Object Tracking based on Path Similarity of Boosted Decision Trees

Koichi Mitsunari[1], Jaehoon Yu[1], Yoshinori Takeuchi[1], and Masaharu Imai[1]
[1]Department of Information Systems Engineering, Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
E-mail : {k-mitunr, yu.jaehoon, takeuchi, imai}@ist.osaka-u.ac.jp

Abstract: For general object recognition, detection and tracking are important and complementary components but require different types of features: detection requires to extract common features of a target category, and tracking requires to extract unique features of each target object. Therefore, it is difficult to efficiently fuse two core components into one. To address the issue, in the paper, we present a novel tracking method using byproducts of a detection process using a boosting classifier composed of binary decision trees, which are the path similarities between binary decision trees. In the experimental result, the proposed method achieved comparable tracking capability to a conventional tracking method using online boosting without any extra computation.

Keywords—**Object Tracking, Path Similarity, Boosted Decision Trees**

1. Introduction

In visual object recognition, object detection locates target objects in images, and object tracking estimates a trajectory of each object in consecutive frames. In both research fields, the precision of detection and tracking is significantly improved for the last five years by the advent of sophisticated approaches [1], [2]. For object detection, it is a well-known knowledge that boosting classifiers using binary decision trees as weak learners can achieve leading-edge detection performance for various target objects, which is also known as boosted decision trees (BDT), such as aggregated channel features (ACF) [3]. Also, for object tracking, there exist tracking methods using online learning algorithms such as online boosting [4], which is a variant of AdaBoost [5] and can adaptively handle color and shape changes.

For further improvement, recent visual object recognition systems complementarily combine object detection and tracking such as tracking-by-detection and detection-by-tracking [6], [7]. However, in general, object detection and tracking require different types of features: object detection uses common features for a target category and object tracking uses unique features for each target. Therefore, it is difficult for conventional systems to process object detection and tracking efficiently. In order to solve this issue, we propose a novel feature extraction method using BDT shared by object detection and tracking. The proposed method provides information for object detection and tracking from an identical BDT and contributes to the reduction of computational cost: node information for detection and edge information for tracking.

The rest of this paper is organized as follows. Section 2 explains a tracking-by-detection framework used for evaluation. Section 3 describes the proposed feature descriptor using BDT, and Section 4 evaluates the proposed method on a multiple object tracking platform. Finally, Section 5 concludes this paper.

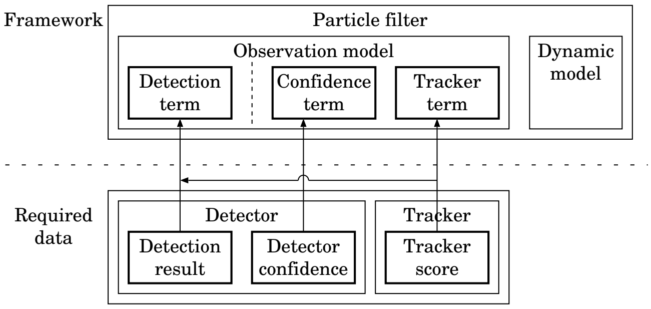

Figure 1. Tracking-by-detection framework

2. Tracking-by-detection framework

To achieve an efficient combination of detection and tracking processes, tracking-by-detection mainly focuses on improving tracking results based on detection results, and detection-by-tracking is vice versa. The proposed method can be used in either case, and one of tracking-by-detection frameworks is adopted for evaluation in this paper: an online multi-person tracking-by-detection [7]. This framework is based on a particle filter to estimate the distribution of each target state from multiple observations and assumptions, and the proposed feature descriptor is also evaluated on the identical particle filter framework.

