Effect of Contextual Information on Object Tracking

Mohammad Hedayati
School of Engineering
University of Waikato
Hamilton, New Zealand
mh267@students.waikato.ac.nz

Michael J. Cree
School of Engineering
University of Waikato
Hamilton, New Zealand
cree@waikato.ac.nz

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

***Abstract*—Local object information, such as the appearance and motion features of the object, are useful for object tracking in videos provided the object is not occluded by other elements in the scene. During occlusion, however, the local object information in the video frame does not properly represent the true properties of the object, which leads to tracking failure. We propose a framework that combines multiple cues including the local object information, the background characteristics and group motion dynamics to improve object tracking in challenging cluttered environments. The performance of the proposed tracking model is compared with the kernelised correlation filter (KCF) tracker. In the tested video sequences the proposed tracking model correctly tracked objects even when the KCF tracker failed because of occlusion and background noise.**

***Index Terms*—video analysis, object tracking, occlusion**

I. Introduction

Over the last few decades an enormous amount of study has been dedicated to object tracking [1]. Object tracking remains a challenging topic in computer vision due to problems caused by changes in size or pose of the object, noise produced by the image acquisition, variation of light, occlusion and background clutter [2, 3]. Moreover, the complexity of the tracking is increased if multiple moving objects are tracked. This is because locating targets and maintaining their identities through a video sequence is a highly challenging problem in crowded environments. Wu et al. [4] performed experiments to evaluate the performance of recent online tracking algorithms, and identified three important components that improve tracking performance. First, background information is necessary, mainly to separate background clutter from the object of interest. Second, local models are particularly useful when the appearance of the target has partially changed and third, the motion model is crucial for object tracking, especially when the motion of the target is abrupt.

In addition to above components when objects are in a groups, they tend to move relative to each other following similar motion pattern. This group dynamic often gives an important cue to approximate the location of object, especially, when local information is poor or abrupt. This research proposes a framework that combines the group dynamic with local object information to improve object tracking in challenging cluttered environments.

978-1-5386-4276-4/17/$31.00

II. Literature Review

Assuming the foreground pixels in the video represent the foreground object, primitive tracking systems used background subtraction approaches to separate the foreground from the background and tracking was then performed by enforcing spatial continuity using Kalman filtering [5, 6]. Colour-based models such as means shift [7] and particle filtering [8] also have achieved considerable success in many tracking applications. Particle tracking is a process of propagating the posteriori distribution of the reference target, according to a system dynamic model. Pérez et al. [9] and Nummiaro et al. [10] proposed two independent solutions that couple the colour information of objects with the dynamic model of the system.

Mean shift is a non-parametric technique for finding the mode of a probability density function by using gradient descent/ascent [7] to find the local minima/maxima of a distribution by iteratively descending/climbing the density gradients until the point of convergence has been found. In computer vision applications, the mean shift was originally employed by Comaniciu and Meer [11] for segmentation purposes and later Bradski [12] utilised the mean shift framework for tracking applications. The mean shift tracker model calculates the centroid of the colour probability distribution within its 2D tracking window, then moves the window centre to the centroid of distribution. Although the mean shift tracker gives reasonable accuracy in a wide range of environments, it is prone to failure, when 1) the object and background have similar features causing the gradient decent search to get stuck in local minima, and when 2) 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 [4]. 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. During tracking the initial sample space is continually updated and the classifier is retrained as result at the time instant, the sampling space can be written as $\{x_0^+, x_1^+, \ldots, x_t^+, x_0^-, x_1^-, \ldots, x_t^-\}$, where the x_t^+ and x_t^- are the positive and negative samples at time t. There are various classifiers already integrated into the tracking-by-detection framework. Support vector tracking [13] used the Support

Vector Machine (SVM) classifier to distinguished foreground motion from the background. Kalal et al. [14] proposed the long-term tracking task based on boosting classifier. The classifier is updated using all extracted appearances up to currant frame that passed the variance filter. Hare et al. [15] employs the structured Support Vector Machine (SVM) to directly link the target's location space with the training samples to reduced the training time. Kernelised Correlation Filters (KCF) tracker proposed by Henriques et al. [16] achieves the fastest and highest performance among the recent top-performing tracking-by-detection algorithms [17]. The key of KCF tracker is that the augmentation of negative samples are employed to enhance the discriminative ability of the track-by-detector scheme while exploring the structure of the circulant matrix [18] for high efficiency.

