Vehicle tracking based on video in fast traffic scenarios

Zheng Liao\textsuperscript{a}, Gang Li\textsuperscript{b}, Xinyue Zhang\textsuperscript{c}

Information Engineering, Chang’an University, Xi’an, Shaanxi, China

\textsuperscript{a}lovelz5566@163.com, \textsuperscript{b}1723915372@qq.com, \textsuperscript{c}870907776@qq.com

Abstract. This paper is aiming at the problem of low tracking accuracy of moving vehicles in fast traffic scenarios, the ECO algorithm based on correlation filtering is applied to real-time tracking to achieve stable and accurate tracking of target vehicles. Through the factorized convolution operator, the extracted features are more comprehensive and efficient; the optimization of the training set effectively reduces redundancy; and the more representative correlation filter is used to prevent over-fitting; a simple update model is used to prevent model drift. The experimental results show that the proposed method finally achieved a tracking effect of 97% recognition rate.

1. Introduction

In recent years, video tracking technology has been widely used in many fields of civil and military. It has attached great importance in the past ten years. Accurate and robust tracking of vehicles in traffic scenarios is a challenging task. First of all, the outdoor environment is complex and changeable, and the vehicle tracking should adapt to the dynamic changes of the scene, especially the changes of lighting conditions, such as sudden changes in light caused by moving clouds, weather changes, day and night switching, etc.; Second, there are interferences from other objects in the scene, such as shadows, past crowds, specular reflections of the vehicle in the scene, etc.; In addition, there are a large number of vehicles in the surveillance scene. While the vehicle's entry and exit in the scene, scale changes, viewing angle changes, and occlusion are all very difficult for vehicle tracking.

The target tracking algorithm is the key technology of the vehicle tracking system. The traditional target tracking algorithms are generally classified into three categories: First is based on the region matching tracking method [1], and the same feature information contained in the moving targets in the tracking region is tracked; Second is based on the feature matching tracking method [2]. Selecting a suitable feature as the template for the moving target in the tracking area, and then extracting the feature information from the target. And comparing the extracted feature information with the template to determine whether the target is the tracking one; there is also a tracking method based on model matching [3]. The core of this method is to determine the structure and model of the moving target based on past experience, and then determine the parameters based on the results to find the target.

However, traditional algorithms are currently unable to adapt to the changes of the complex tracking targets [4]. In recent years, the method of deep learning has achieved success in the field of image recognition and detection with its excellent performance, and has breakthroughs in the application of computer vision classics, such as image classification [5] and target detection [6]; Deep learning has also been deeply studied in the field of visual target tracking [7-9]. From the initial use of deep networks to extract object adaptive features, and then merge other tracking methods to achieve target tracking, and now it has been able to train an end-to-end deep neural network model to directly infer the target...
position. At the same time, more and more deep neural network models, such as recurrent neural network (RNN), automatic coding machine (ADE), and convolutional neural network (CNN), have been applied to the target tracking field, and have achieved good results. The eco tracking algorithm [10] in this paper traces the vehicle through multiple features such as hog features, cnn features, and correlation filtering. And the network is updated with an improved update scheme to achieve stable and reliable tracking of the target vehicle, especially for remote vehicle targets.

Figure 1. Pipeline of FCNT algorithm. (a) Input ROI region. (b) VGG network. (c) SNet. (d) GNet. (e) Tracking results.

2. Target Tracking Based On Deep Learning
In the existing deep learning-based target tracking method, the DLT[11] network proposed by Naiyan Wang et al. first applied the deep learning model to the target tracking, and propose the idea of “offline pre-training + online fine-tuning”, which greatly solved the problem of insufficient training samples in tracking. However, DLT has a low resolution of the data set for offline training and cannot learn a robust matching representation of the matching tracking sequence. In 2015, based on DLT, SO-DLT (structured output deep learning tracker) was proposed. Using a network structure similar to AlexNet, the model uses the structural information of the image itself to directly determine the location of the target frame from the probability map.

Wang et al. [12] design a Fully Convolutional Network Based Tracking (FCNT) algorithm based on the VGG-16 network. The algorithm is based on the feature maps of Conv4-3 and Conv5-3 two-layer output, and the network is selected to train and extract effective features to avoid over-fitting the noise and reduce the feature dimension. Then, the selected features are transported to the respective positioning networks to obtain a heat map, and the heat maps of the two positioning networks are combined to obtain the final tracking result. The algorithm complements the different layer features to achieve effective suppression of tracking frame drift and is more robust to apparent changes in the target. Its framework is shown in Figure 1.

