Engineering Applications of Artificial Intelligence 157 (2025) 111415

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Learning color prompt and position constraint for visual tracking

Xuedong He [a,***], Huiying Xu [a,*], Xinzhong Zhu [a,b,c,**], Hongbo Li [c], Xiao Huang [d], Yunliang Jiang [a,e]

[a] School of Computer Science and Technology, Zhejiang Normal University, Jinhua, Zhejiang, 321004, China
[b] Research Institute of Hangzhou Artificial Intelligence, Zhejiang Normal University, Hangzhou, Zhejiang, 311231, China
[c] Beijing Geekplus Technology Co., Ltd, Beijing, 100101, China
[d] College of Education, Zhejiang Normal University, Jinhua, 321004, China
[e] School of Information Engineering, Huzhou University, Huzhou, 313000, China

ARTICLE INFO

Keywords:
Visual tracking
Discriminative model prediction
Prompt learning
Position offset constraint

ABSTRACT

The flourish of current visual tracking cannot be separated from powerful pre-trained backbone networks. Even the pre-trained networks frozen and used merely as a feature extractor can also obtain substantial tracking performance. However, how to acquire target-aware features suitable for visual tracking has always been a hot research topic to improve tracking robustness. Inspired by prompt learning, we propose the color prompt encoder to guide the acquisition of target-aware capability. Concretely, the color histogram features as a naive feature expression can provide complementary cues, so we employ color histogram features to construct the color target probability as a color prompt. Immediately after, the color prompt constructed is integrated into the unified tracking network to guide the generation of specific target feature maps. Furthermore, Discriminative Correlation Filters (DCF)-based trackers with an online update module can effectively adapt to constantly changing objects, so it is imperative to ensure that credible prediction samples are utilized to refine the tracking model online. Hence, we further devise an uncomplicated position offset constraint method based on target motion inertia to screen more reliable prediction results. Adequate experimental results reveal the validity of the color prompt encoder and position offset constraint in the DCF tracking framework. Our trackers can perform favorably against recent and far more sophisticated trackers on multiple public benchmarks. Concretely, our proposed tracker achieves a 0.815 robustness and 0.305 expected average overlap (EAO) on Visual Object Tracking (VOT) 2020 dataset, which is superior to the baseline in robustness (+2.6 %) and EAO (+0.8 %).

1. Introduction

Visual tracking as a sub-direction of computer vision is intended to predict the target states in a video sequence, notably when only an initial target state is provided. With deep learning shining brightly in this field, two prevailing Discriminative Correlation Filters (DCF)-based and Siamese-based trackers (Javed et al., 2023) have achieved significant development of tracking until now, yet this task is always troubled by some intricate challenge factors, *e.g.*, background clutter, scale variation, and obstructions.

At present, DCF-based (Christoph et al., 2022; Wang et al., 2021; Martin et al., 2017, 2019; Bhat et al., 2019; Henriques et al., 2015) and Siamese-based (Yan et al., 2021; Chen et al., 2021; Bo et al., 2018; Bertinetto et al., 2016a; Cui et al., 2024) frameworks receive widespread attention. Both core ideologies exploit trainable deep networks to learn a target appearance model online or offline. Siamese-based trackers include two sub-branches, with the inputs for each branch being the template and the search region. Specifically, the template is twice as small as the search region. SiamFC(Bertinetto et al., 2016a) as a primitive Siamese tracker constructs the tracking task as a template matching process, which is to match the most similar template from the search image region. Siamese trackers adopt a concise network structure to obtain commendable tracking effect and real-time speed, this paves the way for the application of deep learning in visual tracking. The entire Siamese tracking architecture can be subdivided into three parts: feature extraction, feature fusion, and prediction head. In feature extraction,

* Corresponding author.
** Corresponding author. School of Computer Science and Technology, Zhejiang Normal University, Jinhua, Zhejiang, 321004, China.
*** Corresponding author.
E-mail addresses: hexuedong@zjnu.edu.cn (X. He), xhy@zjnu.edu.cn (H. Xu), zxz@zjnu.edu.cn (X. Zhu).

https://doi.org/10.1016/j.engappai.2025.111415
Received 25 August 2024; Received in revised form 11 April 2025; Accepted 4 June 2025

apart from the AlexNet used by early SiamFC (Bertinetto et al., 2016a), the frequently-used Convolutional Neural Network (CNN) is ResNet (He et al., 2016), which has many versions, among which ResNet50 is the most commonly used option (Yan et al., 2021; Chen et al., 2021). With the development of Transformer (Ashish et al., 2017) in tracking, there are more and more works (Cui et al., 2024; Ye et al., 2022) using Vision Transformer (ViT) (Alexey et al., 2021) as a feature backbone to replace CNN in extracting feature expressions. Additionally, through adopting diverse feature fusion techniques (Chen et al., 2021; Bo et al., 2018; Bertinetto et al., 2016a; Bin et al., 2021; Jun et al., 2024) and anchor-based or anchor-free prediction heads (Bo et al., 2018; Cui et al., 2024; Bin et al., 2021; Chen et al., 2022a), Siamese-based trackers repeatedly refresh the new state-of-the-art records on multiple benchmarks (Huang et al., 2021; Fan et al., 2019; Matthias et al., 2018). Considering that Transformer can achieve long-distance interaction and simplify the feature extraction and fusion of Siamese trackers, the single branch architecture (Ye et al., 2022; Cui et al., 2023; Chen et al., 2022b) using ViT (Alexey et al., 2021) to jointly optimize feature extraction and fusion has been proven to be feasible. However, Siamese trackers exploit the limited template information to couple into the search image patch, and this template normally only adopts the initial template and is not updated (Chen et al., 2021; Bo et al., 2018; Bertinetto et al., 2016a). Although there are also some Siamese trackers (*e.g.*, STARK (Yan et al., 2021), MixFormer (Cui et al., 2024)) that introduce a dynamic template to update the template, they need an additional score prediction branch and two-stage training. Compared with DCF-based trackers, Siamese trackers without the update networks are less prone to cumulative errors in the inference stage, yet it is relatively susceptible to drastic appearance changes and similar objects.