The reviewed tracking algorithms mostly focused on local object information to track interest objects. During occlusion, however, the local object information does not properly represent the true properties of the object, which leads to tracking failure. In contrast to these methods, we proposed a framework that combines contextual information with local object properties to improve the tracking in clutter environment.

III. Proposed Model

The proposed tracking model contains two main modules, namely point level processing and localisation (see Figure 1). The point processing block is based on the assumption that in reality the sample points are rarely independent and they are parts of bigger units, namely the objects that are being tracked [19], and therefore the sample points should have the same motion and similar colour distribution. Consequently, the location of the object can be estimated by tracking the points that sample from the same objects. The point processing block aims to find the best points from a noisy sample space by a series of filtering stages.

In object localisation, two different strategies are used, namely object based and group based localisation. Object based localisation is applied when the sample points correctly represent the local object motion and appearance. The group based localisation is applied when the local information does not properly represent the object, mainly due to occlusion and background clutter.

A. Feature Extraction

To overcome appearance ambiguities and to handle the occlusion, the object features are extracted from three sampling levels; point level, object level and group level. The definition of these features are as follows:

1) *Object template* (w) refers to the rectangular window around the object; it is also referred to as the tracking window.
2) *Point level motion cues* (U_p) are the flow of the sample points extracted from the object template where $U_p = (u_{p,x}, u_{p,y})$ are the motion cues for a given point p. Particularly, given the point $p = (p_x, p_y)$ on the selected template at frame I, we estimate its corresponding location $p' = (p_x + u_{p,x}, p_y + u_{p,y})$ in the frame $I + 1$ using the iterative pyramids Lucas-Kanade method [20]. The sample points are extracted using Shi and Tomasi corner detection [21].
3) *Point level colour cues* ($H_{p'}$) refers to the colour distribution of 15×15 rectangular patches around sample points. Point level colour cues are calculated from histogram of hue and saturation channels in HSV colour space.
4) *Object motion model*, $U_o = (u_{(o,x)}, u_{(o,y)})$, refers to the tracking window displacement. The object motion model is estimated by taking the average of all motion vector (J) at point level for the given object o by,

$$U_o = \frac{\sum_i^J U_{p,i}}{J} \tag{1}$$

5) *Object colour model* (H_o) refers to the colour distribution of the tracking window. The object colour model is calculated from the histogram of hue and saturation channels in HSV colour space.
6) *Group motion model*, $U_g = (u_{(g,x)}, u_{(g,y)})$, is estimated by taking the average motion models of all m objects

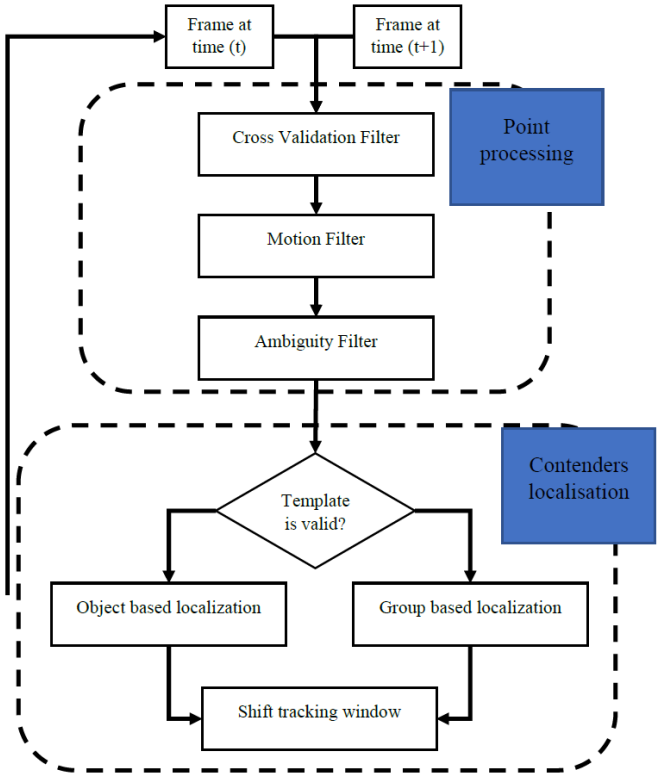

Fig. 1. The block diagram of proposed tracking model. The point processing block aims to find the best points that represent the object property. For object localisation, two strategies is used, Object based localisation is applied when the sample points properly represent the object local properties and when the local information dos not properly represent the object property the localisation is switched to the group based model.

using,

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

7) *Object relative speed*, $U_v = (u_{(v,x)}, u_{(v,y)})$, refers to the relative speed of the individual object with respect to the group motion model, viz

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

8) Background motion, $U_b = (u_{(b,x)}, u_{(b,y)})$, refers to the dominate motion in the frame. The background motion is estimated using the algorithm describe by Hedayati et al. [22].

