ChinaRxiv

AI translation · View original & related papers at
chinarxiv.org/items/chinaxiv-201809.00133

Postprint: Object Tracking Algorithm Based on Improved TLD

Authors: Hu Xin, Gao Jiali

Date: 2018-09-12T00:00:00+00:00

Abstract

This research addresses the problems of the traditional Tracking-Learning-Detection (TLD) object tracking algorithm, including excessive detection time caused by the detection module scanning numerous sub-windows and tracking failure when the target undergoes severe occlusion or deformation during tracking, and proposes an improved TLD object tracking algorithm. The improved algorithm incorporates a ViBe model before the detection module to estimate foreground objects, significantly reducing the detection region. The tracking module replaces the optical flow method in the original algorithm with the SIFT feature matching algorithm to accurately track the target and avoid tracking drift, reducing computational complexity and enhancing the algorithm' s adaptability to environmental changes. Experimental results demonstrate that the improved TLD algorithm achieves enhanced running speed, and the tracking accuracy is considerably improved when the target experiences severe occlusion or drastic illumination changes.

Full Text

Abstract

The traditional tracking-learning-detection (TLD) target tracking algorithm suffers from excessive detection time due to the detection module scanning numerous sub-windows, and often fails when the target undergoes severe occlusion or deformation. To address these issues, we propose an improved TLD target tracking algorithm. The improved algorithm incorporates a ViBe model before the detection module to estimate foreground targets, significantly reducing the detection region. The tracking module replaces the original optical flow method with SIFT feature matching, enabling accurate target tracking while avoiding drift, reducing computational complexity, and enhancing environmental adaptability. Experimental results demonstrate that the improved TLD algorithm

achieves higher running speed and improved tracking accuracy under severe occlusion and dramatic illumination changes.

Key words: TLD algorithm; ViBe algorithm; SIFT feature matching algorithm; tracking drift

0 Introduction

Target tracking technology has always been a hot research topic in computer vision, with wide-ranging applications in automatic video surveillance, vehicle navigation, intelligent robot guidance and positioning, human-computer interaction, intelligent transportation systems, UAV target tracking, and video retrieval [1]. Through extensive research by scholars, video target tracking technology has matured considerably, attracting increasing attention from researchers worldwide.

Traditional tracking algorithms fall into two categories: tracker-based methods and detector-based methods [2]. Tracker-based approaches assume the target remains visible throughout, failing when the target disappears or becomes occluded [3], which limits their applicability to short-term tracking. Detector-based methods track targets by scanning each frame with sliding windows, but require pre-selection of numerous positive and negative samples for detection and learning, making them unsuitable for long-term target detection [4]. Neither approach alone yields satisfactory results for long-term target tracking.

To overcome these limitations, combining target tracking and detection algorithms offers a promising solution. When occlusion or target disappearance causes tracking failure, the detector can redetect the target [5], allowing the tracker to resume. This synergy led to the development of the TLD (Tracking-Learning-Detection) target tracking algorithm, which integrates traditional detection and tracking techniques with an improved online learning mechanism [6] to handle deformation, illumination changes, and partial occlusion. However, the classical TLD algorithm's detection module scans an excessive number of subwindows, resulting in long detection times and reduced system efficiency. Furthermore, severe occlusion, deformation, and illumination changes often cause tracking failure, motivating the improvements presented in this paper.

1 TLD Target Tracking Algorithm

The TLD target tracking algorithm, proposed by Kalal et al. [7] in 2012, is a single-target long-term tracking algorithm consisting of four main modules: tracking, learning, detection, and integration [8]. The algorithm framework is illustrated in Figure 1 [Figure 1: see original paper].

The algorithm executes as follows: First, a video sequence is input, and the target region is manually specified in the first frame [9]. This region is then processed in parallel by the detection, learning, and tracking modules. The tracker detects target motion between consecutive frames, though it fails when

the target moves out of the camera' s view. The detector scans each frame to locate regions similar to the learned target appearance. The learning module improves detector performance through online video processing, evaluating the tracker's results to assess detector errors and generate training samples for model updates. Finally, the integration module combines results from the tracking and detection modules to produce the final target location.

1.1 Tracking Module

To select reliable tracking points, the tracker employs the median-flow algorithm, an improved version of the Lucas-Kanade (LK) optical flow method [10]. As shown in Figure 2 [Figure 2: see original paper], this algorithm uses forward-backward error to predict target location with forward-backward consistency [11]. Starting from initial position at time , tracking forward yields at time , while backward tracking from produces predicted position at time . The displacement deviation between these trajectories is called forward-backward error [11]. Feature points with forward-backward error less than the median value are used to compute the new target window, completing short-term tracking.

1.2 Learning Module

The TLD learning module employs P-N semi-supervised learning to improve detector performance through online processing of each frame. Based on detection results, it generates new positive and negative samples to continuously update the detector. Figure 3 [Figure 3: see original paper] illustrates the P-N learning principle: starting with labeled samples for supervised training to obtain an initial classifier, then using iterative learning to classify unlabeled data. P-experts identify misclassified negative samples, while N-experts identify misclassified positive samples, correcting the training set to improve classifier performance in subsequent iterations [12].