Relatively speaking, DCF-based trackers (Christoph et al., 2022; Bhat et al., 2019; Henriques et al., 2015) can train a discriminative model prediction from far more background information, especially recent DiMP (Bhat et al., 2019; Martin et al., 2020) and ToMP (Christoph et al., 2022) can also adopt a trainable deep learning network architecture. In addition, DCFs are equipped with an online update module to regulate discriminative appearance model updates. We divide the DCFs into four components: feature extraction, target localization, target scale estimation, and online updating module. The related contents of each component will be elaborated in Sec. 2**.** Although DCF-based trackers have obtained impressive performances on prevailing benchmarks, we are aware that these tracking methods have two research points that need to be further perfected. **1)** In visual tracking, feature extraction has always been the top priority before tracking, a fabulous feature representation is positive for training appearance models. Looking at the progress of object tracking, it is not a new thing to achieve improved tracking performance by simply replacing the feature extraction backbone network. ResNet (He et al., 2016) as an excellent CNN backbone network can be confirmed as a preferred feature extractor among numerous CNNs. Furthermore, how to transfer the pre-trained feature attributes from image classification into more suitable feature attributes for visual tracking has always been a research topic worth exploring. Essentially, visual tracking focuses on target perception, with the aim of perceiving the target to be tracked from the image search area. Herein, we propose a novel color prompt encoder, the color prompt constructed from color histogram features is explicitly encoded into an attention map to guide the generation of target-aware features. The color prompt encoder we propose not only supplements the information of conventional color features, but also enhances the target-aware feature expressions of the original feature extraction backbone features through color attention fusion mode. **2)** Another thorny problem worth discussing in visual tracking is how to effectively update models online. Since the tracked target has irregular changes, it is reasonable for the tracking model to be updated appropriately to accommodate its dynamic variability. In this respect, linear interpolation (Martin et al., 2017; Henriques et al., 2015) or optimization with memory (Martin et al., 2019; Bhat et al., 2019) schemes are typically employed. No matter which method is used, it is unworkable to blindly update the model with the predicted target, which will cause tracking the wrong target. Therefore, DCF-like trackers (*e.g.*, SuperDiMP and ToMP) adopt a hard-negative mining (HNM) strategy to predict target tracking status from the acquired response map, which is utilized to control further model updates for achieving more advanced target localization. The HNM approach assesses the target tracking status via the target location and response map, but does not consider the motion inertia of the temporal target. Consequently, we present a position constraint mechanism to improve its existing shortcomings from the temporal position offset aspect, and this method has the advantages of low cost and portability.

To verify the practicability of our designed Color Prompt Encoder (CPE) and Position Constraint Mechanism (PCM), we integrate them into the recent DCF-based trackers (*i.e.*, SuperDiMP(Bhat et al., 2019; Martin et al., 2020) and ToMP (Christoph et al., 2022)) for ablation analysis and comparative experiments. Our trackers show comparative performances on six challenging benchmarks. In brief, our contributions have four aspects: **1)** Enlightened by the Staple (Bertinetto et al., 2016b) tracker, we adopt color histograms to construct color target probability features, and propose multiple combinations of up-sampling and patch embedding with attention and Multi-layer Perceptron (MLP) methods to achieve effective integration of shallow color features and deep CNN features. **2)** Inspired by prompt learning, we use Hanning window pre-processing the color target probability to obtain the color prompt, and the prompt is encoded into attention features to guide the acquisition of target-aware features. **3)** To mitigate the impact of inaccurate tracking results on the online model update process, we propose a position offset constraint method to force the tracked target to meet conventional motion inertia. **4)** We combine the proposed two solutions into SuperDiMP and ToMP trackers to implement comprehensive ablative studies and state-of-the-art experiments on multiple popular tracking benchmarks. Our approach obtains promising performances against other state-of-the-art trackers.

2. Related work

In the context of deep learning, DCF-based and Siamese-based trackers have achieved significant development. The proposed CPE and PCM modules are designed based on the DCF framework. The CPE focuses on improving feature extraction to obtain feature representations of specific target regions, while the PCM focuses on discussing how to better determine the feasibility of predicting targets and enhance the screening accuracy of online model update modules. Hence, the section mainly outlines DCF-based trackers and relevant research works in visual tracking.