B. Point Processing

The point processing block is based on the assumption that the points are rarely independent and they are parts of bigger units called object, and therefore they should have similar motion and colour distribution as the object. Thus the location of the object can be estimated by tracking the sample points that are distributed over the object surface. The point processing block aims to find the sample points that well represent the object using three stage filtering, namely the cross validation filter, the motion filter and the ambiguity filter.

1) Cross Validation Filter: In cross validation filter the forward-backwards error described by Kalal et al. [19] is used to estimate the stability of motion cues at point level. With having 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

$$p' = \begin{cases} 0 & \varepsilon_{\text{FB}} \geq \alpha \\ 1 & \text{elsewhere.} \end{cases} \tag{4}$$

2) Motion Filter: Knowing the background motion (U_b), object motion (U_o) and the motion of sample points (U_p), the probability of the sample point is estimated, and sampling points with more probability that are background are filtered out by,

$$p' = \begin{cases} \text{Background} & d(U_p, U_b) < d(U_p, U_o) \\ \text{Foreground} & \text{elsewhere} \end{cases} \tag{5}$$

Here d is the Euclidean distance function.

3) Ambiguity Filter: When the tracked object is occluded by another, some sample points that belong to one object might move to the other which eventually causes tracking drift. For the occlusion problem of k objects, this task can be formulated as maximising a posterior by,

$$\mathbf{k}^* = \underset{\mathbf{k}}{\operatorname{argmax}}\, S. \tag{6}$$

The vector S indicates how likely sample points p' are generated from object O. To measure the similarity (s) the histogram intersection, proposed by Swain and Ballard [23], 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. Having the object colour distribution (H_O) and point level colour distribution ($H_{p'}$) the similarity score is found by intersection using,

$$s = \sum_i \min(H_{p'}(i), H_O(i)), \tag{7}$$

where i is the bin number of the histogram.

C. Localisation of Objects

We use two different strategies to locate the objects in the frame namely object based localisation and group based localisation. Object based localisation predicts the new location of the tracking window by finding the mass centre of the weighted sample point. However when the object is obscured by some other element in the video or has unpredictable motion the sample points no longer represent the object template. In this case the group motion flow is used to estimate the location of objects. Which strategy is used is determined by estimating the quality of the object template inside the tracking window. The number of sample points before and after point processing block are compared and if more than 60% of the samples points are filtered out in the filtering stages, the object template is not valid and the group based localisation is triggered.

1) Object based localisation: In object based localisation, the quality of the points (after point processing stages) are estimated by finding the colour similarity between each sample point and the object template using Equation 7 and 50% of the points with lowest similarity matches are removed. From the remaining points the centre of mass for the new tracking window is calculated by,

$$C_x = \frac{\sum_i^n s_i p'_{x,i}}{\sum_i^n s_i}, \qquad C_y = \frac{\sum_i^n s_i p'_{y,i}}{\sum_i^n s_i} \tag{8}$$

where n is number of remaining sample points and s is the colour distribution similarity.

2) Group based localisation: To find the approximate location of an object using group information, three values are estimated, being *last valid object motion*, *group motion flow*, and *object relative speed to the group*. Last valid object motion is the last estimated motion vector of the object that does not suffer from occlusion or unpredicted motion (see Equation 1). Group motion model is estimated by taking the average motion models of all valid objects using Equation 2 and the object relative speed is estimated by mean of Equation 3. Having above value the new location of the object is approximated by moving the tracking window by the relative speed of objects to the group by,

$$C'_x = C_x + u_{(v,x)} u_{(g,x)}, \qquad C'_y = C_y + u_{(v,y)} u_{(g,y)} \tag{9}$$

where C' is the new centre of tracking window.

