Journal of computer science and software applications
https://www.mfacademia.org/
ISSN:2377-0430
Vol. 2, No. 3, 2022

Application of an Improved Kernel Correlation Filter Algorithm for Video Tracking in Complex Environments

Rajesh Sharma[1],Qi Sun[2]

Pacific University[1],Pacific University[2]

Rajeshss@gmail.coom[1],sunqiq@pacific.edu[2]

Abstract:

The extensive deployment of video surveillance has underscored the importance of efficient video information processing. Tracking moving objects in video footage constitutes a crucial aspect of this process. The Kernel Correlation Filter (KCF) algorithm is widely employed in the domain of video processing due to its rapid tracking speed and high efficiency. However, its performance can be significantly hindered by environmental factors such as lighting variations, changes in scale and shape, and motion blur. This paper presents an enhanced version of the KCF algorithm, incorporating reliability estimation and a re-detection mechanism to address issues related to severe occlusion and rapid target movement. Additionally, the scale pool method is introduced to improve adaptability. Simulation experiments and real-world video testing demonstrate that the improved KCF algorithm can effectively track moving objects under adverse environmental conditions, yielding satisfactory results.

Keywords:

Video tracking; Kernel Correlation Filter(KCF); Environmental Effect; Scale Adaptation.

1. Introduction

As the society develops rapidly, video surveillance systems are becoming increasingly mature, and are extensively applied in multiple aspects, whether in streets or high-rise buildings. In Guangzhou, for example, according to data released by Guangzhou officials in mid-2015, the number of surveillance cameras in the public domain held by the public security department is 500000, while the number in the private sector must be more. Accordingly, it is estimated that there are at least 1 million surveillance cameras in Guangzhou. The accompanying problem is that the daily amount of video data cannot be processed well, because the 720P camera stores about 800-1200M of video for one hour, and about 24G a day; the 960P camera stores about 1G-1. 5G for one hour, about 30G a day; while the 1080P camera stores about 1. 6-2G of video for one hour, about 45G a day. Assuming that 960P is covered once a week, Guangzhou generates 210 million gigabytes of data every week. It would be powerless to examine these data manually. If there is an algorithm that only needs to provide a data set and can actively find and track the target, it will significantly improve the efficiency and save manpower. This algorithm has appeared, and this video tracking algorithm is the KCF algorithm.

Kernel Correlation Filter (KCF) was proposed by Joao F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista in 2014. The algorithm was also a sensation at the time, because it had a very impressive performance in terms of tracking effect and tracking speed. Subsequently, a large number of scholars have studied it and the industry has successively applied it in practical scenarios.

2. Development Status of KCF Tracking Algorithm at Home and Abroad

The KCF algorithm collects positive and negative samples through the circulant matrix of the area around the target, trains the target detector by ridge regression, and successfully transforms the operation of the circulant matrix into the Hadamad product of the vector, that is, the point multiplication of elements, through the diagonalization of the circulant matrix in the Fourier space, which not only significantly reduces the amount of computation and improves the speed of operation, but also enables the algorithm to meet the real-time requirements. In addition, it maps the ridge regression of linear space to nonlinear space through kernel function, and solves a dual problem and some common constraints in nonlinear space. Similarly, circulant matrix Fourier space diagonalization can be used to simplify the calculation.

KCF algorithm is a kernel correlation filtering algorithm and a discriminant tracking method. It mainly trains a target filter in the tracking process, uses the target filter to detect the position of the target in the next frame, and then uses the newly detected results to update the training set and then update the target filter in the next frame. The specific implementation process of the algorithm is divided into three parts: training, detection and update [1,2].

2.1. Training

The training process of kernel correlation filtering algorithm mainly involves three parts: ridge regression, cyclic matrix and kernel function.

a. Ridge regression

The training aims to determine a ω through the sample set so that the response value of the objective function $f(\vec{x}) = \omega^T \vec{x}$ on the sample vector $\vec{x_i}$ has the smallest square error with the regression target y_i, the formula is:

$$\min_{\omega} \sum_i (f(x_i) - y_i)^2 + \lambda \|\omega\|^2 \quad (1)$$

Where, λ represents the regularization coefficient, in order to prevent the filter from overfitting; the superscript $\rightarrow$ represents the vector.

