Tracking Jockeys in a Cluttered Environment with Group Dynamics

Mohammad Hedayati
hhedayat@waikato.ac.nz
School of Engineering, University of Waikato
Hamilton, Waikato, New Zealand

Michael J. Cree
m.cree@ieee.org
School of Engineering, University of Waikato
Hamilton, Waikato, New Zealand

Jonathan B. Scott
jonathan.scott@waikato.ac.nz
School of Engineering, University of Waikato
Hamilton, Waikato, New Zealand

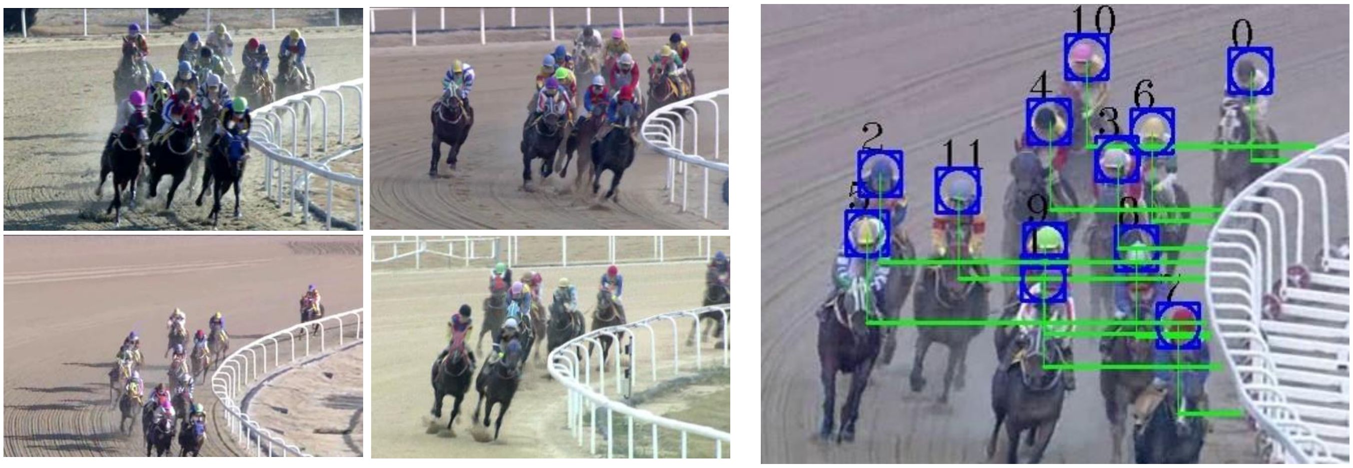

Figure 1: This project aims to automatically detect and track the jockeys around the turning point of the horse race. The turning point is where the heading direction of jockeys is starting to changes. The sample of turning points are shown on the left side of the figure and the final output of tracking is shown in the right image.

ABSTRACT

This project aims to detect and track jockeys at the turning point of the horse races. The detection and tracking of the objects is a very challenging task in a crowded environment such as horse racing due to occlusion. However, in the horse race, the jockeys follow each other's paths and move as a slowly changing group. This group dynamic gives an important cue to approximate the location of obscured jockeys. This paper proposes a novel approach to handle occlusion by the integration of the group dynamic into jockeys tracking framework. The experimental result shows the effect of group dynamics on the tracking performance against partial and full occlusions.

CCS CONCEPTS

• **Computing methodologies → Computer vision**; **Video summarization**; **Object detection**; **Tracking**; *Scene understanding*; *Activity recognition and understanding*.

KEYWORDS

video summarization, sport analysis, horse racing, group dynamics

ACM Reference Format:
Mohammad Hedayati, Michael J. Cree, and Jonathan B. Scott. 2019. Tracking Jockeys in a Cluttered Environment with Group Dynamics. In *2nd International Workshop on Multimedia Content Analysis in Sports (MMSports '19), October 25, 2019, Nice, France.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3347318.3355518

1 INTRODUCTION

Today sports analysis systems draw the attention of many commercial entities and are providing many opportunities for computer vision researchers to study and develop automated sports analysis systems. The primary objective of sports video analysis is to interpret low-level visual features to high-level semantics that can be understood by the end user. To interpret low-level visual features, the use of *prior knowledge* is needed. Sport events have a well-defined content structure with predefined regulations which are known to the audience in advance [23]. The prior knowledge

MMSports '19, October 25, 2019, Nice, France

ACM ISBN 978-1-4503-6911-4/19/10…$15.00
https://doi.org/10.1145/3347318.3355518

can be grouped into two categories: production knowledge and domain knowledge [24]. Production knowledge refers to the information gained from the video production, such as the camera angle and shot type (e.g. close view shot, far view shot). In contrast, the domain knowledge refers to the structure of the game or the property of the key event in the particular sport (e.g. turning point in horse racing; penalty or corner in soccer). Moreover, most sports analysis systems are rule based. These rules are obtain based on prior knowledge and they are specific to the sport in question.