Discriminative Model Prediction: DCFs(Christoph et al., 2022; Martin et al., 2017, 2019; Henriques et al., 2015) minimize an objective of least-squares regression to learn the discriminative target model. The target model is utilized to discriminate the foreground target from the background context, especially the target model needs to be updated in inference. In the early DCF framework, the target model refers to a correlation filter (Martin et al., 2017; Henriques et al., 2015), KCF (Henriques et al., 2015) as a classic DCF can timely learn a correlation filter via Fast Fourier Transform, thus DCF-based trackers have become synonymous with real-time trackers, especially real-time tracking on CPU devices. Now the target model is designed as a convolution kernel (Christoph et al., 2022; Martin et al., 2019; Bhat et al., 2019) in the deep DCF architecture, the convolution kernel is trained to bear condensed and generalized target representation. Based on the structure of representative discriminative model predictions such as ATOM and DiMP (Martin et al., 2019; Bhat et al., 2019), we can split these trackers into feature extraction, target localization, target scale estimation, and online model update. Firstly, the feature extraction methods has gone through hand-crafted (Henriques et al., 2015) to CNN (Martin et al., 2019) and then to ViT (Alexey et al., 2021). Considering single feature difficult to handle various complex scene changes, multi-feature fusion

(Bertinetto et al., 2016b; Bhat et al., 2018) naturally becomes a favorable supplementary means. Current DCF-based trackers mostly adopt ResNet (He et al., 2016) as the feature extraction backbone, and combine it with Transformer structure (Ashish et al., 2017) to achieve target localization and bounding box regression (Christoph et al., 2022). Secondly, the target features obtained from the feature extraction process are inputted into the discriminative model prediction to learn the target model, which is used to process the search image region of the next frame to determine the new target localization. The previous DCFs (Martin et al., 2017; Henriques et al., 2015) obtained the correlation filter through minimizing ridge regression formula in the Fourier domain, but the boundary effects (Danelljan et al., 2015) and the lack of end-to-end deep features led to a decrease in the competitiveness of DCF. Bhat et al. (2019) proposed a new-type DCF-like method, that is Discriminative Model Prediction (DiMP), DiMP learns the weights of the convolution kernel in a trainable way and further updates the kernel weights using an optimization method. The ability of the discriminative target model is substantially heightened by probabilistic regression (Martin et al., 2020). ToMP(Christoph et al., 2022) innovatively adopts Transformer to devise the discriminative model prediction instead of the optimization-based model prediction (Bhat et al., 2019). Finally, the target scale estimation has also made substantial progress from the conventional multi-scale search methods (He et al., 2023a) to the network learning branches (Christoph et al., 2022; Martin et al., 2019). Unlike the current Siamese-based methods using corner predict head (Yan et al., 2021; Bin et al., 2021) directly locate and estimate the location and scale state of the target, the trackers based on the discriminative model prediction predict the target scale after accomplishing the target localization.

Feature Extraction: The ability to extract features largely determines the upper limit of tracking performance. Therefore, our primary research motivation is how to enhance the effect of feature expression, especially the target-aware ability. We should know that exploited discriminative and generalizable features are particularly vital, a variety of visual features have been investigated in visual tracking. ECO(Martin et al., 2017) tested hand-crafted features, pre-trained deep features, and their combination for the impacts on tracking performance and speed. Specifically, ECO-HC using hand-crafted features can achieve a good trade-off between performance and speed on the CPU device. Moreover, multi-feature fusion (Bertinetto et al., 2016b; Bhat et al., 2018; Ma et al., 2015; Kang et al., 2023) can effectively compensate for inadequate expression of a single feature. Normally, distinct features play different roles in various tracking challenges, so it is practicable to improve tracking performance by integrating multiple features. However, based on the current research progress in feature extraction (Christoph et al., 2022; Yan et al., 2021; Chen et al., 2021), we can see that single-layer CNN features seem to be more popular, especially the use of end-to-end deep features (Martin et al., 2019; Bhat et al., 2019) further improves the inadaptability of simple pre-trained features. In addition, recent one-stream trackers (Ye et al., 2022; Cui et al., 2023; Gao et al., 2023) adopt ViT architecture to build joint feature extraction and feature fusion. HIPTrack (Cai et al., 2024) utilized historical locations and visual features to generate historical cues to enhance tracking performance. Considering that using a Vision Transformer (Alexey et al., 2021; Liu et al., 2021) as a feature extraction backbone will result in higher training and inference costs, we still use CNN as the feature backbone network. Although we have noticed that current multi-feature fusion often involves effectively fusing different feature layers of the same backbone network, classical ECO and UPDT (Martin et al., 2017; Bhat et al., 2018) also adopted the approach of fusing hand-crafted features with deep features. Therefore, we are wondering if hand-crafted features can be encoded into the deep network to achieve end-to-end learning. Hence, we draw lessons from some works (Bertinetto et al., 2016b; He and Huang, 2025) to implement a color prompt-guided encoder for complementary learning.