1.3 Detection Module

The detector locates potential target positions in the current frame based on the target model learned from previous observations. TLD uses a cascade classifier consisting of three components: variance filter, ensemble classifier, and nearest neighbor classifier [13]. Rectangular boxes obtained through sliding windows of various sizes are fed into the cascade classifier. The variance filter uses integral images to compute pixel variance, rejecting boxes with variance less than 50% of the tracked image patch variance. Remaining boxes pass to the ensemble classifier, where boxes with average posterior probability greater than 50% are considered target candidates. For surviving boxes, correlation similarity is computed, with values exceeding 0.65 representing the final foreground target. The detection principle is shown in Figure 4 [Figure 4: see original paper].

1.4 Integration Module

The TLD integration module combines results from the detection and tracking modules. If neither module produces a target box, the target is considered absent in the current frame. Otherwise, the box with maximum conservative similarity is selected as the final target location.

2 Improved TLD Algorithm

2.1 Detection Module Improvement

The classical TLD detection module performs global scanning of each frame, generating numerous sub-windows, most of which contain no foreground target. This process results in long detection times and increased algorithmic complexity, degrading real-time performance. We address this by employing the ViBe algorithm for foreground target detection, passing only image patches containing foreground targets to the variance filter. Patches that successfully pass through the entire cascade classifier are identified as target patches, substantially reducing the number of scanned sub-windows and improving cascade classifier efficiency [14].

ViBe is a pixel-level foreground detection algorithm with high real-time performance and accuracy [15]. The algorithm operates in three stages:

a) **Background model initialization**: For each pixel in the first frame, 8-neighbor pixel values are randomly sampled 20 times to construct the background model for that pixel.

b) **Foreground segmentation**: For subsequent frames, the distance between new pixel values and background model samples is computed. If more than 2 samples are within the threshold distance, the pixel is classified as background, and the background model is updated.

c) **Model update**: If a pixel is identified as background, it has a 1/ probability of updating the background model by randomly replacing one sample. Additionally, with 1/ probability, the neighboring pixel' s background model is updated by randomly selecting a value from the 8-neighbor sample set.

2.2 Tracking Module Improvement

The classical TLD tracking module uses the improved optical flow method, which assumes brightness constancy—a condition often violated in practice due to occlusion, light sources, and noise. Significant illumination changes frequently cause target loss. Moreover, when target plane rotation causes deformation, optical flow struggles to form motion vector fields, leading to tracking failure.

We replace optical flow with the SIFT feature matching algorithm, which offers robust matching capabilities for frame pairs and exhibits invariance to illumina-

tion, plane rotation, and target deformation. Even with substantial viewpoint changes, SIFT maintains stable feature matching [16], enabling accurate target tracking in practice. The SIFT algorithm consists of four steps:

a) **Scale-space extremum detection**: The scale space of 2D image I(x,y) at various scales is obtained by convolving I(x,y) with Gaussian kernel G(x,y,). A Difference-of-Gaussian (DoG) pyramid is constructed, and extrema are detected across 26 neighbors in DoG scale space.

b) **Keypoint localization and scale determination**: Unstable keypoints with low contrast and edge responses are removed based on stability criteria.

c) **Keypoint orientation assignment**: Computed from gradient distribution in the neighborhood. The gradient magnitude m(x,y) and orientation (x,y) at each point L(x,y) are calculated as:

$$m(x,y) = \sqrt{[L(x+1,y) - L(x-1,y)]^2 + [L(x,y+1) - L(x,y-1)]^2}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right)$$

d) **Keypoint descriptor generation**: The coordinate axes are rotated to the keypoint' s main orientation to ensure rotation invariance. A 16×16 neighborhood around each feature point generates a descriptor by dividing the region into 16 sub-blocks of 4×4 pixels and computing gradient orientation histograms with 8 bins in each sub-block, forming a 128-dimensional SIFT feature vector [17].

3 Experimental Results

Experiments were conducted using Visual Studio 2013, OpenCV 3.0.0, and MATLAB 2016a on a Windows 7 system with an Intel(R) CoreTM i5-6200U CPU @ 2.30 GHz. We evaluated both TLD and improved TLD algorithms on four test sequences from Visual Tracker Benchmarks.

Table 1 compares the number of scanning windows between TLD and improved TLD algorithms. Table 2 compares tracking speeds. Table 3 compares the number of successfully tracked frames.

Table 1 shows that for the David sequence, the improved algorithm reduced window numbers by approximately 6×; for Football, by 4×; for FaceOcc2, by 8×; and for Jumping, by 4×. **Table 2** demonstrates corresponding speed improvements: David increased from 10.7 to 19.5 fps; Football from 12.3 to 18.6 fps; FaceOcc2 from 14.3 to 21.8 fps; and Jumping from 16.4 to 23.2 fps. These results confirm that ViBe-based foreground estimation drastically reduces scanned sub-windows, accelerating detection and improving overall system efficiency.