The complexity of a sports video analysis system depends on the semantic level of interpretation which categorizes into (1) event detection and (2) high-level analysis [10, 33]. Event detection algorithms are designed to extract a particular event from the sport video, such as a penalty or goals. A high-level analysis system usually deals with extracting high-level semantics, such as contenders' movement, player skill and team strategy. High-level analysis is mostly involved with object detection and tracking. The detection and tracking of the objects is very challenging task in a crowded environment such as horse racing due to occlusion. Occlusion is a fundamental problem in computer vision that can significantly impair performance of object tracking and detection. In cluttered environments, objects might fully or partially be occluded by other obstacles. These obstacles can be the background elements or other tracked objects [27, 35].

Here we proposed a domain-specific tracking framework to tackle the occlusion problem in horse racing. In horse racing, jockeys race around a circular track and the camera typically follows them. Hence, the jockeys relative to each other tend to move as a slowly changing group. This homogeneous characteristic of jockeys is useful to continually track jockeys under partial and full occlusion. This group property, which is called the group dynamic in this paper, is thoroughly investigated in physiological, the physical and social studies [12, 16, 32]. This paper proposes a novel solution to handle the occlusion problem by integrating the group dynamic into the jockey tracking framework.

The remainder of this paper is organized as follows: Related work and the challenges are explored in section 2. In section 3 we propose our framework for automated analysis of horse race videos. Section 4 examines the performance of the proposed system and finally section 5 has closing remarks.

2 BACKGROUND AND PROBLEM STATEMENT

2.1 Problem of Tracking in Occlusion

Over past the few decades various approaches have been proposed for object tracking [1, 5, 8, 14, 18, 19, 21, 34]. Among them the mean-shift and tracking-by-detection have proven to be reliable in a wide range of applications. Mean-shift [1, 5, 8] is a non-parametric technique for finding the mode of a probability density function by using gradient ascent to find the local maxima of a likelihood distribution. The mean-shift based trackers are prone to failure, when 1) the object and background have similar features causing the gradient ascent to get stuck in local maxima, and 2) when the object is completely or partially occluded, the object likelihood is reduced leading to convergence to the wrong point. Recently the tracking-by-detection algorithm has become popular for object tracking [14, 17–19, 21, 25, 34]. The methodology behind these models are similar to the discriminative object detection. Given an initial object location, the goal of tracking-by-detection is to train on-line a classifier to distinguish the tracked object from the background. In each frame, the classifier is updated with the new samples. So that, at frame f, the sampling space can be written as $\{x_{\underline{1}}^{+}, x_{\underline{2}}^{+}, \ldots, x_{\underline{f}}^{+}, x_{\underline{1}}^{-}, x_{\underline{2}}^{-}, \ldots, x_{\underline{f}}^{-}\}$, where the x^{+} and x^{-} are the positive and negative samples and underline numbers are the frame index . However, under occlusion each training update can introduce errors which may lead to tracking drift.

2.2 Occlusion Handling and Data Association

Typically, occlusion handling consists of two main steps: occlusion detection and object retrieval. There are a number of algorithms reported for occlusion detection [9, 30]. To make a clear distinction between the target and occluders, Pan and Hu [30] proposed an algorithm that continuously analyzed the occlusion situation using spatiotemporal information. Dong et al. [9] proposed a tracking-by-detection algorithm which progressively checks the occlusion of the target patch using two distance measures. The first measure is the occlusion threshold which estimates the distance between the target patch and its surrounding patches, and the second measure is defined as target distance, which is the distance between the target patch and patches in the classifier-pool.