Failure Detection: The function of failure detection is convenient for updating the discriminative target model online. Early DCF-based trackers (Henriques et al., 2015; Bertinetto et al., 2016b) adopted a fixed moving average interpolation method to update the target model based on a new predicted sample. Given the redundancy of updating samples, ECO(Martin et al., 2017) updated the target model using an interval scheme. The convolution kernels of recent DCF-like trackers (Wang et al., 2021; Martin et al., 2019; Bhat et al., 2019; Christoph et al., 2021) are online optimized with sample memory. Correct model updating is conducive to maintaining and enhancing the discriminative ability of target models, but it may be destroyed by incorrect targets when the target is in complicated tracking scenes. Therefore, it is necessary to analyze the tracking results and make appropriate update control. Under normal circumstances, the output of the DCF target localization module is a response map approaching a Gaussian distribution, so some works (Martin et al., 2019; Wang and Liu, 2017; Bolme et al., 2010) developed feasible solutions from the response map. MOSSE (Bolme et al., 2010) calculated the mean and variance from the response map to obtain the peak to sidelobe ratio (PSR), the PSR can further perform failure detection. Due to the inaccuracy of PSR, APCE(Wang and Liu, 2017) improved PSR score criterion to realize a high-confidence update. PSR and APCE make decisions based on a comparison of the response map and the set threshold, but do not consider multiple tracking scenarios. ATOM (Martin et al., 2019) considered the hard negative sample to devise a hard negative mining (HNM) strategy, which thinks about the diverse target status, that is *not found*, *normal*, *hard negative*, and *uncertain*. The strategy is also applied to the DiMP variants (Wang et al., 2021, 2024; Bhat et al., 2019; Martin et al., 2020; Christoph et al., 2021). Moreover, He et al. (2022) proposed a voting decision method as an auxiliary failure detection mechanism, and this strategy does not require extra network training. Since PSR and APCE mentioned above did not think over the secondary peak, He et al. (2023b) presented an all-new score index called primary and secondary peak mean energy, these score indexes are combined with HNM for use. Apart from designing strategies based on response maps, some researchers (Yan et al., 2021; Cui et al., 2024; Chen et al., 2022a; Christoph et al., 2021; Bhat et al., 2020; Dai et al., 2020; He and Chen, 2022) also offline trained an extra network branch to detect tracking failure. Unlike many methods that use additional networks, our proposed position constraint mechanism is a low-cost and portable approach. Moreover, a learnable query token (Xie et al., 2024) is used for achieving target context-aware learning. This approach adopts dynamic token fusion instead of directly replacing templates to achieve updated learning of visual tracking.

3. Proposed method

Encoding an RGB color image into deep feature maps through CNN networks is a common practice. Under the traditional DCF architecture, HOG features (Henriques et al., 2015) once become the preferred manual feature and achieve good tracking performance. Moreover, the combination of color features and HOG features (Martin et al., 2017; Bertinetto et al., 2016b; Bhat et al., 2018) can achieve a better performance improvement. With the unified tracking paradigm of deep learning, extracting or fusing features by learnable methods has become a popular research point. However, fusing multiple domain features instead of multi-layer features from a single backbone network has not been effectively explored. The primary task to be addressed in this article is to encode plain color information into deep networks. We first adopt color histogram statistics to obtain color target probability as a color prompt feature, which is combined into the deep network to guide the learning of more specific target features. Moreover, we adhere to common knowledge of object motion and present a position constraint mechanism to rectify the abnormal tracking results.

3.1. Overview

Our proposed two modules are based on the Discriminative Model Prediction (Christoph et al., 2022; Bhat et al., 2019; Martin et al., 2020), which is illustrated in Fig. 1. Our improved trackers adopt the state-of-the-art SuperDiMP and ToMP as the baseline trackers. Taking SuperDiMP as an example, SuperDiMP includes the target localization module of DiMP(Bhat et al., 2019) and the probabilistic bounding-box regression of PrDiMP (Martin et al., 2020). Especially, SuperDiMP can be decomposed into four stages: feature extraction, target localization and target scale estimation, and online model update. Firstly, the image patches from the training and test frames are preprocessed and input into the backbone network to extract deep feature maps. The backbone network adopts a pre-trained ResNet50 to obtain the *conv*4_*x* feature maps as image features. Moreover, the network weights before *conv*4_*x* layer will be frozen. In order to make the features obtained by the image classification network more suitable for visual tracking, SuperDiMP adds a convolutional block to obtain the learned feature expression. In this part, we propose a Color Prompt Encoder (CPE), which is aimed at encoding color information as a prompt to produce target-aware feature maps. We believe that CPE can provide the function of prompt guidance and target attention. The feature extraction stage composed of Backbone and CPE modules is formulated as

$$x_f = \mathscr{C}_p\left(R_{c_4}(x_{im})\right). \tag{1}$$

Here, x_{im} denotes an image patch (*e.g.*, Test frame in Fig. 1). R denotes backbone

network (*i.e.*, ResNet50) and the subscript indicates which layer of feature map the network outputs (*e.g.*, c_4 denotes *conv*4_*x*). $\mathscr{C}_p$ denotes our proposed CPE module. The feature extraction component with our presented CPE is able to obtain more diverse

and specific target features x_f, which are passed through the model predictor module

$\mathscr{M}$ to acquire the target model α (*i.e.*, convolution kernel). The target model α from model predictor $\mathscr{M}$ convolve the test frame features x_f^t to output the expected

Gaussian label scores $y_g \in \mathscr{Y}$.

In the actual training phase, since visual tracking only provides initial sample information, data augmentation is used to obtain the training samples of image features and response scores. The training samples $S_{train} = \left\{\left(x_f^i, y_g^i\right)\right\}_{i=1}^{n}, i \in [1, n]$ is input into the model predictor $\mathscr{M}$ to acquire an target model $\alpha = \mathscr{M}(S_{train})$, n denotes the total number of training samples. The objective function for model predictor $\mathscr{M}$ is formulated as

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \sum_{\left(x_f, y_g\right) \in S_{train}} r\left(\alpha * x_f, y_g\right) + \beta f(\alpha). \tag{2}$$

Here, the objective function includes the residual function $r(\cdot)$ and regularization term $f(\cdot)$ weighted by a scalar β. The $r(\cdot)$ computes an error between the response map $\mathscr{R} = \alpha * x_f$ and the Gaussian label y_g, $*$ denotes the convolution operation.

