Tracking Verification Algorithm Based on Channel Reliability

JunChang Zhang[1,2], ChenYang Xia[1] and JinJin Wan[2]
[1]*Institute of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China*
[2]*Key Laboratory of Photoelectric Control Technology, Luoyang, Henan 471000 China*

Keywords: Correlation filter, Channel reliability, Tracker, Validator, Siamese convolutional neural network

Abstract: For most algorithms, the problem of Tracking target performance degradation in the case of fast moving, illumination changes, target deformation, occlusion, out-of-plane rotation, low-resolution images, etc. This paper proposes a tracking verification algorithm based on channel reliability. The tracker part of the algorithm is tracked by the method of correlation filter based on channel reliability. By calculating the reliability weight of each feature channel of the input correlation filter, and multiplying the weight by the response of the corresponding channel to obtain the final response, so that the target positioning will be more accurate. The validator part uses the Siamese dual input network in the deep learning convolutional neural network. Every few frames, the verifier will verify the results of the tracker part of the algorithm. If the reliability is verified, the tracking result will not be modified. Otherwise, the validator will re-detect the target location and verify the reliability through the Siamese dual-input network. The tracker will regard this location as the new location of our target continues to be tracked, making target tracking more durable and robust. The experimental evaluation of the OTB13 video sequence proves that the proposed algorithm has good adaptability to target fast motion, illumination change, target deformation, occlusion, and out-of-plane rotation, and has good robustness.

1 INTRODUCTION

As one of the basic technologies of computer vision, target tracking technology is widely used in video surveillance, human-computer interaction, robot (Smeulders and Chu, 2014) and other fields. Although the target tracking technology has achieved a series of results in recent years, there are still many difficulties and challenges in object tracking, occlusion, rotation, illumination changes, and posture changes.

Existing model-free visual tracking algorithms are often classified as Discriminating or generating. Discriminating algorithms can be learned by multi-instance learning (MIL), compressed sensing, P-N learning, structured output SVM (Hare, Golodetz, Saffari, Vineet, Cheng, Hicks, and Torr, 2016), online enhancement, and the like. In contrast, the generated class tracker typically treats the tracking as the most similar area of the search to the target. To this end, various object appearance modeling methods have been proposed, such as incremental subspace learning and sparse representation (Fan and Xiang, 2017) Currently, one of the new trends in improving tracking accuracy is the use of deep learning tracking methods (Fan and Ling, 2017, Ma, Huang and Yang, 2015, Nam and Han, 2016) because they have strong discriminative power, as shown in (Nam and Han, 2016). However, the use of deep learning-based tracking algorithms is computationally intensive and less real-time.

Since MOSSE algorithm was proposed, the correlation filter (CF) has been considered as a robust and efficient method for visual tracking problems (Bolme, Beveridge, Draper and Lui, 2010). Currently, the proposed improvements based on the MOSSE algorithm include the inclusion of kernel and HOG features, the addition of color name features or color histograms (Bertinetto, Valmadre, Golodetz, Miksik, and Torr, 2016), and sparse fusion tracking (Zhang, Bibi and Ghanem, 2016), adaptive scales, mitigation of boundary effects (Danelljan, Hager, Shahbaz Khan, and Felsberg, 2015), based on Context-Aware correlation filter (Mueller, Smith, Ghanem, 2017) and fusion of deep convolutional network functions (Ma, Huang and Yang, 2015) algorithm.

Zhang, J., Xia, C. and Wan, J.
Tracking Verification Algorithm Based on Channel Reliability.
DOI: 10.5220/0008095900590064
In *Proceedings of the International Conference on Advances in Computer Technology, Information Science and Communications (CTISC 2019)*, pages 59-64
ISBN: 978-989-758-357-5

Although the speed or accuracy of the tracking algorithms mentioned above has improved, real-time high-quality tracking algorithms are still rare. So seeking trade-offs between speed and accuracy is a trend in future tracking (Mueller, Smith, Ghanem, 2017, Ma, Yang, Zhang and M.H. Yang, 2015). Context-Aware Correlation Filter Tracking (Mueller, Smith, Ghanem, 2017) proposes a new correlation filter framework that can add more background information and incorporate global background information into the learned filters for processing. The algorithm adds background information to the Staple algorithm, and the robustness to large size changes, background clutter and partial occlusion is improved and the impact of speed is relatively small. However, the algorithm is relatively less robust in the target plane, out-of-plane rotation, dramatic illumination changes, and fast motion. Therefore, in order to better and more accurately track the target, a tracking algorithm that balances the advantages and disadvantages of both can be found between real-time and high robustness. Therefore, this paper proposes a video target verification tracking algorithm based on channel reliability.

