ABSTRACT

Multiple player tracking is one of the main building blocks needed in a sports video analysis system. In an uncalibrated camera setting, robust multi-object tracking can be very difficult due to a number of reasons including the presence of noise, occlusion, fast camera motion, low-resolution image capture, varying viewpoints and illumination changes. To address the problem of multi-object tracking in sports videos, we go beyond the video frame domain and make use of information in a homography transform domain that is denoted the homography field domain. We propose a novel particle filter based tracking algorithm that uses both object appearance information (e.g. color and shape) in the image domain and cross-domain contextual information in the field domain to improve object tracking. In the field domain, the effect of fast camera motion is significantly alleviated since the underlying homography transform from each frame to the field domain can be accurately estimated. We use contextual trajectory information (intra-trajectory and inter-trajectory context) to further improve the prediction of object states within a particle filter framework. Here, intra-trajectory contextual information is based on history tracking results in the field domain, while inter-trajectory contextual information is extracted from a compiled trajectory dataset based on tracks computed from videos depicting the same sport. Experimental results on real world sports data show that our system is able to effectively and robustly track a variable number of targets regardless of background clutter, camera motion and frequent mutual occlusion between targets.

Index Terms— Tracking, Particle Filter, Cross-Domain, Contextual Information

1. INTRODUCTION

Tracking multiple targets has been of broad interest in the computer vision community for decades. A visual-based multi-target tracking system should be able to track a variable number of objects in a dynamic scene and maintain the correct identities of the targets regardless of occlusion and any other visual perturbations (e.g. camera motion, illumination changes, and object resolution). Extensive work has been done over the years [1, 2], as it is a very complicated and challenging problem. In this paper, we address the problem of robust multi-target tracking within sports videos (e.g. American football) by tracking players using hybrid information from both the image and field domains.

Human activity analysis has been established in the fields of security surveillance and military applications, but the sports world has been extremely under-serviced. Multiple player tracking is one of the main building blocks needed in an effective sports video analysis system. Knowing the location of each player on the field at each point of the game is crucial for sports experts (e.g. coaches, trainers, and sports analysts) to better understand complex player formations and trajectory patterns, which ultimately depict the effectiveness of their teams’ strategies as well as their opponents’. Being able to effectively track multiple players at one time can enable the development of reliable activity recognition and higher-level processing modules for sports video analysis. Such a tracking building block will have a positive impact on how sports experts analyze game footage, how content providers identify/display particular sports events and highlights accompanied with relevant advertisements, and how end users browse and query large collections of sports video.

Tracking players in the image domain is a difficult and challenging problem for several reasons: (1) Tracking players in sports is hard. Players on the same team have similar appearance information as shown in Fig. 1. This leads to the loss of a player’s track when he/she is moving near other players from the same team. (2) For sports video, it is always recorded in far-field view. Players are blurry and are often captured in low-resolution as exemplified in Fig. 1 (the red bounding box
Motion features (particles) with importance weights are resampled to generate a set of candidate samples and the weights of the samples are updated as Eq.(3). To avoid degeneracy, particles are resampled to generate a set of equally weighted particles by their importance weights.

\[ w_i^t = w_i^{t-1} \frac{p(z_t|x_t) p(x_t^i|z_t)}{q(x_t|x_{t-1}, z_{1:t})} \]  

Using the particle filter framework, we model the observation likelihood and the proposal distribution as follows. For the observation likelihood \( p(z_t|x_t) \), we follow [1] and adopt a multi-color observation model based on Hue-Saturation-Value (HSV) color histograms and a gradient-based shape model using Histograms of Oriented Gradients (HOG). We apply the Bhattacharyya similarity coefficient to define the distance between HSV and HOG histograms respectively. Moreover, we also divide up the tracked regions into two sub-regions (called particles) with importance weights \( w_i \). The candidate samples \( x_t^i \) are drawn from an importance distribution \( q(x_t|x_{t-1}, z_{1:t}) \) and the weights of the samples are updated as Eq.(3). To avoid degeneracy, particles are resampled to generate a set of equally weighted particles by their importance weights.

\[ q(x_t|x_{t-1}, z_{1:t}) = \alpha_1 p(x_t|x_{t-1}) + \alpha_2 p(x_t|x_{t-L:t-1}) + \alpha_3 p(x_t|x_{1:t-1}, T_{1:K}). \] 

2. OUR PROPOSED METHOD

2.1. Particle Filter

The particle filter [7] is a Bayesian sequential importance sampling technique for estimating the posterior distribution of state variables characterizing a dynamic system. It provides a convenient framework for estimating and propagating the posterior probability density function of state variables regardless of the underlying distribution, consisting of essentially two steps: prediction and update. Let \( x_t \) denote the state variable describing the parameters (e.g. appearance or motion features) of an object at time \( t \). The predicting distribution of \( x_t \) given all available observations \( z_{1:t-1} = \{z_1, z_2, \cdots, z_{t-1}\} \) up to time \( t-1 \), denoted by \( p(x_t|z_{1:t-1}) \), is recursively computed in (1).