The particle filter consists of two main components as usual: dynamic and observation models, which define drift and weight of each particle respectively. Figure 1 summarizes the particle filter and the required data used in [7]: the upper half shows the particle filter and the lower half shows the required data for it. As shown in Fig. 1, the observation model is defined by detection, confidence, and tracker terms. If there exists a positive match between a detection result and a tracker, the detection term evaluates the distance between the particle and the associated detection, and primarily guides the tracking process based on the evaluation result. Otherwise, in the case of detection misses or occlusions, the confidence and the tracker terms evaluate the detector confidence and the tracker score at the particle position, and guide the tracking process.

This type of frameworks can be used for any detectors and trackers, but the problem is computational cost caused by processing detection and tracking separately. Especially, for trackers, online learning algorithms are commonly used for adapting a target's color and shape changes, which takes

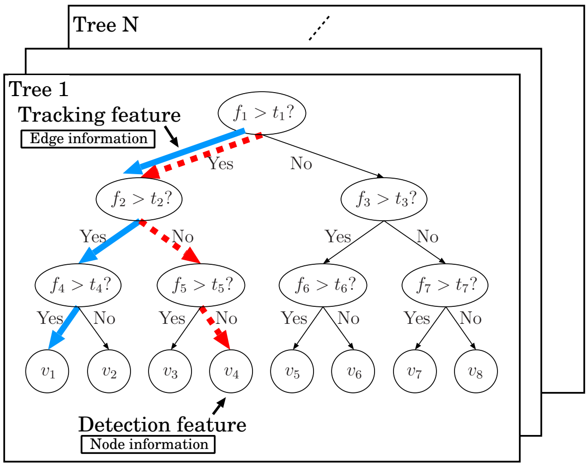

Figure 2. Proposed feature extraction sharing for object detection and tracking

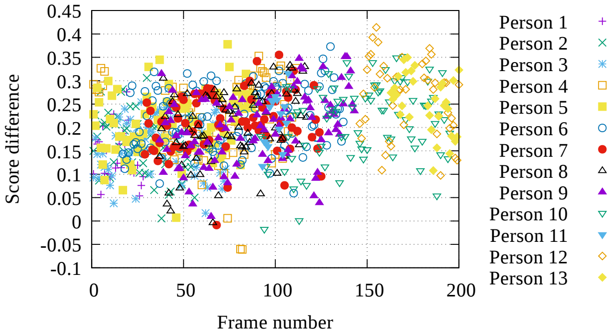

Figure 3. Discrimination capability of the path similarity

Table 1. LUT from $(p \oplus q)$ to $s(p, q)$

$(p \oplus q)$	000	001	01X	1XX
$s(p, q)$	1	2/3	1/3	0

X: don't care

a relatively large amount of computational cost, and for the same reason, in [7], Breitenstein *et al.* used the online boosting [4] for their tracker. The proposed feature descriptor presented in the following section focuses on this computational cost issue and effectively works for both detector and tracker.

3. Tracking using path similarity

This section presents how to define the similarity between objects by utilizing the identical BDT used in detection. As shown in Fig. 2, the BDT consists of N decision trees, and each tree consists of nodes and edges. In detection, only the node information is used to solve whether an object can be classified into the target category or not. Each node except leaves decides the next node based on the comparison between a feature value and a threshold, the decision tree outputs an evaluation value when the process reaches a leaf node, and the classification result is decided based on the sum of all outputs. In contrast to the detection process, the proposed tracking method utilizes the edge information, which is based on the prediction that since corresponding features extracted from an identical object between consecutive frames are similar, the paths of corresponding trees will be as well.