Object retrieval is a challenging step in occlusion handling. Although object retrieval is straightforward when a single object appears in the environment, the complexity of the object reacquiring is increased if multiple moving objects need to be tracked. Data association (DA) techniques are the most effective approaches to handle the occlusion in crowded environments. DA can be viewed as a multi target-management system that maintains the multiple target identities over the course of tracking [2, 28]. In general, DA is the process of matching information of newly observed objects (measurement) with previously observed information (state) [2]. This procedure estimates the region that the targets expected to be seen at the time t with regards to the past states x_{t-1}. This region is called the target *gate* and indicates the valid measurements to contribute to the association process [4, 7, 11, 13, 26]. DA techniques handle the occlusion problem by reacquiring lost objects after occlusion using object detection algorithms. Obviously, the data association performance greatly depends on the quality of the detector, and also the type of objects that need to be tracked.

3 GROUP DYNAMIC TRACKING

The approximate distance of the jockeys from the fence at the turning point of the race is the key factor for evaluating the performance of jockeys (see Figure 1). This paper is focused on the tracking stage. To have a fully automated system, first, we need to extract the *frame sequences around the turning points* (turning segment), then detect jockeys at each segment and finally track and extract their trajectory. The approach for extracting turning segments is reported in our earlier work [15]. The detection of jockeys is accomplished by locating of the jockey's cap with using well-known histogram of gradient (HOG) object detection framework [6]. The cap is selected as the main feature of the detection algorithm due to three main reasons: firstly, caps have a rigid and unique structure, secondly, the occlusion between the jockeys' caps is much less than the other parts of jockeys, and thirdly, the colors of the caps are usually different to the others, which reduces the uncertainty of tracking under partial occlusion.

The backbone of the proposed tracking model is based on the assumption that when objects are in a group, they tend to move relative to each other following a similar motion pattern. This group dynamic often gives an important cue to approximate the location of an object, especially, when local information is poor or abrupt.

The tracking framework can be separated into three modules: sampling, localization and data association. The aims of the sampling module are to find the sample points that belong to the same object and check the occlusion situation. For localization, two different strategies are used, namely, object-based and group-based localization. Object-based localization is applied when the sample points correctly represent the local object property. The group based localization is used when the object local information does not properly represent the object, mainly due to occlusion and background clutter. In general data association is consider as a multi-target management system to maintain multiple jockeys' identities over time and initialize the tracking. The main task of data association in our proposed framework is to automatically *initialize the jockeys tracking* based on detection response.

3.1 Sampling

To reduced the feature ambiguities, the object features are sampled from three different levels 1) point, 2) object and 3) group. The sampling at point level rests on the assumption that the distributed points over a cap template have the similar motion and color distribution as the cap itself. Accordingly, the combination of these sample points are used to create the object level representation. The group level information is built based on the assumption that the jockeys should follow a similar movement pattern. Therefore the location of any jockeys in the group can be estimated by finding its relative motion to the motion of the group. The object template or tracking windows (w) is represented by a rectangular patch around the jockeys cap. The following features are constructed from the object template,

- *Point level motion cues*, $U_p = (u_{p,x}, u_{p,y})$, are estimated using the iterative pyramids Lucas-Kanade method [3] for all sample points p.
- *Object motion model*, $U_o = (u_{(o,x)}, u_{(o,y)})$, refers to the template displacement.
- *Point level color cues* (H_p) refers to the color distribution of 15×15 rectangular patches around sample points. H_p are estimated from histograms of hue and saturation channels in HSV color space.
- *Object level color cues* (H_o) refer to the color distribution of the tracking window. H_o is created by accumulating all n points level color cue by,

$$H_o = \sum_i^n H_{(p,i)} \tag{1}$$

- *Group motion model*, $U_g = (u_{(g,x)}, u_{(g,y)})$, is estimated by taking the average motion models of all m objects using,

$$U_g = \frac{\sum_i^m U_{(o,i)}}{m} \tag{2}$$

- *Object relative speed*, $U_r = (u_{(r,x)}, u_{(r,y)})$, refers to the relative speed which is element wise division of individual object to the group motion, viz

$$U_r = \frac{U_o}{U_g}. \tag{3}$$

- *Similarity measures*, s, indicates how likely sample points p are generated from object o. To measure this similarity the histogram intersection [31] is used. It is especially suited to comparing histograms for recognition in our case, because it does not require the accurate separation of the object from its background or occluding objects in the foreground. Using the object colour distribution (H_o) and point level colour cue (H_p), the similarity measures for each sample point, i, is estimated by,

$$s = \sum_j \min(H_{(p,i)}(j), H_o(j)), \tag{4}$$

where j is the bin number of the histogram.
- *Background motion*, $U_b = (u_{(b,x)}, u_{(b,y)})$, refers to the dominate motion in the frame.