Nam et al. [13] propose a multi-domain network (MDNet). MDNet applies all sequences of video to pre-training for target tracking. The network uses uncorrelated image data to reduce the training needs of tracking data. In 2015, Martin Danelljan's Convolutional Features for Correlation Filter Based Visual Tracking (Deep-SRDCF) [14] was based on KCF to extract features using a deep convolutional network. In 2016, David Held proposed Learning to Track at 100 FPS with Deep Regression Networks (GOTURN) [15] to directly use the offline training + online tracking solution to increase the speed directly to 100fps.

In the literature [16], studying the deep structure of different convolutional networks and comparing the performance of the network, it is concluded that the network structure of VGG-M has faster detection speed and stronger performance. The structure of VGGNet is very simple, as shown in TABLE 1. The entire network uses the same size convolution kernel size (3*3) and maximum pool size (2*2). It consists of 5 layers of convolutional layer, 3 layers of fully connected layer, and softmax output layer. The layer-to-layer is separated by max-pooling, and the activation units of all hidden layers use the ReLU function.
Among them, the small convolution kernel is an important feature of VGGNet. VGGNet uses multiple convolutional layers of smaller convolution kernels (3x3) instead of a convolutional layer with a larger convolution kernel. On the one hand, parameters can be reduced. On the other hand, it is equivalent to performing more nonlinear mapping, which can increase the fitting/expression ability of the network. The VGG-M network is a medium-architecture convolutional network whose network structure is shown in Table 1. Its good target feature expression and reasonable network architecture make it more suitable for target tracking. In this paper, the eco algorithm extracts the feature subsets by using the combination of the first (Conv-1) and the last (Conv-5) convolutional layer in the vgg-m network to reduce the dimension and achieve the tracking of the target vehicle. This method has stronger robustness and faster speed and improves the problem of overfitting.

3. Target Vehicle Tracking Algorithm

ECO is based on C-COT [17] and is an algorithm based on correlation filtering. The core idea of the correlation filtering algorithm is to match the target template with the image block obtained by the sliding window in the search area, and respond to the corresponding image block at the maximum position as the target image block. Compared with the traditional features, the image convolution features extracted by the deep neural network have good anti-interference ability. Using neural networks to simulate the entire process of correlation filtering, the filter coefficients can be converted to a layer of neural networks, and then deriving the formula of forward and backward propagation to achieve end-to-end training of the network, the algorithm speed can reach tens or even hundreds of frames per second.

| Name  | Type      | Filter Size \( \times \) Stride | Output Size |
|-------|-----------|-------------------------------|-------------|
| conv1 | Convolution| \( 7 \times 7 \times 2 \)      | \( 109 \times 109 \times 96 \) |
| relu1 | ReLU      |                               | \( 109 \times 109 \times 96 \) |
| norm1 | LRN       |                               | \( 109 \times 109 \times 96 \) |
| pool1 | Max-Pooling| \( 3 \times 3 \times 2 \)      | \( 54 \times 54 \times 96 \) |
| conv2 | Convolution| \( 5 \times 5 \times 2 \)      | \( 26 \times 26 \times 256 \) |
| relu2 | ReLU      |                               | \( 26 \times 26 \times 256 \) |
| norm2 | LRN       |                               | \( 26 \times 26 \times 256 \) |
| pool2 | Max-Pooling| \( 3 \times 3 \times 2 \)      | \( 13 \times 13 \times 256 \) |
| conv3 | Convolution| \( 3 \times 3 \times 1 \)      | \( 13 \times 13 \times 512 \) |
| relu3 | ReLU      |                               | \( 13 \times 13 \times 512 \) |
| conv4 | Convolution| \( 3 \times 3 \times 1 \)      | \( 13 \times 13 \times 512 \) |
| relu4 | ReLU      |                               | \( 13 \times 13 \times 512 \) |
| conv5 | Convolution| \( 3 \times 3 \times 1 \)      | \( 13 \times 13 \times 512 \) |
| relu5 | ReLU      |                               | \( 13 \times 13 \times 512 \) |
| pool5 | Max-Pooling| \( 3 \times 3 \times 2 \)      | \( 6 \times 6 \times 512 \) |