The path of a d-depth binary tree can be represented by d-bit binary representation: given 0 for a left edge and 1 for a right edge, the bold solid and the bold dotted paths in Fig. 2 are represented as $(000)_2$ and $(011)_2$, respectively. In the binary representation, the most significant bit represents the edge from the root node and the least significant bit represents the edge to a leaf node. This bit order is according to importance in path similarity. Then, the path similarity is defined as the ratio of common edges between two paths to depth d, and can be calculated by the following equation:

$$s(p, q) = 1 - \frac{\lfloor \log_2(2(p \oplus q) + 1) \rfloor}{d}, \tag{1}$$

where p and q are the binary representations of two paths for comparison, and $\oplus$ is a bitwise exclusive OR operation. By substituting $(000)_2$ and $(011)_2$ to the equation, the path similarity between the bold dotted path and the bold solid path can be easily obtained, and is $1/3$. The path similarity of the entire BDT is simply defined as the mean of each path similarity:

$$S(P, Q) = \frac{1}{N} \sum_{n=1}^{N} s(p_n, q_n), \tag{2}$$

where N is the number of decision trees, and P and Q are two sets of N paths, which are $\{p_1, \dots, p_N\}$ and $\{q_1, \dots, q_N\}$.

The proposed path similarity is simple but enables to achieve comparable discrimination capability with the conventional online boosting method [7]. As a preliminary simulation, its discrimination capability has been examined on TUD-Crossing pedestrian dataset [6]. In the preliminary experiment, a BDT is trained by ACF [3] and AdaBoost [5] using INRIA person dataset [8], which consists of 2048 depth-two decision trees, and soft cascade [9] pruning of boosting is disabled to evaluate the genuine discrimination capability. Figure 3 shows the distribution of each path similarity difference between objects from consecutive frames, where positive score differences represent that the target object can be discriminated from others and negative ones for vice versa. From the result, the ratio of the number of positives to the total number is up to 99.4%. Taking into consideration that the final tracking prediction is based on multiple cues such as objects' location and size, it is sufficient to use the proposed feature descriptor for practical tracking applications.

Although a logarithm function and a division by constant are used in Eq. (1) for convenience of mathematical expression, the path similarity can be easily implemented in both software and hardware platforms. Table 1 shows a lookup table (LUT) from $(p \oplus q)$ to $s(p, q)$ with 3-bit width. As shown in Table 1, the path similarity $s(p, q)$ depends on only the leading 1 in the binary representation of $(p \oplus q)$, where X is "don't care", which means that $s(p, q)$ does not depend on it. Obviously, if a small LUT is allocated for the path similarity, it is not necessary to use troublesome functions such as the logarithm. Also, in hardware platform, finding leading 1 is much easier with a dedicated circuit even without LUTs.