In order to facilitate the solution, convert the above formula into matrix form:

$$\min_{\omega} \|X\omega - y_i\|^2 + \lambda \|\omega\|^2 \quad (2)$$

Where, X represents the sample matrix composed of sample vectors, and $X = [x_1, x_2, \dots, x_n]^T$, y represents the regression vector composed of regression target y_i, $Y = [y_1, y_2, \dots, y_n]^T$ [3]. In order to find the minimum value, if the derivative of Eq. (2) on ω is 0, it is obtained that:

$$\omega = (X^T X + \lambda I)^{-1} X^T y \quad (3)$$

Where, I represents a unit matrix with the same dimension as X. Convert Eq. (3) to plural form:

$$\omega = (X^H X + \lambda I)^{-1} X^H y \quad (4)$$

Where, the superscript H is expressed as a conjugate transpose, and Eq. (4) includes the process of finding the inverse of the matrix. Because the amount of computation of matrix inversion increases exponentially when the matrix dimension is large, the running speed of the algorithm will be greatly reduced. Therefore, the circulant matrix is introduced into the algorithm to replace the inverse of the matrix.

b. Circulant matrix

The sample matrix used in the training process is obtained by cyclic displacement of the target sample, and the cycle of the target sample can be obtained from the permutation matrix. The

n × 1 sample images are represented by the vector $\vec{x} = [x_1, x_2, x_3, \dots, x_n]^T$ and the cyclic displacement of the sample vector in the vertical direction is based on the matrix *P* [5]:

$$P = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \quad (5)$$

The vector x is shifted once in the vertical direction and can be obtained by $Px = [x_n, x_1, x_2, \dots, x_{n-1}]^T$, and the vector after *n* times of translation is the same as x. Putting the n vectors together forms a one-dimensional cyclic matrix. The schematic diagram of its effect is shown in Fig. 1, where C represents cyclic displacement of the data in parentheses.

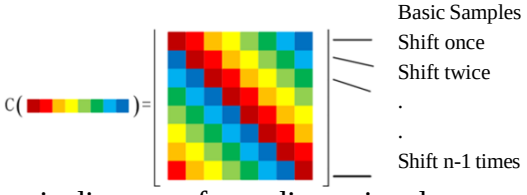


Fig 1. Schematic diagram of one-dimensional vector cyclic shift

For two-dimensional images, the training samples can be obtained from the horizontal and vertical cyclic shift, in which the cyclic displacement of the sample vector in the horizontal direction is based on the matrix *Q*:

$$Q = \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \quad (6)$$

The image matrix X can be translated u times and v times vertically and horizontally by P^uXQ^v. The cyclic shift diagram of the two-dimensional image in the horizontal and vertical directions is shown in Fig. 2, in which the middle one is the initial image [5].

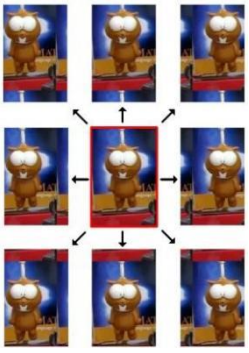
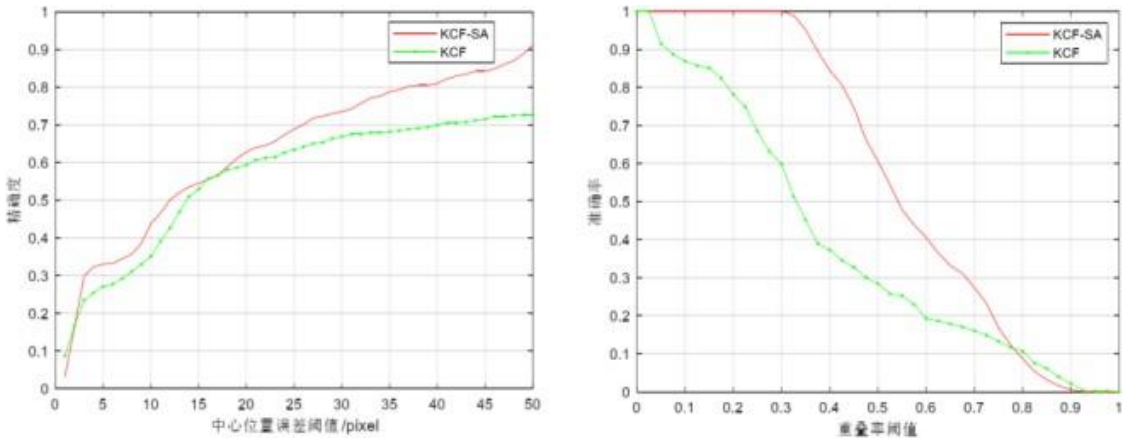


Fig 2.Schematic diagram of cyclic shift of two-dimensional image.

