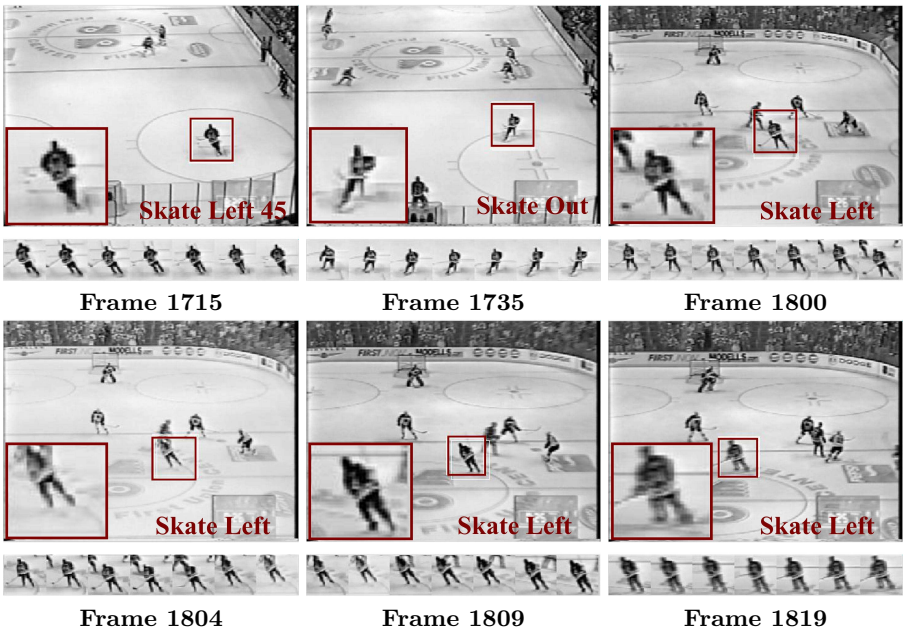


**Fig. 5. Experimental results in hockey sequence 2.**

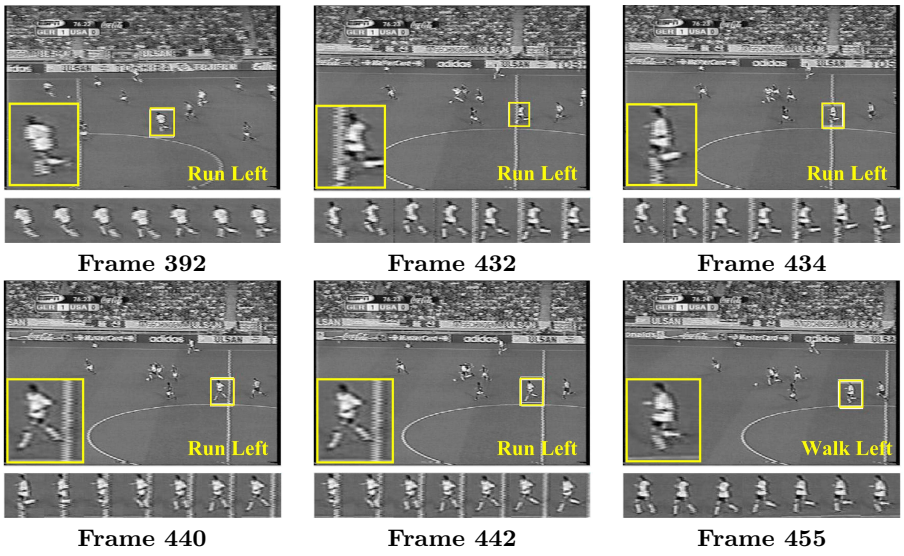


**Fig. 6. Experimental results in soccer sequence 1.**

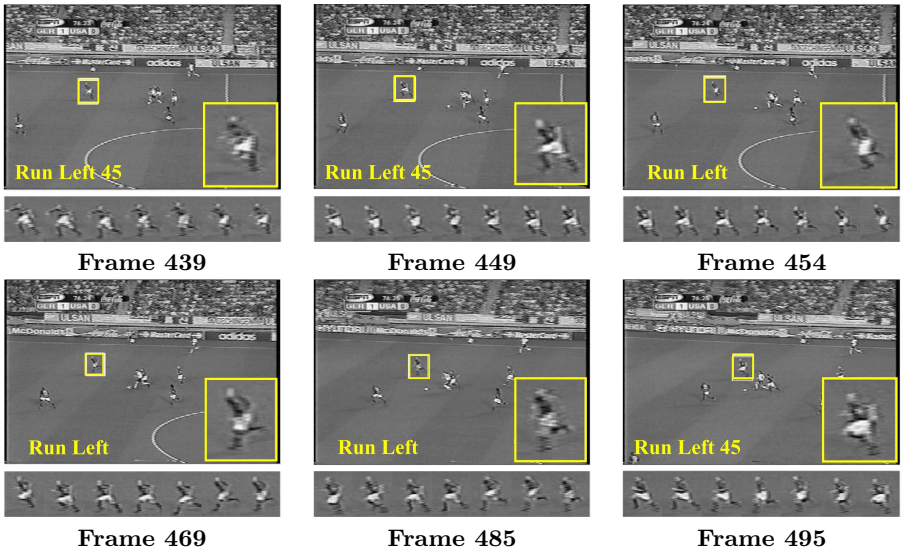


**Fig. 7. Experimental results in soccer sequence 2.**