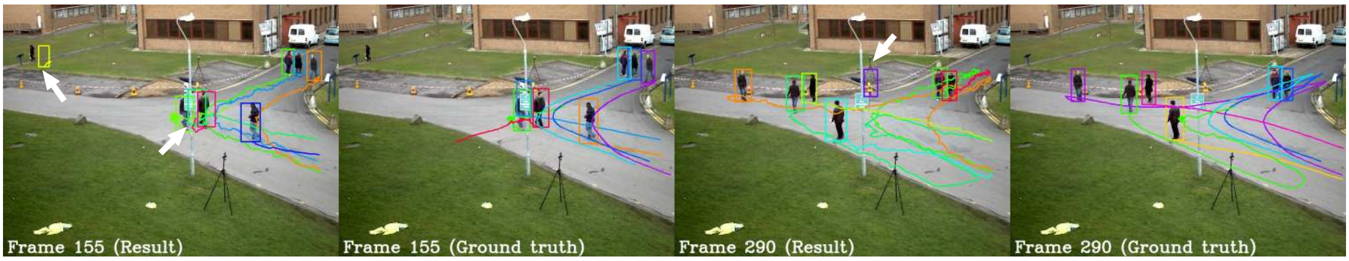

Figure 4. Tracking results and ground truth on PETS'09 S2L1

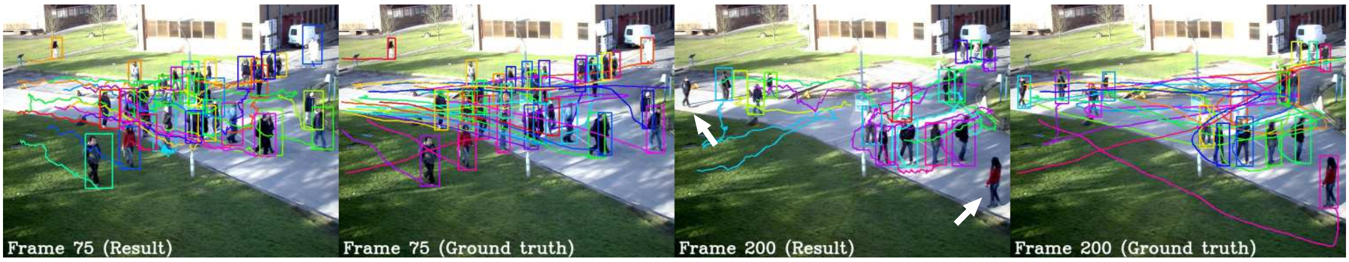

Figure 5. Tracking results and ground truth on PETS'09 S2L2

Table 2. PETS'09 dataset [10]

Sequence	Resolution	Frames	Tracks	Density	Camera	Ground truth
S2L1	768x576	795	19	5.8	static	[11]
S2L2	768x576	436	43	23.6	static	[12]

Table 3. MOTA and MOTP comparison

Dataset	Method	MOTA↑	MOTP↑
PETS'09 S2L1	Breitenstein *et al.* [7]	**0.797**	0.563
	proposed	0.712	**0.702**
PETS'09 S2L2	Breitenstein *et al.* [7]	**0.500**	0.513
	proposed	0.417	**0.650**

4. Evaluation

This section examines the tracking performance of the proposed method on multiple object tracking (MOT) framework [7], and the same BDT described in the preliminary experiment is used for it. The evaluation is conducted on two sequences S2L1 and S2L2 of PETS'09 dataset [10] and the details are shown in Table 2. The ground truth annotation of each dataset is available: [11] for S2L1 and [12] for S2L2.

Figures 4 and 5 show pairs of a tracking result and a ground truth trajectories of PETS'09 S2L1 and S2L2, where each rectangle represents a target's position and each line represents a trajectory. From the result, it is confirmed that the proposed method correctly estimates most trajectories, but fails to track some targets, which are pointed by arrows in Fig. 4 and 5. The tracking performance on the entire datasets is shown in Table 3. For quantitative analysis, two major MOT metrics, MOTA and MOTP, are selected from CLEAR MOT metrics [13]: MOTA indicates errors of false positives, false negatives, and ID switches, and MOTP indicates the average overlap between annotated and predicted bounding boxes. In both MOTA and MOTP, a higher value indicates a better result. In Table 3, the proposed method shows higher MOTP but lower MOTA than the conventional method. This result represents that the proposed method can precisely localize objects in each frame but assigns different IDs for an identical object while the tracking process.

For more detailed analysis, we classify the tracking results into four patterns in Fig. 6: combinations of success/failure in detection and tracking. As shown in Fig. 6, each tracking process is properly guided by a correct detection result as (a) or by detector confidences and tracker scores as (b). In both cases, it is easily confirmed that the particles of each tracker are concentrated on surround of its target. However, as shown in (c) and (d), in the case that the target is occluded by other objects, it is difficult to correctly track it because both detector confidences and tracker scores are unreliable due to foreground objects, even if the target is correctly detected. This is the main cause of degrading the proposed method's MOTA in Table 3.

In the current state, it is difficult to directly compare the tracking performance between the proposed method and the conventional method, but considering both MOTP and MOTA it is possible to say that the proposed method achieved comparable tracking results to the conventional method [7].