According to Eq. 4, the inverse operation is carried out to calculate the solution of ω. It is impossible to realize real-time calculation because of the large amount of calculation. A new ω formula is obtained by diagonalizing X by Discrete Fourier Transform (DFT), as shown in Eq. (7) . The original inverse operation with a large amount of calculation is changed into a dot product and point division operation in the frequency domain. This kind of method significantly reduces the amount of computation for solving ω, thus ensuring the real-time performance of KCF algorithm.

$$\hat{\omega} = \frac{\hat{\vec{x}} \odot \hat{\vec{y}}}{\hat{\vec{x}}^* \odot \hat{\vec{x}} + \lambda} \quad (7)$$

In the actual smart kitchen target tracking, multiple adjustments such as target deformation, illumination changes, and occlusions will inevitably occur due to the complexity and variability of the kitchen environment. The tracking effect of the traditional KCF algorithm is not good in the case of serious target occlusion and fast movement. In this project, the improved KCF algorithm is proposed, and the reliability estimation and re-detection mechanism are added to the algorithm to solve the problem of serious occlusion and fast motion of the target[7]. The comparison of accuracy and accuracy between the improved KCF algorithm and the traditional KCF algorithm is shown in Fig. 3.



(a) Accuracy (b) Accuracy

Fig 3 Comparison of accuracy and accuracy between improved KCF and KCF

When determining the evaluation model of personnel and material state, fuzzy comprehensive evaluation is introduced because there are many evaluation index parameters and the ambiguity of each factor, but the results of fuzzy comprehensive evaluation often have a certain degree of subjectivity. It synthesizes by improving KCF and fuses multi-class state evaluation results by taking advantage of filtering methods in dealing with uncertain information. In addition, it effectively avoids the occurrence of contradictory phenomena when high conflict evidence is fused by modifying the basic probability assignment, and improves the accuracy of the evaluation results.

3. Limitations

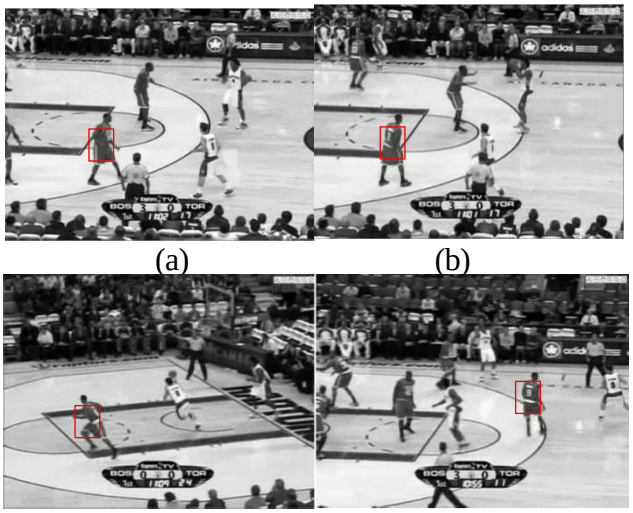
But over time, KCF's experimental results in various scenes are not satisfactory.

First, the size of KCF does not change from beginning to end because the target frame has been set during the tracking process. However, the size of the target in the tracking sequence changes from time to time, which causes the target frame to drift during the tracking process of the tracker, resulting in tracking failure.

Second, KCF does not solve the problem when the target is obscured in the tracking process, but this problem has always been a big problem in the tracking community.

Third, when sufficient samples are selected, the amount of computation is too large to ensure the real-time performance of the tracking algorithm. Therefore, most detection-based algorithms sacrifice the number of samples to ensure the real-time performance of the algorithm, which leads to poor robustness of the tracking algorithm[8].