In the tracking inference phase, the target model α convolutes the test frame feature map x_f^t to obtain a response map $\mathscr{R}^t = \alpha * x_f^t$, the location (x_l, y_l) of the $\mathscr{R}^t$ peak value is determined as the next frame target position. After requiring the coordinate (x_l, y_l) from the target localization, the bounding box regressor based on IoUnet (Martin et al., 2019) performs the target scale estimation. Different from target localization, the target scale estimation uses the *conv*3_*x* and *conv*4_*x* features to estimate the final target state, that is $(x_s, y_s, w_s, h_s) = \mathscr{S}\left(R_{c_3,c_4}\right)$, $\mathscr{S}$ denotes the target scale estimation module. Owing to the DCF-based tracker allowing storing target samples to online update the target model α, it is especially crucial to screen the target and pick out better samples for storage. Therefore, SuperDiMP adopts the HNM method (Martin et al., 2019) to analyze the target flags predicted by the target localization module, these flags are used to optimize the model predictor by adjusting the sample set and update the model learning rate online. Though the HNM method has achieved appreciable improvement effects, which is often not enough in complex scenes. As a supplement, we devise a Position Constraint Mechanism (PCM) to constrain the position offset of the tracked target from conventional motion inertia.

3.2. Color prompt acquisition

ResNet is used as a general feature extraction backbone network, and the network weights are partially frozen during training. Therefore, SuperDiMP and ToMP adopt a learnable convolution block to get target classification features. However, the representational ability of single-layer feature map is limited, even if the used *conv*4_*x* feature layer of ResNet is not frozen. Considering the combination of color histogram information can achieve a gain effect in the traditional DCF framework, but

there are few reports under the deep network architecture. As is shown in Fig. 2, we adopt color histogram features to construct color target probability as a color prompt. We first refer to the mentality of Staple (Bertinetto et al., 2016b) to generate Color Target Probability (CTP). CTP is obtained by foreground and background probability maps. First, we demarcate the foreground mask $m_F = F(x_{im}, p_s)$ and background mask $m_B = B(x_{im}, p_s)$ from the image patch based on the target position and scale p_s (*i.e.*, the red bounding box in Fig. 2). Here, F denotes that only the RGB values of bounding box inside is counted as foreground color histogram, while the function of B is just the opposite. Next, we extract color histograms for foreground and background mask regions respectively, which are formulated as

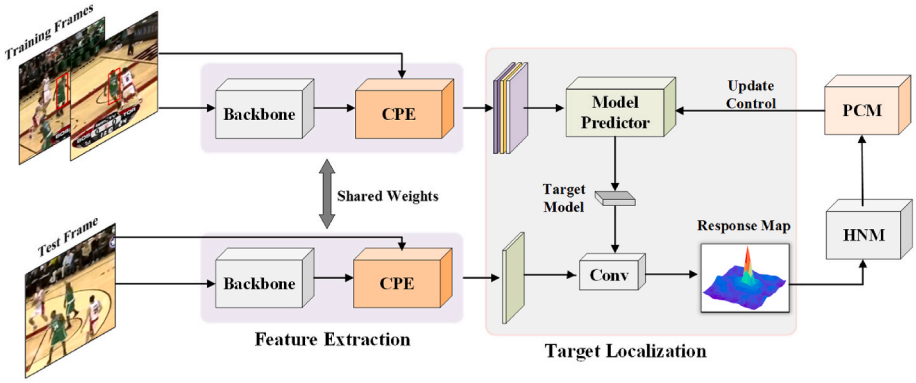

Fig. 1. Overview of our proposed Color Prompt Encoder (CPE) and Position Constraint Mechanism (PCM) based on Discriminative Model Prediction. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

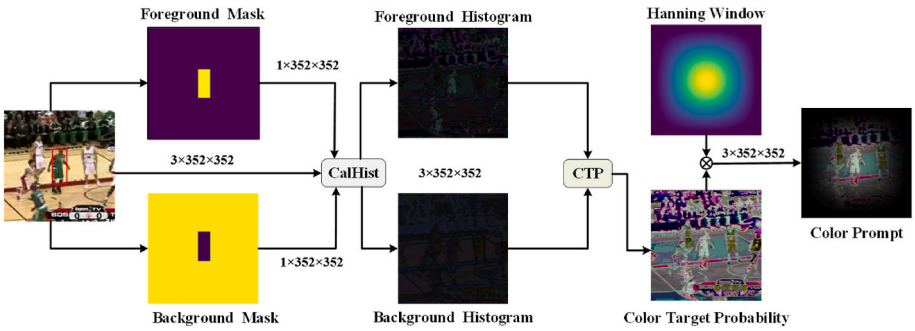

Fig. 2. Acquisition of Color Prompt. Color Prompt is obtained by the element-wise multiplication (⊗) of Color Target Probability (CTP) and Hanning window, while CTP is based on foreground and background histogram maps. Taking the input image dimension of 3 × 352 × 352 as an example, we can be seen that the color prompt feature map processed by the color prompt module has the same dimension as the original input image. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

$$H_{hist}(x_{im}, m) = \frac{CalHis\left(\frac{x_{im}}{\beta} \circ m\right)}{sum(m)}, m \in [m_F, m_B]. \tag{3}$$