The algorithm in this article consists of two parts: a tracker and a validator. The validator is implemented by the Siamese network in the deep learning convolutional neural network. These two parts are independent of each other and work in harmony. Advantages (1): The channel reliability method is used to make the target positioning more accurate. That is, each feature channel is added with a corresponding weight, and then summed. (2): Verify the result of the tracker every few frames. When the verification system finds that the result of tracking a certain frame is incorrect, it will re-target the target to find the target position information and put the target new. The position returns to the tracker as the target position of the error frame, causing the tracker to continue tracking from this position.

2 THE TARGET VERIFICATION TRACKING ALGORITHM BASED ON CHANNEL RELIABILITY

The algorithm in this paper consists of two parts: tracker T and verifier V. The tracker is implemented using a correlation filter method based on Context-Aware to ensure real-time and location of the target. At the same time, the tracker sends a verification request to the validator with a fixed number of frames and responds to feedback from the validator by adjusting the tracking or updating model. The validator is implemented using the Siamese network in the deep learning convolutional neural network. After receiving a request from the tracker, the validator will first verify that the tracking results are correct and then provide feedback to the tracker. The overall block diagram is shown in Figure 1.

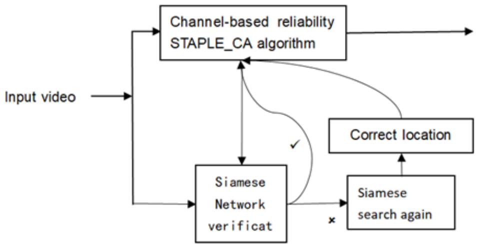


Figure 1: Overall block diagram of video target verification tracking algorithm based on channel reliability.

2.1 Channel Reliability Estimation

Channel reliability is calculated by constraining the properties of least squares solutions during the filter design process. The channel reliability score is used to represent the weight of each channel filter response when positioned, as shown in Figure 2.

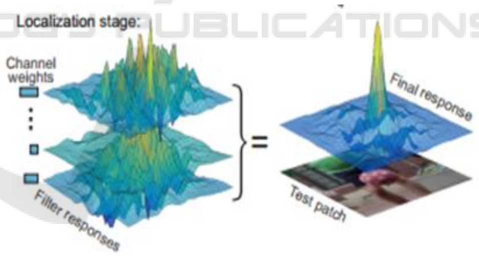


Figure 2: Channel reliability weights calculated in the constraint optimization step of correlation filter learning reduce the noise of the weighted average filter response.

The characteristic channel reliability of the target positioning phase is obtained by multiplying the learning channel reliability measurement value and the channel detection reliability measurement value. Assume that N_d is the total number of channels for a given correlation filter Hog feature. The corresponding set of N_d mutually independent channel features is $f=\{f_d\}_{d=1:N_d}\left(f_d \in R^{d_w \times d_h}\right)$. A discriminative feature channel f_d produces a filter w_d whose output $f_d * w_d$ is almost identical to the ideal response g.

On the other hand, the output response is noisy on feature channels with low discriminating power, and the global response error due to least squares will significantly reduce the peak of the maximum response. Therefore, the value of channel learning reliability is the maximum response of the learned filter.

In the channel detection reliability measurement phase, the expressive power of the main mode in each channel response can indicate the detection reliability of each channel. In addition, Bolme et al also proposed a similar method to detect target loss. Our measure is based on the ratio between the second and first major mode in the response map, i.e. $\rho_{\max 2} / \rho_{\max 1}$.