The following image is a tracking image obtained without an improved algorithm, such as Fig.4.



(a) (b)

(c) (d)

Fig 4 Track the original image

This algorithm realizes tracking, but the tracking is extremely vague. Sometimes the fixed target box can completely capture the tracking target, and sometimes it can only capture part of the tracking target's torso. This is mainly due to the inaccuracy of the data set and the need to scale the initial test target. After the correlation filter, the corresponding response response is obtained, and then the sizes of different scales of response are compared to determine the largest one, which is the optimal target scale value. The optimal scale value is applied to track the target, which solves the problem of target scale change in the process of target tracking[9].

4. Experimental Improvement and Analysis

4.1. Experimental Improvement and Analysis

The improvement of the scale is also the core part of this work. The KCF algorithm uses a single scale, which will lead to the decline of tracking accuracy when the target is deformed and occluded. The scale pool method is introduced into the SAMF algorithm. The scale pool is PoolS={0. 985, 0. 99, 0. 995, 1. 0, 1. 005, 1. 01, 1. 015}. Its idea is very simple, which is to calculate seven scales for the target of the candidate region in the comparison stage. Compared with the target in the previous frame, the target with the largest response value is determined as the target in the current frame.

Depth estimation will inevitably lead to errors. In order to avoid this error, the actual size is used to correct the calculated size. The Eq. (8) is as follows:

$$S = \text{scale} \cdot \frac{\text{trueScale}}{\text{scale}} \tag{8}$$

In the formula, trueScale represents the size value of the tracking area in the first picture, scale represents the scale value, and S is the scale value we have calculated[10].

The schematic diagram of the algorithm is as follows, such as Fig.5 and Fig.6.

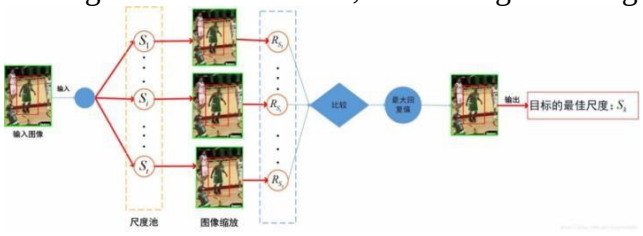


Fig 5 Schematic diagram of pool

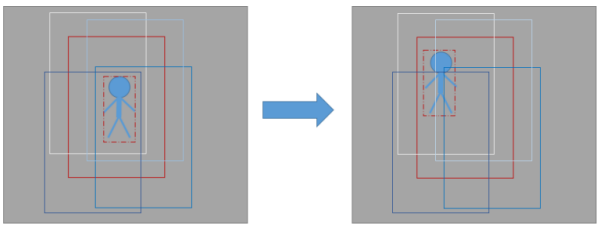
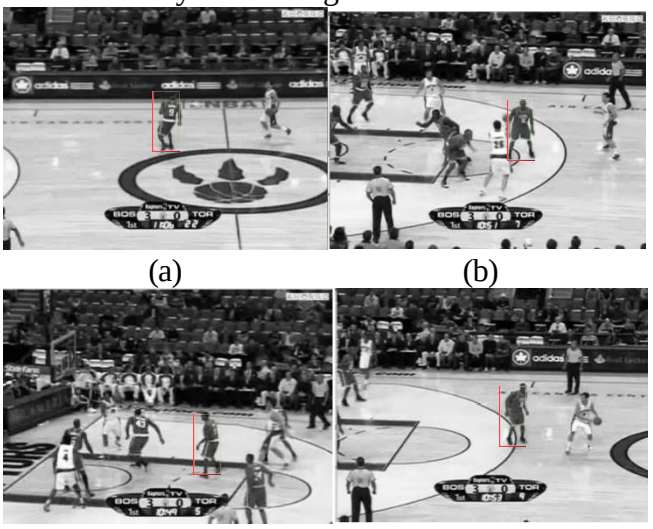


Fig 6 Accurate positioning map

The tracking effect of the same video after processing has been greatly improved. The following figure suggests that the method of introducing scale pool class can realize scale adaptive tracking in a small range and improve the accuracy of tracking.