To reduce uncertainty in localization, three type of filters are applied on the above features, namely, cross-validation filter, motion filter and ambiguity filter.

Cross validation filter applies the forward-backwards error estimation [20] to find the stability of motion cues. With the sample point p at frame I and its corresponding location p' in the frame $I + 1$, the backwards flow of point p' to the frame I is computed. The forward-backwards error ε_{FB} of a point p is defined as the Euclidean distance between the original point and the forward-backward prediction. In the filtering stage the points are removed if their forward–backwards error is larger than some threshold (α), that is

$$p = \begin{cases} 0 & \varepsilon_{\text{FB}} \geq \alpha \\ 1 & \text{elsewhere} \end{cases}. \tag{5}$$

Motion filter uses binary classifier to remove the points that are more likely generated from the background element. Knowing the background motion (U_b), object motion (U_o) and the motion of sample points (U_p), the motion filter is constructed by,

$$p = \begin{cases} 0 & d(U_p, U_b) < d(U_p, U_o) \\ 1 & \text{elsewhere} \end{cases}, \tag{6}$$

where d is the Euclidean distance function.

Ambiguity filter is applicable when multiple tracked objects overlap each other, Fig. 2 a. In this case, some sample points that belong to one object might move to the other and eventually causes tracking drift, Fig. 2 b. To avoid this situation the similarity measures (Equation 4) of the points inside the overlap region are obtained with respect to all k occluded objects. Assuming the sample point originated from the object n, the point is removed if the similarity measures to its own object template $H_{(o,n)}$ is not the highest among all k objects. Eventually, if more than 50% of sample points are removed by the ambiguity filter it specifies that the object is occluded with other tracked objects, hence the group-based localization should be used. The effect of the ambiguity filter is illustrated in Fig. 2.

3.2 Localization

During occlusion, the local object information does not properly represent the true properties of objects. To handle occlusion the proposed system uses two types of localization methods named as object-based and group based localization. Object-based localization is used when the sample points correctly represent the object appearance and motion information. The group based localization is applied when the local information does not properly represent the object, mainly due to occlusion and background clutter.

3.2.1 *Object-based localization.*

Object-based localization determines the new location of tracking windows based on the features that are extracted from point and object level. In this strategy, all points are weighted with respect to their similarity measures s, and then the tracking windows shifts by estimating the center of the sample point distribution as,

$$M^t(x, y) = \frac{\sum_i^n s_i p_i(x, y)}{n} \tag{7}$$

where n is number of sample points and $M^t(x, y)$ is the new center of the tracking window.

Template updating is responsible to update the object appearance model and create new sample points. Wherever the template is valid, it is assumed that the object is not occluded by background element or other tracked object, so that object appearance model can be updated by,

$$H_o^t = H_o^{t-1} + \sum_i^n H_{(p,i)} \tag{8}$$

H_o^{t-1} is last object level color cues and n is the number of sample points

3.2.2 *Group-based localization.*

Group-based localization approximates the location of occluded objects using the group dynamic. Two measures are used in here 1) Group motion that is determined by Equation 2 and 2) object relative speed which obtains by Equation 3. From these two values, the new location of the object is determined by moving the tracking window by,

$$M^t(x, y) = M^{t-1}(x + u_{(r,x)}u_{(g,x)}, y + u_{(r,y)}u_{(g,y)})) \tag{9}$$

$M^{t-1}$and M^t are the old and new location of the center of tracking window respectively.

3.3 Data Association

The result of object detection algorithm often noisy and may include many false detections. Therefore the detection result cannot be used directly for tracking initialization. We employ data association to initialize tracking from the noisy and uncertain detection respond. Let the trajectories of n jockeys' caps at time t are represented by the sequence of states, $\mathbf{X_t} = \{\mathbf{x_t^1}, \dots, \mathbf{x_t^n}\}$, and the measurements O be the output of the cap detection algorithm at time t, $\mathbf{O_t} = \{\mathbf{o_t^1}, \dots, \mathbf{o_t^m}\}$. Our association task is to assign n tracks to m new detected caps with the capability of initiating new jockeys and terminating false trajectories. This problem can be simplified by building the assignment matrix, A, given by,