\[ p(x_t|z_{1:t-1}) = \int p(x_t|x_{t-1}) p(x_{t-1}|z_{1:t-1}) dx_{t-1} \]  

At time \( t \), the observation \( z_t \) is available and the state vector is updated using Bayes rule, as in (2), where \( p(z_t|x_t) \) denotes the observation likelihood.

\[ p(x_t|z_{1:t}) = \frac{p(z_t|x_t)p(x_t|z_{1:t-1})}{p(z_t|z_{1:t-1})} \]  

In the particle filter framework, the posterior \( p(x_t|z_{1:t}) \) is approximated by a finite set of \( N \) samples \( \{x_t^i\}_{i=1}^N \) (called particles) with importance weights \( w_i \). The candidate samples \( x_t^i \) are drawn from an importance distribution \( q(x_t|x_{t-1}, z_{1:t}) \) and the weights of the samples are updated as Eq.(3). To avoid degeneracy, particles are resampled to generate a set of equally weighted particles by their importance weights.

\[ w_i^t = w_i^{t-1} \frac{p(z_t|x_t) p(x_t^i|z_t)}{q(x_t|x_{t-1}, z_{1:t})} \] 

rules, players in different video clips have similar trajectories as shown in Fig. 2. This demonstrates that using prior player trajectories (e.g. from a trajectory dataset) can help improve player tracking. Therefore, we attempt to implement a robust multi-object tracking system using cross-domain contextual information from both the field and image domains. In our algorithm, we employ the particle filter framework [6] to guide the tracking process. The cross-domain contextual information is integrated into the framework and acts as a guide for particle propagation and proposal.
To decide the values of $\alpha_1$, $\alpha_2$, and $\alpha_3$, we can use a cross-validation set. For simplicity, $\alpha_1$, $\alpha_2$, and $\alpha_3$ are equal and set to be 1/3 by experience in our experiments.

2.2. Intra-trajectory Contextual Information

For a tracked object from frame 1 to $t-1$, we obtain $t-1$ points: $\{p_1, p_2, \cdots, p_{t-1}\}$, which correspond to a short trajectory denoted as $T_0$. Our aim is to predict the next state at time $t$ using the previous states in a non-trivial data-driven fashion. As shown in Fig. 2, for each object, its previous states can help to predict its next state in the field domain. For simplicity, we just consider the most recent $L$ points in the trajectory to predict the state at time $t$. To obtain robust intra-trajectory information, we adopt $p_{i-L}$ as the start point, and all other more current points to define the difference as $\nabla p_i = (p_{i-L+1} - p_{i-L})/l$, where $\nabla p_i$ is also denoted as $\nabla p_i = (\nabla x_i, \nabla y_i)$, $l = 1, 2, \cdots, L$. In this way, given $\nabla p_{1:L-1}$, the probability of $p_L$ is defined as:

$$p(\nabla p_L | \nabla p_{1:L-1}) = \frac{e^{-\frac{1}{2}(\nabla p_L - u \nabla p_i)^T \Sigma^{-1} (\nabla p_L - u \nabla p_i)}}{2\pi |\Sigma|^2}$$  \hspace{1cm} (5)

Here $\Sigma$ is assumed to be diagonal matrix. To consider the temporal information, each $\nabla p_i$ is weighted with $\lambda_l$ defined as $\lambda_l = e^{-\frac{1}{2}(l-1)^2}$. Based on the weight $\lambda_1$, $u \nabla p_i$ and $\Sigma$ are defined as $u \nabla p_i = \sum_{l=1}^{L-1} \lambda_l \nabla p_i$ and $\Sigma = diag(\delta x_1, \delta y_1)$, where $\delta x_1 = (\sum_{l=1}^{L-1} \lambda_l)^{-1} - \sum_{l=1}^{L-1} \lambda_l^2$, and $\delta y_1$ has the same form. Finally, $p(x_t | x_{1:L-1})$ in Eq.(4) is defined as $p(x_t | x_{1:L-1}) = p(\nabla p_L | \nabla p_{1:L-1})$.