(a) (b)

(c) (d)

Fig 7 Modification drawing

In the image above, the tracking algorithm is no longer confused because it disappears from the field of view, blurs the figure, is obscured, rotates the back, changes the resolution, and has a similar background, but only tracks the torso or missing. Each frame of the picture can firmly grasp the tracking target.

According to the selection of features, excellent features are the basis for accurate tracking. The earliest MOSSE algorithm uses a single grayscale feature. Due to the single feature, the processing speed of the MOSSE algorithm is very fast, reaching 669FPS/S, but its accuracy is very low, less than 0.5[11]. Subsequently, the color feature (CN) is used, and the accuracy of the algorithm is greatly improved; KCF algorithm uses directional gradient histogram (HOG) feature, which improves the accuracy to about 0.7 (because there is a certain deviation in different data sets, there is no specific and accurate value), and the speed also reaches 172FPS; after that, it also starts to look for high-quality feature expression, but there is no obvious result. Therefore, the method of fusion of multiple features is considered to improve the accuracy of tracking, and the gray fusion feature of HOG+CN+ is used in SAMF [12]. It's a simple vector superposition [13].

The Eq. (9) is as follows [14]:

$$K^{XX'} = \exp\left(-\frac{1}{\sigma^2}\left(\|X\|^2 + \|X'\|^2\right) - 2F^{-1}\left(\hat{X} \odot \hat{X}'^*\right)\right)$$

$$K^{XX'} = \exp\left(-\frac{1}{\sigma^2}\left(\|X\|^2 + \|X'\|^2\right) - 2F^{-1}\left(\sum_C \hat{X}_C \odot \hat{X}_C'^*\right)\right) \tag{9}$$

X is a single feature extracted in the traditional KCF algorithm, while Xc is a mixture of three features. The algorithm is equivalent to a simple vector superposition of the three features[15].

4.2. Experimental Analysis

The calculation of the ring matrix directly converts all the samples into a diagonal matrix for processing, because when the cyclic matrix deals with the samples, the samples are not real samples, only virtual samples exist. Therefore, we can directly use the unique characteristics of the circulant matrix to directly transform the sample matrix into a diagonal matrix for calculation. This can greatly speed up the calculation between matrices, because the operation of diagonal matrices only needs to calculate the values of non-zero elements on the diagonal. Eq. 7 expresses the eigen decomposition of a general circulant matrix. The shared, deterministic eigenvectors Flie at the root of many uncommon features, such as commutativity or closed- form inversion[16]. To put it simply, it performs feature decomposition, that is, it simplifies operations and speeds up speed.

The correlation filter is improved according to the previous MOSSE algorithm, which can be said to be the ancestor of CSK, STC, Color Attributes and other tracker. Correlation Filter originated from the field of signal processing and was later used in image classification[17].

The simplest idea of applying Correlation Filter to tracking is that correlation is a measure of the similarity between two signals[18]. The more similar the two signals are, the higher the correlation value is. In the application of tracking, it is necessary to design a filter template to maximize the response when it acts on the tracking target[19]. The position of the maximum response value is the position of the target. Correlation filtering was originally a signal processing thing, and it was introduced into tracking after the publication of MOSSE. The speed and accuracy of the calculation are very good, so it has achieved good results[20].

5. Conclusion

KCF algorithm is an excellent target tracking algorithm, but it is not perfect in complex environment. This work improves the KCF algorithm and integrates the MOSSE algorithm and the SAMF algorithm, which improves the speed of the algorithm, can be adjusted according to the situation, and can track the target for a long time. In addition, it adds HOG features to solve the problem of poor tracking results of the algorithm in fuzzy processing. Experimental results show that the robustness and tracking speed of the proposed algorithm are better than the original KCF algorithm.

References

[1] Li J, Wei J, Jiang J, Lu Y, Liu L, Tang Y, Li X. A method for extracting spatiotemporal information of dynamic targets in multi-view surveillance video[J/OL]. Journal of Surveying and Mapping: 1- 20[2022-03-09]. http://kns. cnki. net/kcms/detail/11. 2089. P. 20220113. 1740. 002. html

[2] Ouyang G, Zhong B, Bai B, Liu X, Wang J, Du J. Application and latest research progress of deep neural network in target tracking algorithm [J]. Small Microcomputer System, 2018, 39(02):315 -323.