$$A = \left[\begin{array}{c|c} C_{m,n} & B_{m,m} \\ \hline T_{n,n} & 0_{m,n} \end{array}\right] =$$

$$\left[\begin{array}{cccc|cccc} c_{1,1} & c_{1,2} & \cdots & c_{1,n} & b_{1,(n+1)} & 0 & \cdots & 0 \\ c_{2,1} & c_{2,2} & \cdots & c_{2,n} & 0 & b_{2,(n+2)} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ c_{m,1} & c_{m,2} & \cdots & c_{m,n} & 0 & 0 & \cdots & b_{m,(n+m)} \\ \hline t_{(m+1),1} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & t_{(m+2),2} & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & t_{(m+n),n} & 0 & 0 & \cdots & 0 \end{array}\right] \tag{10}$$

where the columns of the matrix A correspond to the tracks, and the rows to possible jockeys. Each element of the assignment matrix is equal to one association hypothesis. The sub matrices C , B and T are responsible for assigning n tracks to m measurements, initiating new and terminating ongoing trajectories. Let i be the row and j be column of sub matrices in A then the association hypothesis of these matrices are defined by

(1) Matrix, C, is the association hypothesis to assign objects to the track, namely

$$c_{i,j} = \begin{cases} 1 + \frac{1}{d_{i,j}} & \text{if} \quad o_i < r \\ 0 & \text{elsewhere,} \end{cases} \tag{11}$$

where d is the Euclidean distance between the center of the detected cap and tracking window and r is radius of searching area (Gate).

(2) Matrix, B, is the association hypothesis to initialized new *potential tracks*, given by

$$b_{i,(n+j)} = \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{elsewhere,} \end{cases} \tag{12}$$

Here the term *potential new track* is used because we delay the decision about the birth of jockey until enough observations are collected from the association hypotheses.

(3) Matrix, T, is the association hypothesis to terminate tracks, and is

$$t_{(m+i),j} = \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{elsewhere.} \end{cases} \tag{13}$$

Once the assignment matrix is built, the data association is treated as an assignment problem. To solve the assignment problem we used Katta and Murty algorithms [22]. With every newly detected cap, the tracks states are updated according to three characteristics, namely (1) potential track when there is not sufficient evidence to prove the track belongs to the true jockeys, (2) confirmed track is a track that belongs to a valid jockeys, and (3) false track is a track that comes from false alarm and should be deleted. If the detected cap is assigned into the same potential track three times over a five frame period, it is considered the new jockey has arrived into a scene.

4 EXPERIMENTAL RESULT

To evaluate the final tracking system we conduct two different experiments. The first step evaluates the performance of the tracking and the data association together. In this experiment, we build the ground truth data from ten turning segments and then annotate

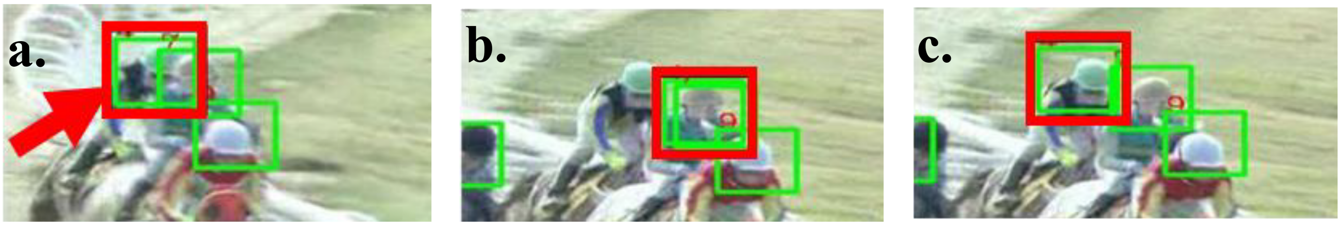

Figure 2: The effect of the ambiguity filter. (a) object of interest before occlusion. (b) tracking result without the ambiguity filter, and (c) tracking result with the ambiguity filter after occlusion.