While improving speed, tracking accuracy must also be maintained. **Table 3** compares successful tracking frames. For David, with severe illumination changes and pose variation causing plane rotation and occlusion, TLD tracked 628 frames while improved TLD tracked 698 frames. For Football, with severe deformation, occlusion, and target disappearance/reappearance, TLD successfully tracked only 164 frames compared to 325 frames for improved TLD. For FaceOcc2, with occlusion and plane rotation, TLD tracked 729 frames while improved TLD successfully tracked all frames. For Jumping, with illumination changes and camera jitter, TLD tracked 302 frames while improved TLD tracked all frames.

Figure 5 [Figure 5: see original paper] shows tracking results on the Football sequence, targeting the number "24" on a player' s jersey. Yellow boxes indicate TLD results; red boxes show improved TLD results. Both algorithms successfully track the target at frame 138 (Figure 5a). At frame 200 (Figure 5b), rapid player movement causes plane changes and severe occlusion, causing TLD to fail while improved TLD succeeds. At frame 362 (Figure 5c), when the target is fully occluded and reappears, TLD exhibits drift and fails, while improved TLD successfully relocates the target.

Figure 6 [Figure 6: see original paper] tracks a speaker' s mouth using webcam footage. At frame 60 (Figure 6a), both algorithms succeed. At frame 183 (Figure 6b), hand occlusion causes TLD to drift to the face (tracking failure), while improved TLD maintains accurate tracking. When background illumination changes dramatically (Figure 6c), TLD fails while improved TLD succeeds.

4 Conclusion

The traditional TLD target tracking algorithm enables long-term online tracking with minimal prior information, offering fast tracking speed and high real-time performance, and can reacquire targets after disappearance. Our improved algorithm incorporates ViBe foreground estimation before the detection module, drastically reducing scanned sub-windows while maintaining tracking accuracy and accelerating detection, thereby improving overall system efficiency. Replacing the tracking module' s optical flow with SIFT feature matching solves the original TLD' s tracking drift problem, as SIFT' s stability against viewpoint changes, affine transformations, and illumination variations significantly enhances tracking robustness. Compared with original TLD, the improved algorithm demonstrates substantially better tracking performance.

However, limitations remain. Adding ViBe foreground detection increases the detection module' s computational proportion, potentially affecting system coordination. Additionally, strengthening the learning module' s accuracy and sensitivity warrants further research.

References

[1] Tong Yuan, Fei Shumin, Shen Jie. Fast target tracking method based on TLD framework [J]. Application Research of Computers, 2018, 35 (1): 317-320.

[2] Guo Qiuyan. Video target tracking based on improved TLD algorithm [J]. Computer Engineering and Design, 2017, 38 (9): 2551-2555.

[3] Zhou Xin, Qian Qiumeng, Ye Yongqiang, et al. Improved TLD video object tracking method [J]. Chinese Journal of Image and Graphics, 2013, 18 (9): 1115-1123.

[4] Wang Tiedong, Ren Shiqing. An improved TLD target tracking algorithm [J]. Jiangsu Science and Technology Information, 2008, 35 (1): 52-53, 56.

[5] Zhou Junna, Chen Wei, Wang Ke, et al. Sparse prototype object tracking algorithm based on TLD [J]. Computer Engineering, 2017, 43 (06): 236-240.

[6] Liu Shu, Di Hongwei, Yao Manhong. TLD target tracking algorithm based on adaptive scale [J]. Optical Technology, 2017, 43 (6): 542-546.

[7] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2012, 34 (7): 1409-.

[8] Yang Yufeng, Li Weitong, Xu Lei, et al. An improved TLD tracking algorithm [J]. Technological Innovation and Application, 2016 (28): 63-64.

[9] Qu Haicheng, Shan Xiaochen, Meng Yu, et al. TLD target tracking algorithm for detecting regional dynamic adjustment [J]. Computer Application, 2015, 35 (10): 2985-2989.

[10] Wang Zhenhao. Improved face detection and tracking algorithm based on TLD [J]. Science and Technology Innovation Guide, 2013 (22): 50-51.

[11] Fu Miao, Xing Zangju. Improvement of TLD tracking algorithm by frame difference method [J]. Electronic Design Engineering, 2017, 25 (7): 183-186.

[12] Gong Xiaobao. Research on target tracking algorithm based on TLD framework [D]. Chengdu: Southwest Jiaotong University, 2014.

[13] Zhang Dan, Chen Xingwen, Zhao Shuying. Improved random forest TLD target tracking method [J]. Journal of Dalian University of Nationalities, 2016, 18 (3): 255-259.

[14] Xu Yong. Target tracking algorithm based on TLD framework [D]. Nanjing: Nanjing University of Aeronautics and Astronautics, 2017.

[15] Liu Chun, Zhai Zhiqiang. Improved ViBe motion target detection algorithm [J]. Sensor and Microsystem, 2017, 36 (1): 123-126.

[16] Feng Yan. Target detection algorithm based on feature matching in dynamic context [D]. Xi' an: Xi' an University of Electronic Science & Technology, 2014.

[17] Fu Weiping, Qin Chuan, Liu Jia, et al. Image target matching and location based on SIFT algorithm [J]. Instrument Report, 2011, 32 (1): 163-169.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.