Here, the bin size β of histogram is 16 by default, $x_{im} \in [0,255]$. $\circ$ denotes the

reserved RGB image region based on the foreground or background mask. $CalHis(\cdot)$

counts the same value of each pixel in the image region, $sum(\cdot)$ denotes the summation function. Finally, we obtain the color target probability map based on the foreground histogram map H^F_{hist} and background histogram map H^B_{hist}, which is formulated as

$$\mathscr{P} = \frac{H^F_{hist}}{H^F_{hist} + H^B_{hist} + \lambda}. \tag{4}$$

Here, $\mathscr{P}$ denotes the color target probability, and the hyperparameter λ is set as 0.01, which can prevent the denominator of $\mathscr{P}$ from being zero.

To eliminate the interference of background probability information, we use the Hanning window to smooth $\mathscr{P}$ for attaining the ultimate color prompt map $\mathscr{P}_c$. After obtaining the color prompt map $\mathscr{P}_c$, we next will discuss in detail how to encode it into the deep network.

3.3. Color prompt encoder

In order to integrate the color prompt into deep networks properly, we propose the Color Prompt Encoder (CPE) to encode the color prompt into ResNet to guide the generation of target-aware feature maps. Considering the inconsistency between the dimensions of the color prompt and backbone features, we propose two methods, bilinear interpolation (BI) and patch embedding (PE), to maintain consistency between the two feature dimensions. In addition, we also propose a spatial-temporal attention PE module to achieve the fusion of color prompt (CP) features and deep features, the final encoding methods include BI_MLP, PE_MLP and PE_ATT.

As is show in Fig. 3, since the shape of our $\mathscr{P}_c$ is the same as that of the original RGB image, the width and height sizes of the deep feature are down-sampled by stride 16, we initially perform a bilinear interpolation method to resize the shape of $\mathscr{P}_c$ as that of deep features, and then we use a multilayer perceptron (MLP) layer and a Sigmoid function to map $\mathscr{P}_c$, which is formalized as

$$f_{\mathscr{P}_c} = \delta(MLP(\gamma(\mathscr{P}_c))). \tag{5}$$

Here, γ denotes the resize operation. The channels of MLP layer are (Wang et al., 2021; Alexey et al., 2021; Alexey et al., 2021, 2021; Javed et al., 2023), 3 is the input channel, 16 is the hidden channel and 1 is the output channels. As is shown in Fig. 3, the MLP layer is comprised of two CBRs (*i.e.*, Conv3 × 3+BN + ReLU) and one 1 × 1 convolution (*i.e.*, Conv1 × 1). The mapped features $f_{\mathscr{P}_c}$ is conducted element multiplication with the deep features, which is formulated as

$$f_{prompt} = c^3_{br}\left(R_{c_4}(x_{im})\right) \otimes f_{\mathscr{P}_c}. \tag{6}$$

Here, c^3_{br} denotes CBR. To make the feature prompt smoother, we use a Conv1 × 1 to process and an InstanceNorm to normalize f_{prompt} features for obtaining the final target-aware classification features, which is formulated as

$$x_f = N\left(c^1\left(f_{prompt}\right)\right). \tag{7}$$

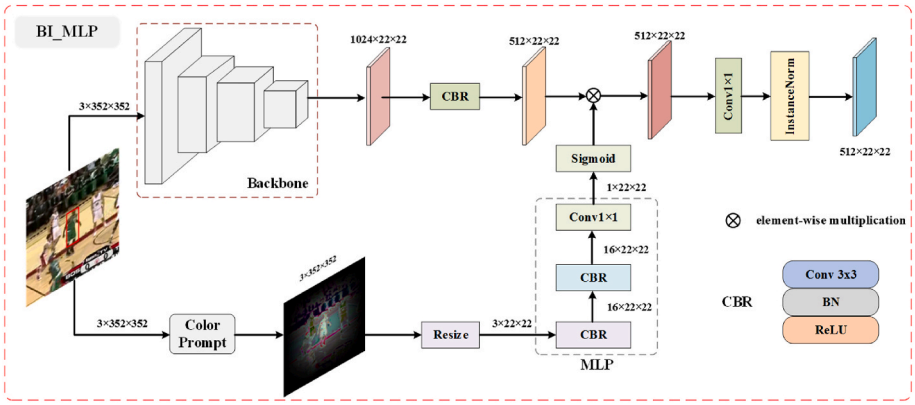

Fig. 3. Structure of BI_MLP. The proposed BI-MLP module uses bilinear interpolation to resize the CP map, and adopts MLP and Sigmoid to obtain the CP attention map, which is attached to enhance the target perception effect.

Here, c^1 denotes Conv1 $\times$ 1, N denotes InstanceNorm layer.

The CPE module with BI_MLP mentioned above has expounded the basic components of CPE. Inspired by ViT, we adopt learnable patch embedding to directly down-sample the spatial size of the color prompt and increase its channel size. Similarly, MLP and Sigmoid are used to map color features to obtain color attention maps, which is formalized as

$$f_{\mathscr{P}_c} = \delta(MLP(P_E(\mathscr{P}_c))). \tag{8}$$

Here, P_E denotes the patch embedding, which is CBR with Conv16 $\times$ 16 with stride 16. The structure of the MLP layer is consistent with that of BI_MLP module. Since the CP channel mapped by PE is 256, the channels of MLP layer are (Javed et al., 2023).