3.1. Factorized Convolution Operator

ECO simplifies feature extraction, factoring convolution operations, and reducing model parameters. The model in this article is based on C-COT. Interpolation is performed by \( t \) in C-COT to convert the Feature Map to a continuous spatial domain. The result \( J(x) \) is the extracted feature:

\[
J_d \{ x^d \}(t) = \sum_{n=0}^{N_d-1} x^d[n]b_d(t - \frac{T}{N_d}n)
\]
Here, $\tilde{b}_d$ is an interpolation kernel with period $T > 0$. The result $J_d \{x^d\}$ is thus an interpolated feature layer, viewed as a continuous $T$-periodic function. We use $J\{x\}$ to denote the entire interpolated feature map, where $J\{x\}(t) \in \mathbb{R}^D$.

According to the previous results, the score is calculated using the multi-channel convolution filter $f = (f_1 \ldots f_D)$ of continuous time period:

$$S_f \{x\} = f \ast J\{x\} = \sum_{d=1}^{D} f^d \ast J_d \{x^d\}$$  

(2)

Finally, the objective function is confirmed. The Gaussian function $y_j$ is constructed for the training sample $x_j$, the least square objective function is constructed by L2 Norm, and the weight penalty term $w$ is added to obtain the formula (3). Then perform Fourier transform through Parseval’s formula to get the formula (4).

$$E(f) = \sum_{j=1}^{M} \alpha_j \left| \left| S_f \{x_j\} - y_j \right| \right|^2_{L^2} + \sum_{d=1}^{D} \left| \left| \hat{w} \ast \hat{f}^d \right| \right|^2_{L^2}$$  

(3)

$$E(f) = \sum_{j=1}^{M} \alpha_j \left| \left| \hat{S}_f \{x_j\} - \hat{y}_j \right| \right|^2_{L^2} + \sum_{d=1}^{D} \left| \left| \hat{w} \ast \hat{f}^d \right| \right|^2_{L^2}$$  

(4)

Here, the hat $\hat{g}$ of a $T$-periodic function $g$ denotes the Fourier series coefficients $\hat{g}[k] = \frac{1}{T} \int_{0}^{T} g(t)e^{-\frac{2\pi i k t}{T}} dt$ and the $\ell^2$-norm is defined by $\left| \left| \hat{g} \right| \right|^2_{\ell^2} = \sum_{-\infty}^{\infty} \left| \hat{g}[k] \right|^2$. The Fourier coefficients of the detection scores (2) are given by the formula $\hat{S}_f \{x\} = \sum_{d=1}^{D} \hat{f}^d \hat{X}^d \tilde{b}_d$, where $X^d$ is the Discrete Fourier Transform (DFT) of $X^d$.

ECO extracts feature subsets for dimensionality reduction, uses a subset of the original features, and selects the C dimension from the D-dimensional features. The basic C-COT is that each dimension has a filter corresponding to it, and the D-dimensional feature has D filters, but in fact many filters have a small contribution, as shown in Figure 2. The ECO selects only C filters that contribute more, and then each feature is represented by a linear combination of the C filters.
Figure 2. Visualization of the learned filters corresponding to the last convolutional layer in the deep network

Figure 3. The training set

Thus, using the linear feature of the convolution, we can get the new score function (5) and the objective function (6) through a D*C coefficient matrix P:

\[
S_{gf} \{x_i\} = Pf^T J \{x_i\} = \sum_{c,d} p_{d,e} f^c \ast J \{x^d\} = f^T P f \text{ (5)}
\]

Where P is a matrix of DxC, each row represents the linear combination coefficient of all C filters for the filter corresponding to the feature of one dimension, and needs to be learned in the first frame, and remains unchanged in the subsequent tracking.

\[
E(f, P) = \|z^T P \hat{f} - \hat{y}\|^2 + \sum_{c=1}^{C} \| \hat{w} \ast \hat{f}^c \|^2 + \lambda \|P\|^2 \text{ (6)}
\]

Where z = J \{x_i\}, a final setting is added to constrain P. The purpose of the objective function (6) is to obtain a filter f that minimizes E based on the input training data.
3.2. Generative Sample Space Model

ECO classifies samples by Gaussian Mixture Model (GMM), which simplifies the training set and results in different Components. Figure 4.