2.3. Inter-trajectory Contextual Information

Given the dataset introduced in Section 3.1, for the short trajectory $T_0$, we can obtain its $K$ nearest neighbors by use of dynamic time warping (DTW) [8], and the $K$ trajectories are denoted as $T_{1:K}$. For each $T_k$, $k = 1, \ldots, K$, we calculate the Euclidean distance between its points and $p_{t-1}$, and select the point $p_s$ with the smallest distance. Then, we select $L$ points from the point $p_s$ to $p_{t+L-1}$ in trajectory $T_k$ to obtain $p_k(\nabla p_i | \nabla p_{1:L-1})$ as the same as Eq.(5), where $\nabla p_i = p_i - p_{t-1}$, and $p_i$ is a certain point in field domain. Given $T_0$ and $T_{1:K}$, the probability of $\nabla p_i$ for each point $p_i$ in field domain is defined as:

$$p(\nabla p_i | T_0, T_{1:K}) = \sum_{k=1}^{K} \eta_k p_k(\nabla p_i | \nabla p_{1:L-1})$$  \hspace{1cm} (6)

where $\eta_k$ is the weight of the $k$-th trajectory and is set to be $\eta_k = \exp\left(-\frac{1}{2} Dist(T_k, T_0)^2 - u_0^2\right)$. The $Dist(T_k, T_0)$ is the distance between two trajectories, and $u_0$ and $\delta_0$ are obtained from the dataset. For each trajectory in the database, we can obtain its $K$ nearest neighbors, and calculate their distances. Then, based on all the distances, $u_0$ and $\delta_0$ can be obtained.

Based on $T_0$ and the $K$ nearest neighbors, $p(x_t | x_{1:t-1}, T_{1:K})$ in Eq.(4) is defined as $p(x_t | x_{1:t-1}, T_{1:K}) = p(\nabla p_i | T_0, T_{1:K})$.

This inter-trajectory contextual information is useful and effective to improve the object tracking, because the players in different video clips have similar trajectories as shown in Fig. 2. For a trajectory $T_0$, if there is no similar trajectory in the dataset, the $K$ nearest neighbors have very small weights $\eta_k$ as shown in Eq.(6). As a result, the probability $p(\nabla p_i | T_0, T_{1:K})$ is very small, and no useful inter-trajectory contextual information can be exploited. However, this happens rarely if the dataset is large-scale.

3. EXPERIMENTAL RESULTS

3.1. Dataset and Implementation Details

Our dataset contains 93 low-resolution videos of different football plays from 10 different teams, each around 400 frames long. Each video contains footage of a single football play shot from a PTZ camera with a sideline view high above the field. Fig. 1 depicts a typical view from this camera. The dataset is very complex. For each team, there are different background colors and environments as shown in Fig. 2. Every video is pre-processed to register frames to an overhead model of the football field using the method described in [9], thereby enabling us to determine players’ locations in football field coordinates.

It is time-consuming to build the database manually. Therefore, we implement a simple method that does not make use of inter-trajectory context information and adopt interactive object tracking. For each video clip, we track 8 to 13 players per frame. To evaluate the performance of the proposed tracking approach, we randomly select 5 video sequences as the testing set, and the rest are used for building the database. For the testing video clips, we create a tracking ground truth bounding box of the target in each frame for quantitative evaluation by manually annotating the data.

To evaluate the performance of our tracker, we use a score based on the PASCAL challenge object detection score: Given the detected bounding box $ROI_D$ and the ground truth bounding box $ROI_GT$, the overlap score evaluates as $score = area(ROI_D \cap ROI_GT)/area(ROI_D \cup ROI_GT)$. For each track, we get the average score. Then, we average these scores to obtain the evaluation score for the video. We compare our method with two state-of-the-art visual trackers for sports video analysis [1, 10]. For the baselines, we use publicly available code and adopt the same parameters as the authors.

3.2. Results and Analysis

Fig. 3 shows the probability map of intra-trajectory and inter-trajectory contextual information for a short trajectory in red. The pixel with high probability may be the next position.
of state. Based on the probability map, we can confirm the contextual information is effective to help predict the state. Moreover, the standard deviation in x coordinate is higher, as players are more likely to run straight forward. The quantitative results are summarized in Table 1. This table gives the average tracking scores of each approach in five sequences, and our method achieves more than 30% improvement. We also show the tracking results for the three trackers in Fig. 4. From the results we can see that although the traditional tracking approaches cannot track the players in American football well, our proposed method can track the players robustly and stably. That is because there is not enough appearance information in the image domain for methods [1, 10]. However, the cross-domain contextual information is effective to improve object tracking.

4. CONCLUSION

In this paper, we propose a novel and effective method to track multi-players in low-resolution videos of American football by use of cross-domain context information. Because the camera motion is eliminated in field domain, object intra-trajectory context information and inter-trajectory context information are helpful to predict the players states. Experimental results on many real-world challenging video clips demonstrate our proposed method is effective and useful to improve the multi-object tracking performance. Our cross-domain tracker is generic, and can also be used in other fields, such as video surveillance.

5. REFERENCES