5. Conclusion

This paper proposed an object tracking feature descriptor based on BDT's path similarity. The underlying idea is that similar features generate similar BDT's paths. The proposed feature descriptor was examined in terms of discrimination capability and tracking performance through the preliminary

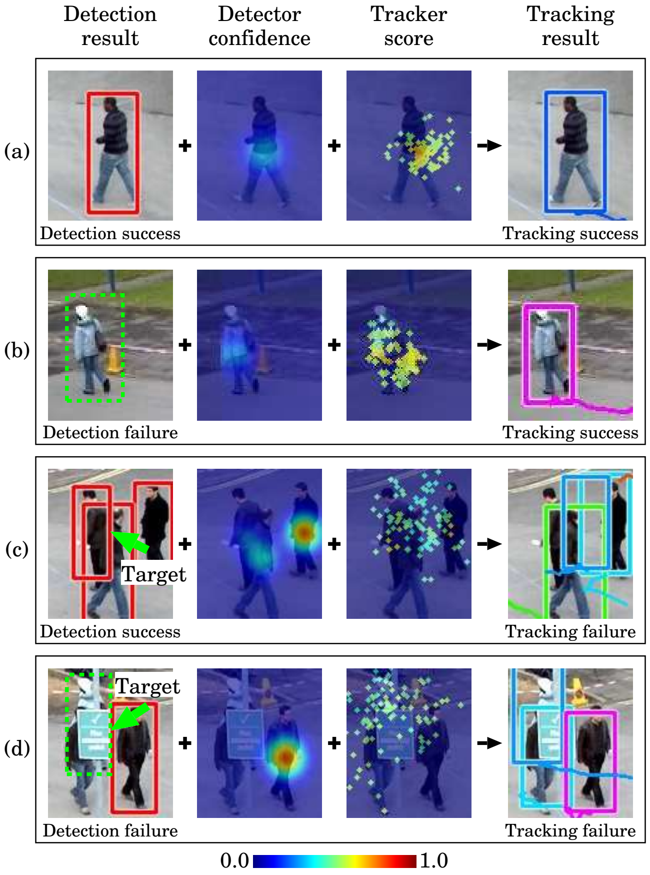

Figure 6. Three required data of observation model and its tracking result

experiment and the MOT benchmark. In both evaluation, even though a boosting classifier with shallow, depth two, decision trees was used, the preliminary experiment showed that the proposed method has comparable discrimination capability to the conventional method, and the MOT benchmark showed that the proposed method is applicable to practical applications.

Since the decision trees can be deepened in training phase and deeper trees can provide more resolution of path similarity, the tracking performance of the proposed method can be easily improved by retraining the boosting classifier. Moreover, considering the fact that the boosting classifier used in this paper is trained as a dedicated detector, there also exists the possibility of improving tracking performance by manipulating the structure of decision trees: even if a decision tree is altered for tracking in training phase, boosting algorithm can recover classification performance by adaptively training its subsequent decision trees.

The proposed method is a promising feature descriptor especially for embedded systems because it has a computational cost advantage by sharing BDT for detection and tracking and can be easily implemented. For a future work, we are planning to improve the proposed method's performance based on the ideas mentioned above and implement it in a hardware platform.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP16K16085.

References

[1] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.

[2] W. Luo, X. Zhao, and T. Kim, "Multiple object tracking: A literature review," *arXiv:1409.7618 [cs]*, Sep. 2014.

[3] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[4] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, vol. 1, Jun. 2006, pp. 260–267.

[5] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. and Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[6] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, Jun. 2008, pp. 1–8.

[7] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1820–1833, Sep. 2011.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, vol. 1, Jun. 2005, pp. 886–893.

[9] L. D. Bourdev and J. Brandt, "Robust object detection via soft cascade," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2005, pp. 236–243.

[10] J. Ferryman and A. Shahrokni, "PETS2009: Dataset and challenge," in *Proc. IEEE Int. Workshop Performance Evaluation of Tracking and Surveillance*, Dec. 2009, pp. 1–6.

[11] B. Yang and R. Nevatia, "Multi-target tracking by on-line learning of non-linear motion patterns and robust appearance models," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, Jun. 2012, pp. 1918–1925.

[12] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 58–72, Jan. 2014.

[13] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image and Video Process.*, vol. 2008, pp. 1–10, Jan. 2008.