The lower part of Fig. 3 is the traditional training set. Each time one frame is updated, one sample is added. Then, the samples in the training set after consecutive frames are highly similar, that is, it is easy to overfit the most recent frames. The upper part is the ECO approach. ECO uses the Gaussian Mixture Model (GMM) to generate different components. Each component basically corresponds to a group of similar samples, and there are significant differences between different components. This makes the training set a variety.

According to the joint probability distribution \( p(x, y) \) passing the sample \( x \) and the target output \( y \), a new objective function is obtained:

\[
E(f) = \mathbb{E}\left\{ \left\| S_f \{x\} - y \right\|_2^2 \right\} + \sum_{d=1}^{D} \left\| w_f^d \right\|_2^2
\]

(7)

Equation (7) is to replace the first term of equation (3) with the joint probability distribution based on the sample. In addition, since the \( y \) shape of the target output is consistent, it is a Gaussian function with a peak at the center of the target, but the position of the peak is not the same. Then ECO sets \( y \) to be the same, and the shift amount of the peak position is reflected on \( x \), which can be easily processed in the frequency domain. Then \( p(x, y) \) is simplified, just calculate \( p(x) \). Here we use GMM to model:

\[
p(x) = \sum_{l=1}^{L} \pi_l \mathcal{N}(x; \mu_l; I)
\]

(8)

Where \( L \) represents the number of components, reducing the previous \( M \) samples to \( L \).

The update process is as follows: each time a new sample comes in, initialize a component \( m \):

\[
\pi_m = \gamma \quad \text{and} \quad \mu_m = x_j
\]

(9)

If the number of components exceeds the limit \( L \), one with the least weight is discarded. Otherwise, merge the two most recent components \( k \) and \( l \):

\[
\pi_n = \pi_k + \pi_l, \quad \mu_n = \frac{\pi_k \mu_k + \pi_l \mu_l}{\pi_k + \pi_l}
\]

(10)

Finally, the approximate objective function (11) is obtained, and the original \( M \) samples are reduced to the average of \( L \) components.

\[
E(f) = \sum_{l=1}^{L} \pi_l \left\| S_f \{\mu_l\} - y_0 \right\|_2^2 + \sum_{d=1}^{D} \left\| w_f^d \right\|_2^2
\]

(11)
3.3. *Model Update Strategy*

The ECO uses a common interval frame update method, which simply stipulates that it is updated every Ns frame. The frequency of model update is reduced, of course saving time, and the drift of the model can be avoided, and the effect is improved to a certain extent, but Ns cannot be made too large, otherwise the model will not keep up with the change of the target.

4. *Experimental Result*

This article uses the actual traffic scene video to verify the results. The test video is taken from the camera on the West Long Expressway. The test video comes from the West Long Expressway. The camera that shoots the video belongs to the universal camera on the highway. The height of the camera is set to meet the general standards of the highway. In order to verify the versatility of the vehicle tracking algorithm, three different high-speed road segments were selected to capture the video, and the tracking results of various models were tested to ensure the final result.

After the location of the target vehicle is given in the first frame of the video, the target vehicle can be tracked stably and effectively. As shown in Figure 4-6, the experiment proves that the tracking algorithm has strong robustness in the tracking of the target vehicle.

![Figure 4. Scenarios 1](image1.png)

![Figure 5. Scenarios 2](image2.png)
Figure 6. Scenarios 3

Figure 7. Scenarios 4

Figure 8. Scenarios 5
Table 2. The detection result of different scenarios

| scenarios           | Total number of frames | the number of error detection frames | the frame detection rate% |
|---------------------|------------------------|--------------------------------------|----------------------------|
| Xi’an Xichang Road Scenarios 1 | 106                    | 3                                    | 97.17                      |
| Xi’an Xichang Road Scenarios 2 | 106                    | 2                                    | 98.11                      |
| Xi’an Xichang Road Scenarios 3 | 14                     | 0                                    | 100.00                     |
| Xi’an Xichang Road Scenarios 4 | 50                     | 2                                    | 96                         |
| Xi’an Xichang Road Scenarios 5 | 51                     | 2                                    | 96.08                      |
| Average detection rate |                        |                                      | 97.47                      |

Figure 9. The frame detection rate

As shown by the above results, the algorithm has a good detection effect on various models and different road sections, and can accurately identify the target vehicle when the similar color models overlap. Especially for the tracking of small targets at the far end of the video, the effect is stable and accurate, and the expected goal is achieved.