IV. Evaluation method

The main purpose of this evaluation is to show how group property and background motion information improve the tracking performance under occlusion and background clutter. The robustness of proposed tracking model is compared with a state of art tracking algorithm, namely the kernel correlation filter (KCF), which achieves the highest performance among the recent top-performing trackers [16]. To do this evaluation three entities are defined: the tracker output (T), the correct result or the ground truth (GT) and distance function (d) which is a measure of the similarity between tracker output and the ground truth [24]. The tracker output and the ground truth are deliminated by bounding boxes. The relative overlap of the ground truth and the tracker output determines the tracking accuracy according to,

$$d(T, GT) = \frac{T \cap GT}{T \cup GT}. \tag{10}$$

When $d = 0$ there is no overlap between ground truth and tracking output bounding boxes, whereas $d = 1$ occurs when the two bounding boxes are identical. An object is considered correctly tracked if the tracking output is within a distance threshold of ground truth where the most common threshold to consider correct tracking is 0.5 [24].

Two challenging videos are used for this evaluation. The videos are of a group of five people walking together passing obstacles such as trees and other persons in the scene. The five people are the *objects* to track. The ground truth was built manually by extracting the bounding box of each object in the group for every frame of video. It should be noted that manual selection was used to initialise the trackers with the location of objects in the first frame of each video as it is tracking of an initially located object that is under investigation here.

The tracking performance for individual objects per video are shown in Figures 2 and 3. As shown in Figure 2 the performance of both models are identical until just before the objects walk behind the tree. It is clear from Figure 4, the KCF tracker failed to track four of the five objects when they are occluded by the tree. This poor performance is due to two main reasons: first the KCF algorithm did not encode the background motion information, therefore it does not distinguish between the background element (tree) and tracked object. This leads to the second and bigger problem which exists in almost all tracking-by-detection algorithms. It was highlighted above (Section II) the goal of the tracking-by-detection algorithm is to train the online classifier to distinguish the tracked object from the background, but each training update can introduce error. To be specific at the point of occlusion, the tree is considered as a tracked object and the classifier is trained with the wrong features which leads to the tracking drift. A trace of this drawback also is seen in Figure 3 when the tracked object is occluded by other objects in the scene.

V. Conclusion

This paper combined background motion information and group motion dynamic with local object information to improve tracking under occlusion and background clutter. The performance of proposed tracking model is compared with KCF tracker. The comparison result indicates the KCF tracker performance is poor in comparison with the proposed model, particularly in the presence of occlusion and background noise.