Note that this ratio penalizes cases when multiple similar objects appear in the target vicinity since these result in multiple equally expressed modes, even though the major mode accurately depict the target position. To prevent such penalizations, the ratio is clamped by 0.5. Therefore, the per-channel detection reliability is estimated as:

$$w_d^{(\det)} = 1 - \min(\rho_{\max 2} / \rho_{\max 1}, \frac{1}{2}) \tag{1}$$

2.2 Algorithm for Correlation Filter of Context-Aware Based on Channel Reliability

The traditional correlation filter tracking algorithm uses ridge regression to classify. A_0 is a circular matrix of all cyclically translated image blocks:

$$\min_W \|A_0 W - y\|_2^2 + \lambda_1 \|W\|_2^2 \tag{2}$$

Unlike traditional correlation filter frameworks, more background information is added to the framework of Context-Aware Correlation Filter.

In each frame, we sample the k Context-Aware image blocks $a_i \in R^n$ around the target $a_0 \in R^n$ according to a uniform sampling strategy (k=4). The corresponding cyclic matrices are $A_i \in R^{n \times n}$ and $A_0 \in R^{n \times n}$.

These Context-Aware image blocks contain global background information that causes various interference factors and different background forms, which can be considered as true negative samples. Intuitively, you need to learn a filter that has a high response to the target and a filter $W \in R^n$ that is close to zero response to the background image information patch block. The purpose is achieved by adding a Context-Aware image patch block as a normalization constraint to a standard formula (2). The result is as follows, the response regression of the target image block is the ideal response y, and the context image block is returned to zero by the parameter constraint λ_2.

$$\min_W \sum_{d=1}^{N_d} \|A_0 W - y\|_2^2 + \lambda_1 \sum_{d=1}^{N_d} \|W\|_2^2 + \lambda_2 \sum_{i=1}^{k} \left\| A_i \sum_{d=1}^{N_d} W \right\|_2^2 \tag{3}$$

Where A_i corresponds to a cyclic matrix formed by all cyclic shifts of image block A_i based on contextual background information obtained around the target. N_d indicates the number of associated filter feature channels.

Therefore, the final response of the algorithm is the product of the maximum response value obtained by formula (3) and the reliability estimation value $w_d^{(\det)}$ of the feature channel detection, so that the position information of the target can be more accurately located.

2.3 Siamese Verification Network

This paper uses the Siamese network (Comaniciu, Ramesh and Meer, 2000) to design the verifier V. The network consists of two convolutional neural network (CNN) branches and processes two inputs separately. In this network, VGGNet (Perronnin, Sanchez and Mensink, 2010) was borrowed from the architecture of CNNS and an additional area pooling layer was added. In the detection process, since V needs to process a plurality of regions in the image, and select one candidate most similar to the target as an output result. Therefore, the region pooling layer can simultaneously process a group of regions in each frame of image, thereby significantly reducing the amount of computation.

When the tracking result from T is input to the Siamese network, if its verification score is lower than the threshold τ_1 , V considers that the frame target tracking fails. In this case, V still uses the Siamese network to re-detect the target. Unlike the verification phase, the test needs to verify multiple image patches in a local area and find the target with the highest score.

The square area of size $\beta(w^2+h^2)^{\frac{1}{2}}$ is centered on the position of the tracking result in the verification frame, which is the detection area. Where w and h are the width and height of the tracking target, and β is the target size factor.

The target candidate set generated by the sliding window is recorded as $\{c_i\}_{i=1}^{N}$, and the detection result is obtained by:

$$\hat{c} = \underset{c_i}{\arg\max}\, v(x_{obj}, c_i), i = 1,2,...,N \tag{4}$$

$v\left(x_{obj}, c_i\right)$ represents the verification score between the tracking target x_{obj} and the candidate target c_i.

After obtaining the test results, we determine whether to use it as an alternative to the tracking result based on the verification score, as shown in Figure 3.