As shown in Fig. 4, the proposed PE_MLP method adopts the same modules of BI_MLP to map and encode the color attention maps $f_{\mathscr{P}_c}$ and deep features.

The two tactics of encoding color prompts into deep features mentioned above have been introduced. We propose to replace MLP with a spatial and channel attention module to guide and prompt deep features from multiple dimensions. The detailed network structure is shown in Fig. 5, the channel attention module adopts adaptive average pooling to compress spatial dimensions, followed by the two-layer convolution kernels and a Sigmoid function mapping to obtain channel attention maps, and the basic structure of the spatial attention module is similar, and the variation of the feature dimension size is indicated in Fig. 5. Finally, the concatenation operation by channel is used to fuse the spatial and channel feature maps, which is formulated as

$$f_{prompt} = cat(f_b \otimes SA(P_E(\mathscr{P}_c)), f_b \otimes CA(P_E(\mathscr{P}_c))). \tag{9}$$

Here, $f_b = c^3_{br}(R_{c_4}(x_{im}))$, SA denotes the spatial attention module, CA denotes the channel attention module, the *cat* denotes concatenation operation.

3.4. Position constraint mechanism

We make a detailed description for the proposed Color Prompt Encoder, which is beneficial to creating more diverse and specific target classification features. Furthermore, most DCF-based trackers are to seek a feasible target position by yielding a robust response map. However, the target model of these trackers will be updated online according to the initial annotated frame and the predicted results, so the screening of the predicted results is vitally crucial. Despite SuperDiMP and ToMP using the HNM method to reach an appreciable performance, the results obtained from the response map are not necessarily credible in complex scenes. In addition, the HNM method will output the final position offset and target status from the response map, yet it has the defect of insufficient utilization of position offset.

Accordingly, we propose the Position Constraint Mechanism (PCM) to identify tracked anomalies based on the position offset of the previous target position and the current predicted position, which is depicted in Fig. 6. The intention of PCM provides supplementary cues for the HNM method, which regulates updates of the sample memory and model learning rate (0.01 by default). The sample memory is used to store initial samples and predicted samples, and its capacity is set to a maximum of 50. In addition, the memory is added or replaced in the case of *normal* and *hard negative*, and the *hard negative* status will double the model learning rate (*i.e.*, 0.02). The *not found* status determines the target as lost if the peak value of $\mathscr{R}$ is lower than 0.25. For *uncertain* status, the target state will also be output normally, but no update will be operated. Due to the complexity of tracking conditions, HNM cannot cover all aspects. Therefore, we propose a position offset constraint method that follows motion inertia to ensure the higher credibility of the updated samples as much as possible.

We can be seen from Fig. 6 that PCM mainly determines the target motion range based on the target position $(x_l^{t-1}, y_l^{t-1}) = \text{argmax}(\mathscr{R}^{t-1})$ in the previous frame and the preset constraint radius $\boldsymbol{\pi}$ (*i.e.*, red circle). If the target position (x_l^t, y_l^t) in the next frame is outside the circle, it is considered that the target motion is abnormal, which is formulated as

$$\boldsymbol{o}_p = \left\| (x_l^{t-1}, y_l^{t-1}) - (x_l^t, y_l^t) \right\|_2. \tag{10}$$

Here, $\boldsymbol{o}_p$ denotes the position offset. In order to better measure the effectiveness of position offset $\boldsymbol{o}_p$, we set the constraint radius $\boldsymbol{\pi}$ based on the size of the initial target scale (w, h), which is formulated as

$$\boldsymbol{\pi} = d\frac{\sqrt{wh}}{2}. \tag{11}$$

Here, the parameter d is the displacement scale, which defaults to 0.8. In this PCM, the $\boldsymbol{o}_p > \boldsymbol{\pi}$ indicates that the predicted results are not recommended to update the sample set S_{train}.

Moreover, the above-mentioned practice is for a single-frame case, and we have also considered the condition of multi-frame position offset. Firstly, we initialize the previous position storage space and update it using a first-in-first-out method. Subsequently, we calculate the position offsets between the current frame position.

(x_l^t, y_l^t) and multiple stored positions $\left[(x_l^{t-n}, y_l^{t-n}) \cdots (x_l^{t-1}, y_l^{t-1})\right]$, that is $\left[\boldsymbol{o}_p^{t-n} \cdots \boldsymbol{o}_p^{t-1}\right]$, and finally take the average value for comparison with the constraint radius $\boldsymbol{\pi}$.