This paper uses the frame detection rate as the calculation method of vehicle detection accuracy. Compare the detection results of each frame, screen out the video frames of the vehicle, and record the false detection and the missed detection as error detection. The accuracy of the detection is obtained by
subtracting the number of error detection frames from the total number of frames and then comparing the total number of frames. The results are shown in TABLE 2 and Figure 9.

The experimental results show that the proposed algorithm has strong robustness when tracking target vehicles. The tracking accuracy is 97%, and the timeliness is good, especially for the small target at the far end.

5. Conclusion
In this paper, the ECO target tracking algorithm of deep learning can effectively track the target by extracting multiple features of the target and combining the filtered correlation filters to prevent overfitting. At the same time, the diversity of training samples reduces redundancy and speed. The algorithm has strong robustness and achieves the purpose of tracking the vehicle.

References
[1] Wang Shengnan, Yu Mei, Jiang Gangyi, et al. Overview of Vehicle Detection and Tracking Methods Based on Video Image Processing in Intelligent Transportation Systems. Journal of Computer Applications, 2005, 22 (9): 9 -14.
[2] S. SAHA, S. BASU, M. NASIPURI, et al. Localization of License Plates from Surveillance Camera Images: A Color Feature Based ANN Approach. International Journal of Computer Applications, 2010, 1 (23): 25.
[3] K. STEPHEN, F. MINORU, A. NORIO, et al. Detection and Recognition of Vehicle License Plates using Template Matching, genetic algorithms and neural networks. International Journal of Innovative Computing, Information and Control, 2009, 5 (7): 1975 - 1985.
[4] Gu Baiyuan, Wang Rongben, Guo Lie, et al. Research on multi-vehicle detection method on the road based on machine vision [J]. Automotive Engineering, 2006, 28 (10): 902 - 905.
[5] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 2012 International Conference on Neural Information Processing Systems. Lake Tahoe, USA: MIT, 2012. 1097 − 1105.
[6] Girshick R. Fast r-cnn. In: Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago, Chile: IEEE, 2015. 1440 − 144.
[7] Guan Hao, Xue Xiang-Yang, An Zhi-Yong. Advances on application of deep learning for video object tracking. Acta Automatica Sinica, 2016, 42 (6): 834 − 847.
[8] Bertinetto L, Valmadre J, Henriques J F, et al. Fully convolutional siamese networks for object tracking. In: Proceedings of the 2016 European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016. 850 − 865.
[9] Danelljan M, Robinson A, Khan F S, et al. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In: Proceedings of the 2016 European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016. 472 − 488.
[10] Danelljan, M., et al., ECO: Efficient Convolution Operators for Tracking. arXiv preprint arXiv: 1611. 09224, 2016.
[11] Wang N, Yeung D Y. Learning a deep compact image representation for visual tracking [C]//Advances in Neural Information Processing Systems, 2013: 809 - 817.
[12] Wang L, Ouyang W, Wang X, et al. Visual Tracking with Fully Convolutional Networks [C]//IEEE International Conference on Computer Vision. IEEE Computer Society, 2015: 3119 - 3127.
[13] Nam H, Han B. Learning Multi-domain Convolutional Neural Networks for Visual Tracking [C]. Computer Vision and Pattern Recognition. IEEE, 2016: 4293 - 4302.
[14] Martin Danelljan, Gustav Häger, Fahad Khan, Michael Felsberg. Learning Spatially Regularized Correlation Filters for Visual Tracking [C]//ICCV, 2015: 4310 - 4318.
[15] David Held, Sebastian Thrun, Silvio Savaresi. Learning to Track at 100 FPS with Deep Regression Networks [C]//ECCV, 2016: 2152 - 2159.
[16] CHATFIELD K, SIMONYAN K, VEDALDI A, et al. Return of the devil in the details: delving
deep into convolutional nets [J]. Computer Science, 2014: 1 — 11.

[17] Danelljan, M., et al. Beyond correlation filters: Learning continuous convolution operators for visual tracking. in European Conference on Computer Vision. 2016: Springer.