Table 1: Tracking Performance for Ten Selected Videos

ID	Total jockeys	Detected Cap	Hit	Miss	CTR	Number of Frames
h1	11	10	10	0	1	25
h2	12	12	11	1	0.91	50
h3	11	10	8	2	0.8	75
h4	12	12	12	0	1	50
h5	13	12	12	0	1	25
h6	12	12	12	0	1	25
h7	12	12	11	1	0.91	50
h8	12	10	9	1	0.90	25
h9	12	12	12	0	1	75
h10	12	12	12	0	1	25

the jockeys' caps at the first and at the end of each segment. The resolution test videos are 800×600 with frame rate of 25 frame per second. The tracking is considered successful if the center of the tracking box, matches the ground truth at the end of the tracking course. The performance of a tracking algorithm is measured by calculating the ratio of successfully tracked jockeys to the total number of *detected jockeys*. This measurement is called the correct tracking ratio (CTR). Hit indicates the total number of correct tracked jockeys and Miss is the number of unsuccessful tracking. As can be seen the result on Table 1, the overall tracking accuracy shows a promising result with average CTR of .94, wherein 9 out of 10 cases the correct tracking ratio is above 0.90.

Here we should emphasize that the final tracking performance in table 1 is based on detected cap rather than total jockeys. This is because, if the cap not being detected three times over a five frame sequence the tracking will not initialize. Therefore the initialization process is directly related to object detection rather than tracking.

The effect of group dynamic on tracking without data association can be seen in the second experiment. For this evaluation, we manually initialize tracking at first frame. Here two long horse race video footages (around 250 frames) are used and the tracking performance is estimated based on intersection over union (IoU) methodology [29]. The ground truth data was built manually by annotating the bounding box for each jockey at every tenth frame of the video. Furthermore, to investigate the effect of the group dynamic, we examine the output of the proposed model with kernel correlation filter (KCF) [18]. As it is shown in Figure 3 the overall performance of KCF is slightly better in comparison to the proposed model for *unocclouded objects*. This is mainly due to the localization strategy of the proposed tracking system. Localization of the proposed tracking system is constantly shifting between object-based and group-based. This shift causes a slight deviation of the tracking bounding box. The benefit of the group dynamic became visible when objects obscured by other objects. As it is shown in Figure 3, the KCF lost the track of objects 2, 6 and, 9 in the first video (V1), and object 6 in second video (V2).

5 CONCLUSION

In this work, we proposed a domain-specific framework to obtain the trajectory of jockeys from the turning segments of the horse race. The success of the proposed system is owed to the group movement property of horse races. This paper was focused on tracking the jockeys so that we used simple object detection to locate jockeys cap in the frame. For future work, we plan to apply deep convolution neural networks to detect multiple parts of jockeys to improve detection which consequently boosts data association and automatic jockeys initialization performance.

REFERENCES

[1] John G Allen, Richard YD Xu, and Jesse S Jin. 2004. Object tracking using camshift algorithm and multiple quantized feature spaces. In *Proceedings of the Pan-Sydney area workshop on Visual information processing*. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 3–7.

[2] M. Betke and Z. Wu. 2016. *Data Association for Multi-Object Visual Tracking.* Morgan & Claypool.

[3] Jean-Yves Bouguet. 2001. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation, Microprocessor Research Labs* 5, 1-10 (2001), 4.

[4] Yizheng Cai, Nando de Freitas, and James J Little. 2006. Robust visual tracking for multiple targets. In *European conference on computer vision*. Springer, Springer Berlin Heidelberg, Berlin, Heidelberg, 107–118.

[5] Dorin Comaniciu and Peter Meer. 1999. Mean Shift Analysis and Applications. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2 (ICCV '99)*. IEEE Computer Society, Washington, DC, USA, 1197–. http://dl.acm.org/citation.cfm?id=850924.851593

[6] Navneet Dalal and Bill Triggs. 2005. Histograms of Oriented Gradients for Human Detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01 (CVPR '05)*. IEEE Computer Society, Washington, DC, USA, 886–893. https://doi.org/10.1109/CVPR.2005.177

[7] Bao Dang, An Tran, Tien Dinh, and Thang Dinh. 2010. A real time player tracking system for broadcast tennis video. In *Asian Conference on Intelligent Information and Database Systems*. Springer, Springer Berlin Heidelberg, Berlin, Heidelberg, 105–113.

[8] Mandar Dixit and KS Venkatesh. 2009. Combining edge and color features for tracking partially occluded humans. In *Asian Conference on Computer Vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, 140–149.

[9] Xingping Dong, Jianbing Shen, Dajiang Yu, Wenguan Wang, Jianhong Liu, and Hua Huang. 2017. Occlusion-aware real-time object tracking. *IEEE Transactions on Multimedia* 19, 4 (2017), 763–771.