If the test result is unreliable (the verification score of the test result is less than the threshold τ_2), then we do not change the tracking result of the tracker; and the algorithm reduce the verification interval V, and enlarge the size of the local area to search for the target, repeat the above process until the detection To a reliable result. Then restore the verification interval and the size of the search area. Return the results from the validator to the tracker T and continue tracking down from the revised target new location. In order to effectively reduce the calculator calculation time, the algorithm chooses to verify every ten frames.

For our paper, the verification interval V is initially set to 10; the verification τ_1 and detection thresholds τ_2 are set to 1.0 and 1.6 respectively. The parameter β is initialized to 1.5 and can be adaptively adjusted based on the detection result.

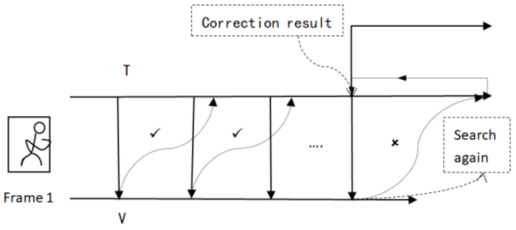


Figure 3: Tracking-Verification.

3 EXPERIMENTAL VERIFICATION AND RESULTS ANALYSIS

3.1 Experimental Configuration

In order to evaluate the tracking performance and efficiency of the proposed algorithm, the experimental results in this paper are based on the Core i7, 3.6GHz CPU, Win10 system, through the Matlab R2016a software testing OTB13 dataset Obtained using the algorithm of this paper. The test dataset contains attributes such as lighting changes, occlusion, fast movement, scale changes, motion blur, and in-plane rotation. In the experiment, this paper selects 10 algorithms to compare the result (DAT, DCF_CA, DSST, SAMF, MEEM, KCF, LCT, Staple, STAPLE_CA and Our), and then “ Our ” represents the algorithm that we proposed.

3.2 Quantitative Analysis

Quantitative analysis is a commonly used standard for measuring algorithm tracking results. This section uses the average center position error (CLE) and overlap rate (OR) to evaluate the performance of the algorithm. CLE is the Euclidean distance between the target's true center position and the center position calibrated by the tracking algorithm. The overlap ratio of the tracking is the ratio of the area where the tracking succeeds to the real bounding box:

$$\text{Score} = \frac{Area(B_T \cap B_G)}{\text{Area}(B_T \cup B_G)}$$

Where B_T represents the tracking target frame of each frame, and B_G represents the real bounding box of the corresponding frame.

Table 1 and Table 2 show the comparison results of the center position average error and the average value of the tracking bounding box overlap rate of the tracking results of different algorithms in each video sequence, respectively.

Table 1: Center position average error.

	KCF	LCT	Staple	MEEM	STAPLE_CA	TLD	DSST	ours
Basketball	7.89	***6.38***	18.51	19.25	62.6	253.6	10.8	**5.12**
CarDark	5.66	4.26	1.29	1.81	***1.12***	28.32	**1.03**	**1.03**
David3	5.52	5.32	4.16	7.02	***3.41***	265.6	88.5	**3.18**
Football1	5.19	5.20	3.52	4.26	**3.37**	99.33	8.74	***3.49***
Lemming	83.93	17.08	158.45	14.49	***5.89***	22.70	81.9	**4.87**
Deer	23.41	21.74	***6.03***	9.59	52.4	71.37	16.7	**4.72**
Jogging-1	145.0	***7.74***	151.1	13.19	60.8	12.39	110	**3.35**
Girl	10.91	4.69	13.0	***4.34***	60.1	9.90	10.6	**3.48**
MountainBike	**5.41**	6.70	***6.20***	8.15	54.0	185.7	7.72	7.73

Table 2: Average value of bounding box overlap.