[3] Guan H, Xue X, An Z. A Video Target Tracking Method Using Online Convolutional Networks [J]. Small Microcomputer Systems, 2017, 38(04):872-875.

[4] Geng Y, Hu H, Meng Y, Shi Q, Zhang W. Research on intelligent early warning method of sudden large passenger flow in subway stations based on video recognition[J]. Intelligent Computer and Application, 2022, 12(02):170-173+177.

[5] Liang J. Design of people density detection system based on video recognition[J]. Electronic Design Engineering, 2021, 29(23):152-157. DOI:10. 14022/j. issn1674-6236. 2021. 23. 031.

[6] Bai L. AI video recognition system easily captures [N]. Chongqing Daily, 2021-08-22(001). DOI: 10. 28120/n. cnki. ncqrb. 2021. 005709.

[7] Hu C. Research on Image and Video Recognition Technology Based on Single-node Photonic Storage Pool Computing[D]. Taiyuan University of Technology, 2021. DOI: 10. 27352/d. cnki. gylgu. 2021. 000707.

[8] Zheng H. Research on Video Recognition of Safety Protection Measures for Electric Power Construction Personnel[D]. Guangdong University of Technology, 2021. DOI:10. 27029/d. cnki. ggdgu. 2021. 000180.

[9] Li J, Dong Y, Du Z, Shen Y. Efficiency Analysis of Construction Elevator Based on Video Recognition[J]. Building Construction, 2021, 43(02):320-321+325. DOI:10. 14144/j. cnki. jzsg. 2021. 02. 048 .

[10] Chu K, Zhu L, Zhang J. Improved TLD Target Tracking Algorithm Integrating KCF and HOG[J]. Journal of Changzhou University(Natural Science Edition), 2022, 34(01):60-67.

[11] Liu T, Wang K. Aerial target tracking simulation based on KCF algorithm[J]. Laser and Infrared, 2021, 51(10):1396-1400.

[12] Hui K. Research on improved method of target tracking based on KCF [D]. Xi'an University of Technology, 2021. DOI: 10. 27398/d. cnki. gxalu. 2021. 000536.

[13] Yin L. Research on KCF method for scale adaptation and anti-occlusion [D]. Shenyang Jianzhu University, 2021. DOI: 10. 27809/d. cnki. gsjgc. 2021. 000278.

[14] Chu K, Zhu L, Zhang J. Improved TLD Target Tracking Algorithm Integrating KCF and HOG[J]. Journal of Changzhou University(Natural Science Edition), 2022, 34(01):60-67.

[15] Feng Z, Wang H, Liu J, Zhu L. Anti-Occlusion and Automatic Correction KCF Algorithm[J]. Journal of Shenyang University of Aeronautics and Astronautics, 2021, 38(05):44-50.

[16] Ye Q, Yuan L, Lv K. Scale-adaptive Kernel Correlation Tracking Algorithm for Feature Fusion [J]. Computer Engineering and Design, 2022, 43(02): 420-426. DOI: 10. 16208/j. issn1000-7024. 2022. 02. 017.

[17] Tan S, Sun X, Chan W, Qu L, Shao L. Robust Face Recognition With Kernelized Locality-Sensitive Group Sparsity Representation[J]. IEEE Transactions on Image Processing, 2017, 26(10): 4661- 4668.

[18] Tao Q, Zhan S, Li X. Robust face detection using local CNN and SVM based on kernel combination[J]. Neurocomputing, 2016, 211: 98-105.

[19] Xia Y, Zhang B, Coenen F. Face occlusion detection using deep convolutional neural networks[J]. Pattern Recogn Artif Intell, 2016, 30(9): 401-408.

[20] Cui J, Zhang H, Han H, Shan H, Chen X. Improving 2D Face Recognition via Discriminative Face Depth Estimation[J]. 2018 International Conference on Biometrics(ICB), 2018: 140-147.

[21] Mery D, Banerjee S. Recognition of Faces and Facial Attributes Using Accumulative Local Sparse Representations[J]. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018: 1947-1951.

[22] W. Zuo, X. Wu, L. Lin, et al. Learning Support Correlation Filters for Visual Tracking[J].