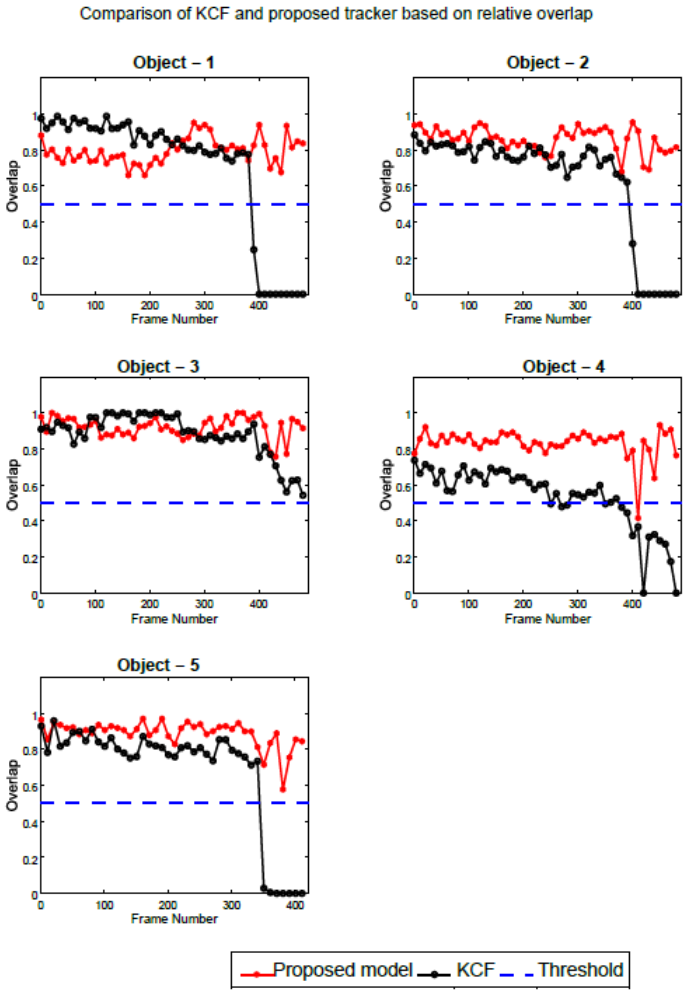

Fig. 2. Relative overlap of individual objects for the Waikato-1. The red lines are the result of our proposed model, the black lines show the result of KCF tracker and the blue lines indicate the distance threshold when set to 0.5.

References

[1] L. Yang, Hanxuan Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.

[2] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[3] D. E. Maggio and D. A. Cavallaro, *Video Tracking: Theory and Practice*, 1st ed. Wiley Publishing, 2011.

[4] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

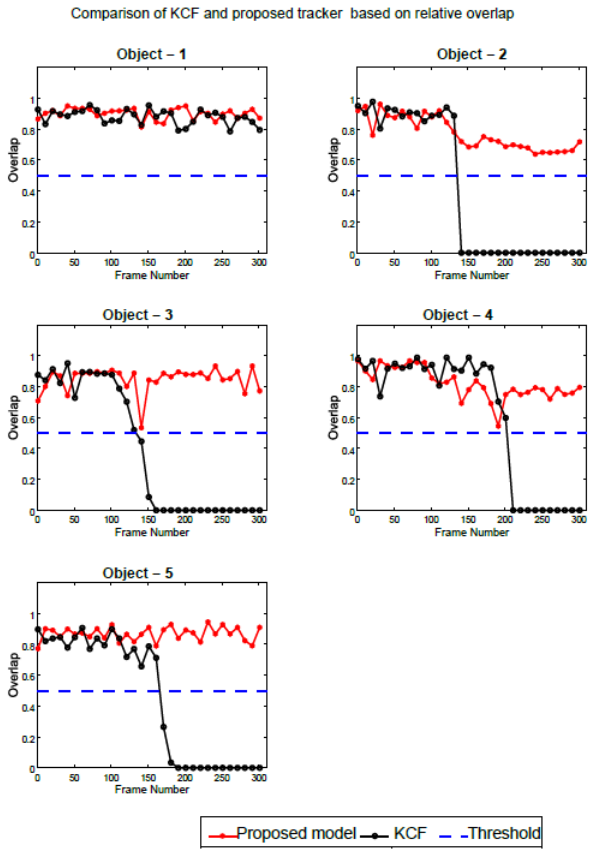

Fig. 3. Relative overlap of individual objects for the Waikato-2. The red lines are the result of our proposed model, the black lines show the result of KCF and the blue lines indicate the distance threshold when set to 0.5.

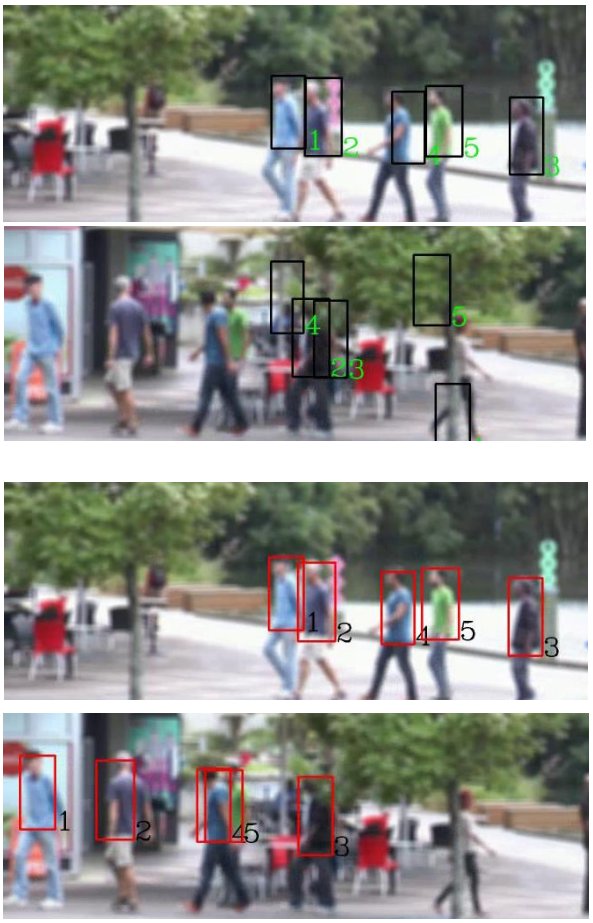

Fig. 4. Black rectangular box shows the result of KCF and the red box illustrate the proposed tracking output just before (top image) and after (bottom image) occlusion in Waikato–1.