	KCF	LCT	Staple	MEEM	STAPLE_ CA	TLD	DSST	ours
Basketball	0.676	*0.744*	0.676	0.631	0.346	0.020	0.673	**0.949**
CarDark	0.628	0.683	0.869	0.836	*0.931*	0.423	*1.000*	**1.000**
David3	0.749	0.756	*0.775*	0.692	0.933	0.096	0.532	**1.000**
Football1	0.724	0.670	0.719	0.709	*0.973*	0.385	0.419	**1.000**
Lemming	0.347	0.701	0.232	0.650	**0.734**	0.500	0.272	*1.000*
Deer	0.600	0.606	0.773	0.698	0.338	0.606	*0.789*	**1.000**
Jogging-1	0.113	*0.671*	0.124	0.547	0.231	0.626	0.225	**0.971**
Girl	0.562	0.691	0.534	*0.698*	0.380	0.555	0.314	**0.904**
MountainBike	0.727	0.733	*0.756*	0.677	0.140	0.217	*1.000*	**1.000**

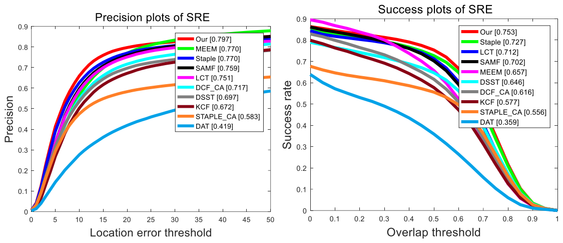
Note: The best and second best results are marked in bold and black italics, respectively

In general, the smaller the average error and the larger the overlap rate, the more accurate the tracking result. According to the results of the average position error of the center position in Table 1 and the average value of the overlap rate of the bounding box in Table 2, the average error of the center position of the target and the tracking frame overlap in the tracking process of the algorithm in this paper. The rate performance is better than the benchmark algorithm STAPLE_CA, especially in the case of the rotation of the target plane, the partial occlusion of the target, and the disorder of the target background, the robustness is improved.

3.3 Qualitative Analysis

This paper uses the OTB13 evaluation benchmark to perform three experiments on 51 video sequences: One-pass Evaluation (OPE), Temporal Robustness Evaluation (TRE), and Spatial Robustness Evaluation (SRE) Experiments. All these evaluation indicators represent the performance of the tracker in the form of an accuracy map and a success rate diagram, which means that the tracker can successfully track the percentage of the total number of frames in the video at different thresholds.

By testing 51 video sequences, the experimental results of the accuracy score map (a) and the success score graph (b) of the obtained SRE are shown in Fig. 4. From the experimental results in Fig. 4, the legend illustrates the ranking scores for each tracker, and our algorithm ranks first on the top. From the legend can be analyzed that the performance of the proposed algorithm is improved compared with the other nine different types of algorithms. Compared with the benchmark algorithm STAPLE_CA, although the tracking speed is about half of the benchmark algorithm, the average accuracy score and the average AUC score performance are improved by more than 10%.

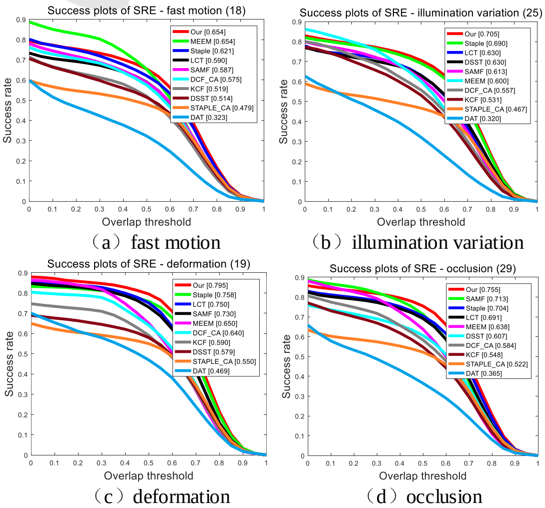


(a) accuracy score map (b) success score map

Figure 4: OTB13 video sequence algorithm evaluation results. The legend illustrates the ranking scores for each tracker, and our algorithm ranks first on the top(in SRE Evaluation Experiments).

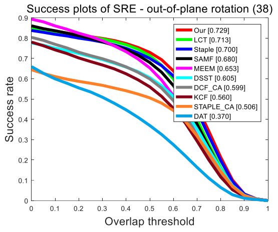
The 51 video sequences provided in OTB13 contain 11 attributes: illumination changes, occlusion, fast motion, scale changes, motion blur, and in-plane rotation. Fig 5(a)-(e) represent test results for partial attribute success rates of a video sequence.