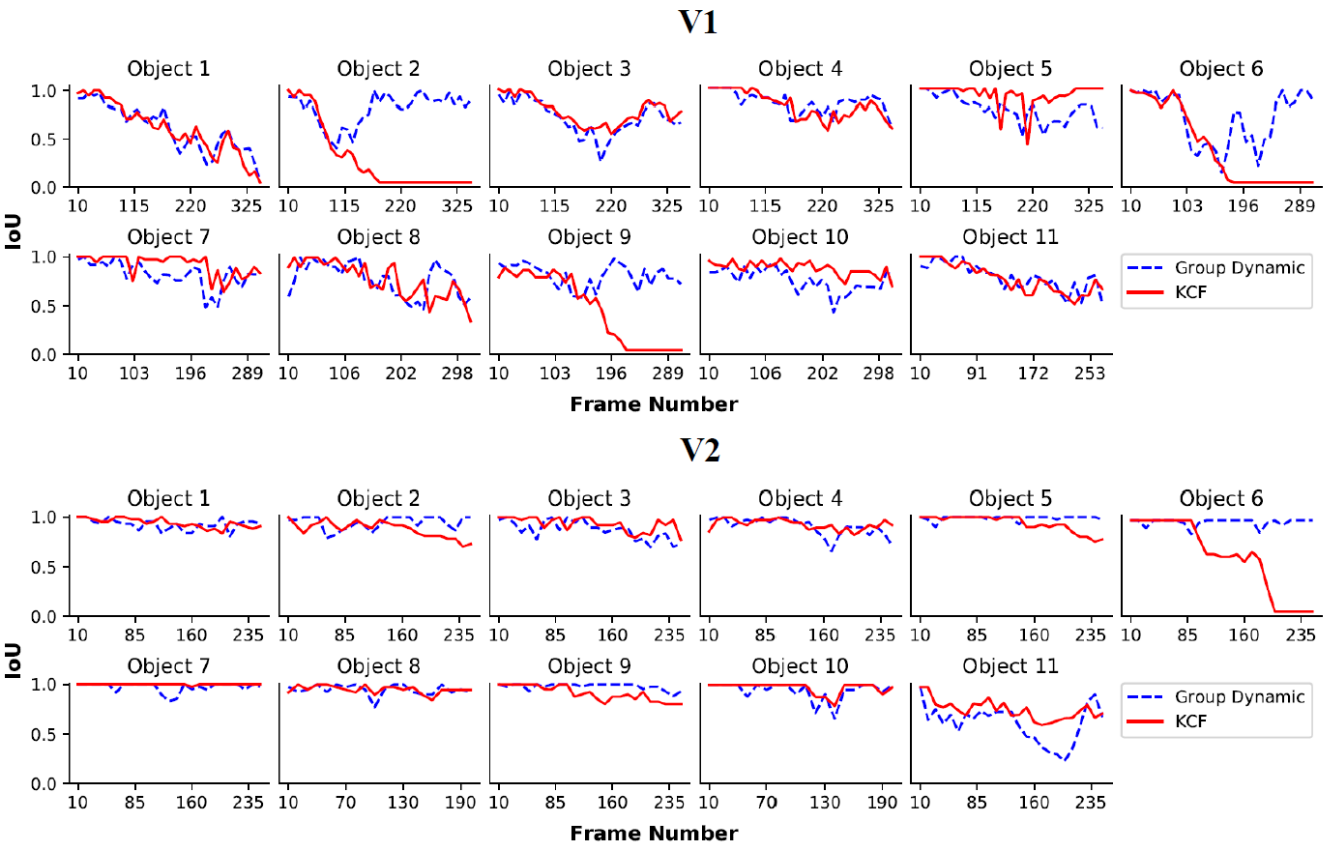

Figure 3: Tracking result of individual objects at every tenth frame for video V1 and V2. The performance of KCF is slightly better in comparison to the proposed model for *unocclouded objects*. The benefit of the group dynamic became visible when objects were obscured by other tracked objects. As it is shown in the graphs, the KCF lost the track of objects 2, 6 and, 9 in V1, and object 6 in V2.

[10] Tiziana D'Orazio and Marco Leo. 2010. A review of vision-based systems for soccer video analysis. *Pattern recognition* 43, 8 (2010), 2911–2926. https://doi.org/10.1016/j.patcog.2010.03.009

[11] Weina Ge and Robert T Collins. 2008. Multi-target Data Association by Tracklets with Unsupervised Parameter Estimation. In *In British Machine Vision Conference*, Vol. 2. Citeseer, 5.