[5] Y. Seo, S. Choi, H. Kim, and K.-S. Hong, "Where are the ball and players? soccer game analysis with color-based tracking and image mosaick," in *International Conference on Image Analysis and Processing*. Springer, 1997, pp. 196–203.

[6] J. Han, D. Farin, W. Lao *et al.*, "Automatic tracking method for sports video analysis," in *Proc. Symposium on information theory in the Benelux, Brussels, Belgium*, 2005.

[7] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2. IEEE, 1999, pp. 1197–1203.

[8] A. Blake and M. Isard, "The condensation algorithm-conditional density propagation and applications to visual tracking," in *Advances in Neural Information Processing Systems*, 1997, pp. 361–367.

[9] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," *Computer vision–ECCV*, pp. 661–675, 2002.

[10] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and vision computing*, vol. 21, no. 1, pp. 99–110, 2003.

[11] D. Comaniciu and P. Meer, "Robust analysis of feature spaces: color image segmentation," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1997, pp. 750–755.

[12] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," in *Intel Technology Journal*, 1998, pp. 214–219.

[13] S. Avidan, "Support vector tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 8, pp. 1064–1072, 2004.

[14] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[15] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr, "Struck: Structured output tracking with kernels," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2096–2109, 2016.

[16] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine*

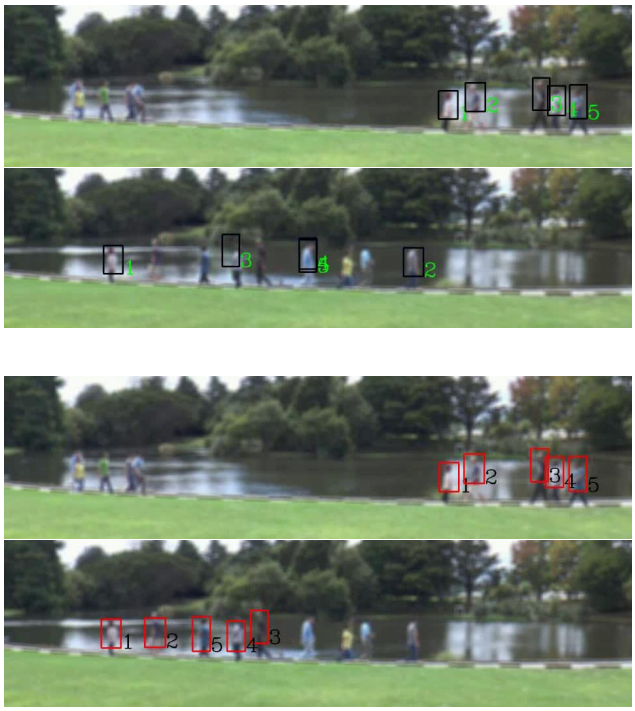

Fig. 5. Black rectangular box shows the result of KCF tracking and the red box illustrate the proposed tracking output just before (top image) and after (bottom image) occlusion in Waikato–2

Intelligence, vol. 37, no. 3, pp. 583–596, 2015.

[17] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration." in *ECCV Workshops (2)*, 2014, pp. 254–265.

[18] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European conference on computer vision*. Springer, 2012, pp. 702–715.

[19] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *20th international conference on Pattern recognition (ICPR)*. IEEE, 2010, pp. 2756–2759.

[20] J.-Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 5, no. 1-10, p. 4, 2001.

[21] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994, pp. 593–600.

[22] M. Hedayati, M. J. Cree, and J. Scott, "Scene structure analysis for sprint sports," in *International Conference on Image and Vision Computing New Zealand (IVCNZ)*, 2016, pp. 1–5.

[23] M. J. Swain and D. H. Ballard, "Color indexing," *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.

[24] A. Milan, K. Schindler, and S. Roth, "Challenges of ground truth evaluation of multi-target tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 735–742.