Through the analysis of the OTB13 video sequence success rate evaluation graph of Fig. 5, it can be obtained that the algorithm which calculates the channel reliability for each feature channel in the input correlation filter, and adds the deep learning dual input Siamese network to the correlation filter, has attributes ranked first in condition of fast motion, deformation, illumination variation, occlusion, out of plane rotation. So compared with other algorithms, the algorithm has certain advantages, and the performance has been improved to some extent. Especially in the condition of tracking target fast movement, illumination change, target deformation, occlusion, and out-of-plane rotation, the algorithm is more advantageous.



（a）fast motion （b）illumination variation

（c）deformation （d）occlusion



（e）out of plane rotation

Figure 5: OTB13 video sequence success rate evaluation. The success plots of ten challenging attributes. The legend illustrates the ranking scores for each tracker. Our algorithm has attributes ranked first in condition of fast motion, deformation, illumination variation, occlusion, out of plane rotation.

4 CONCLUSIONS

In this paper, the channel reliability method is used to calculate the reliability weight of each feature channel and weighted to make the target location more accurate. The depth-learned dual-input Siamese network is used to verify and re-search the results of the correlation filter.

Through the evaluation benchmark analysis of OTB video sequences, the experimental results show that the algorithm has a certain degree of performance for fast motion, illumination change, target deformation, occlusion, and target rotation outside the plane.

ACKNOWLEDGEMENTS

This research and Algorithm implementation in this paper is supported by the 2018 Aviation Funds of China.

REFERENCES

A. W. Smeulders., D. M. Chu., R. Cucchiara., S. Calderara., A. Dehghan., and M. Shah., 2014. *The Journal*. *IEEE TPAMI*. Visual tracking: An experimental survey.

S. Hare., S. Golodetz., A. Saffari., V. Vineet., M.-M. Cheng., S. L. Hicks., and P. H. Torr., 2016. *The Journal*. *IEEE TPAMI*. Struck: Structured output tracking with kernels.

H. Fan and J. Xiang., 2017. *The Journal*. *IEEE TCSVT*. Robust visual tracking with multitask joint dictionary learning.

H. Fan and H. Ling., 2017. *The conference*. *CVPRW*. SANet: Structure-aware network for visual tracking.

C. Ma., J. B. Huang., X. Yang., and M.H. Yang., 2015. *The conference*. *ICCV*. Hierarchical convolutional features for visual tracking.

H. Nam and B. Han., 2016. *The conference*. *CVPR*. Learning multi-domain convolutional neural networks for visual tracking.

D. S. Bolme., J. R. Beveridge., B. Draper., Y. M. Lui., 2010. *The conference*. *IEEE Conference on Computer Vision and Pattern Recognition*, *CVPR*. Visual object tracking using adaptive correlation filters.

L. Bertinetto., J. Valmadre., S. Golodetz., O. Miksik., and P. H. S. Torr., 2016. *The conference*. *The IEEE Conference on Computer Vision and Pattern Recognition*. Staple: Complementary learners for real-time tracking.

T. Zhang. A. Bibi. and B. Ghanem., 2016. *The conference*. *CVPR*. In defense of sparse tracking: Circulant sparse tracker.

M. Danelljan., G. Hager., F. Shahbaz Khan., and M. Felsberg., 2015. *The conference*. *IEEE International Conference on Computer Vision*. Learning spatially regularized correlation filters for visual tracking.

Matthias Mueller, Neil Smith, Bernard Ghanem., 2017. *The conference*. *CVPR*. Context-Aware Correlation Filter Tracking.

C. Ma, X. Yang, C. Zhang, and M.H. Yang., 2015. *The conference*. *CVPR*. Long-term correlation tracking.

D. Comaniciu, V. Ramesh, and P. Meer. 2000. *The conference*. *CVPR*. Real-time tracking of non-rigid objects using mean shift.

F. Perronnin, J. Sanchez., and T. Mensink., 2010. *The conference*. *ECCV*. Mensink. Improving the fisher kernel for large-scale image classification.