[12] E. T. Hall. 1966. *The Hidden Dimension: Man's Use of Space in Public and Private.* The Bodley Head Ltd.

[13] Jungong Han, Dirk Farin, Peter H.N. de With, and Weilun Lao. 2005. Automatic Tracking Method for Sports Video Analysis. In *26th Symposium on Information Theory in the Benelux*. 309–316.

[14] Sam Hare, Stuart Golodetz, Amir Saffari, Vibhav Vineet, Ming-Ming Cheng, Stephen L Hicks, and Philip HS Torr. 2016. Struck: Structured output tracking with kernels. *IEEE transactions on pattern analysis and machine intelligence* 38, 10 (2016), 2096–2109.

[15] Mohammad Hedayati, Michael J Cree, and Jonathan Scott. 2016. Scene structure analysis for sprint sports. In *2016 International Conference on Image and Vision Computing New Zealand (IVCNZ)*. IEEE, 1–5.

[16] Dirk Helbing and Peter Molnar. 1995. Social force model for pedestrian dynamics. *Physical review E* 51, 5 (1995), 4282.

[17] David Held, Sebastian Thrun, and Silvio Savarese. 2016. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, 749–765.

[18] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. 2012. Exploiting the circulant structure of tracking-by-detection with kernels. In *European conference on computer vision*. Springer Berlin Heidelberg, Berlin, Heidelberg, 702–715.

[19] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. 2015. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 3 (2015), 583–596.

[20] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. 2010. Forward-backward error: Automatic detection of tracking failures. In *Pattern recognition (ICPR), 2010 20th international conference on*. IEEE Computer Society, Washington, DC, USA, 2756–2759.

[21] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. 2012. Tracking-Learning-Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 7 (July 2012), 1409–1422. https://doi.org/10.1109/TPAMI.2011.239

[22] Katta and Murty. 1968. An Algorithm for Ranking all the Assignments in Order of Increasing Cost. *Operations Research* 16, 3 (1968), 682–687.

[23] Yiannis Kompatsiaris, Bernard Merialdo, and Shiguo Lian. 2012. *TV content analysis: Techniques and applications.* CRC Press.

[24] Baoxin Li, James H. Errico, Hao Pan, and Ibrahim Sezan. 2004. Bridging the semantic gap in sports video retrieval and summarization. *Journal of Visual Communication and Image Representation* 15, 3 (2004), 393–424.

[25] Wei-Lwun Lu, J-A Ting, James J Little, and Kevin P Murphy. 2013. Learning to track and identify players from broadcast sports videos. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35, 7 (2013), 1704–1716.

[26] Wei-Lwun Lu, Jo-Anne Ting, James J Little, and Kevin P Murphy. 2013. Learning to track and identify players from broadcast sports videos. *IEEE transactions on pattern analysis and machine intelligence* 35, 7 (2013), 1704–1716.

[27] Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, Xiaowei Zhao, and Tae-Kyun Kim. 2014. Multiple object tracking: A literature review. *arXiv preprint arXiv:1409.7618* (2014).

[28] Dr Emilio Maggio and Dr Andrea Cavallaro. 2011. *Video Tracking: Theory and Practice* (1st ed.). Wiley Publishing.

[29] Anton Milan, Konrad Schindler, and Stefan Roth. 2013. Challenges of ground truth evaluation of multi-target tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 735–742.

[30] Jiyan Pan and Bo Hu. 2007. Robust occlusion handling in object tracking. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, IEEE, Washington, DC, USA, 1–8.

[31] Michael J Swain and Dana H Ballard. 1991. Color indexing. *International journal of computer vision* 7, 1 (1991), 11–32.

[32] John FA Taylor. 1958. *The psychology of perception: A philosophical examination of Gestalt theory and derivative theories of perception*. Vol. 55. 77–81 pages.

[33] Graham Thomas. 2011. Sports TV applications of computer vision. In *Visual Analysis of Humans*. Springer Berlin Heidelberg, Berlin, Heidelberg, 563–579.

[34] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. 2015. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 9 (2015), 1834–1848.

[35] Alper Yilmaz, Omar Javed, and Mubarak Shah. 2006. Object Tracking: A Survey. *ACM Comput. Surv.* 38, 4, Article 13 (Dec. 2006).