3.5. Training details

Our CPE and PCM modules can be readily combined with recent DCF-based frameworks, such as SuperDiMP and ToMP. We substitute for the original ClsFeat with our CPE module to perform end-to-end training. The obtained target classification features by our method are fed into the model predictor $\mathscr{M}$ for getting a stronger target model α. Our training settings are consistent with SuperDiMP and ToMP trackers

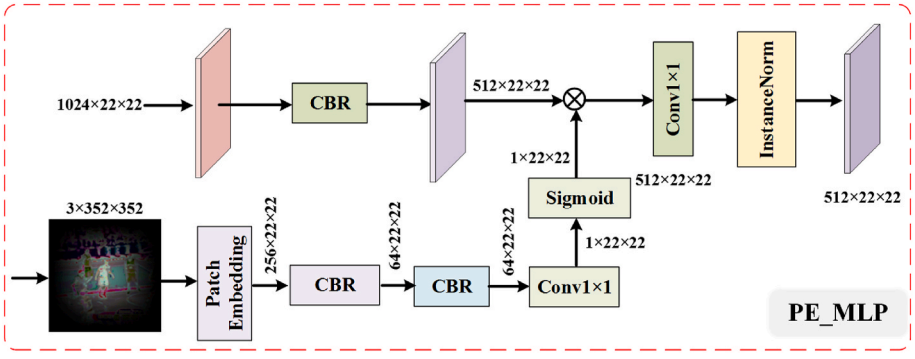

Fig. 4. Structure of PE_MLP. The proposed PE_MLP module uses patch embedding to down-sample and encode the CP map, and it also adopts MLP and Sigmoid to obtain the CP attention map.

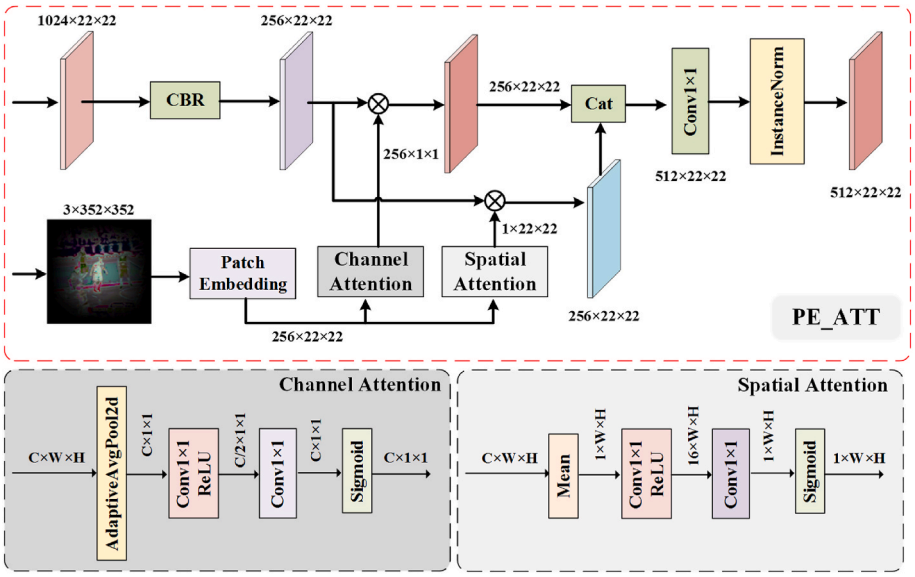

Fig. 5. Structure of PE_ATT. The proposed PE_ATT module uses patch embedding to down-sample and encode the CP map, and it adopts spatial and channel attention to replace the origin MLP layer.

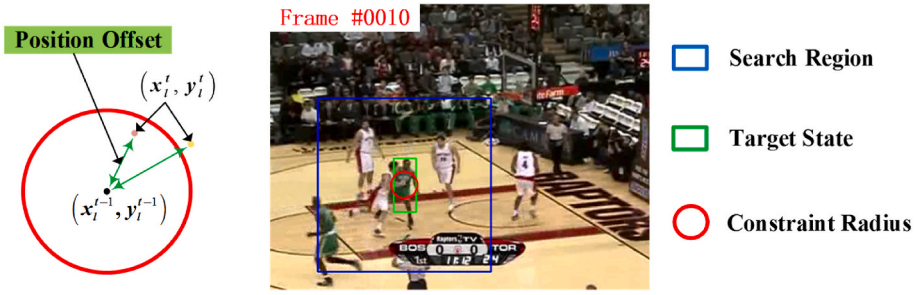

Fig. 6. Visualization of the Position Constraint Mechanism (PCM). We provide an intuitive explanation for the search region, target state, and constraint radius in image frame #0010 from the Basketball sequence of OTB100. The red circle is drawn based on the previous target position as the center and the devised constraint radius. The concept of our proposed PCM is that the target localization in the next frame is outside the circle to indicate tracking anomalies, without considering it as a model update sample. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

except for the CPE module, yet our training devices are two NVIDIA GeForce RTX 3090 GPUs. Moreover, the PCM method adopts the predicted position from the tracking model and has no training demand, which can be regarded as a screening method to enhance the stability of the model update.

4. Experiments

4.1. Experimental details

To verify the practicability of our designed CPE and PCM modules, we integrate them into the recent DCF-based trackers (Christoph et al., 2022; Bhat et al., 2019; Martin et al., 2020) (*i.e.*, SuperDiMP and ToMP).

We should note that SuperDiMP uses ResNet50 as the backbone network, and the target localization and scale estimation network is built in the form of convolutional networks with a training epoch of 50. ToMP uses two versions, ResNet50 and ResNet101, but actual testing has found that ResNet101 is not better than ResNet50. The target localization uses Transformer to build the encoder and decoder architecture, and then uses *ltrb* representation (Tian et al., 2019) to implement bounding box regression. Due to the complexity and slow convergence of the Transformer structure, the ToMP training epoch is 300. Given that SuperDiMP and ToMP have similar structures, our proposed CPE and PCM are also applicable to both. To sum up, we mainly conduct ablation experiments based on SuperDiMP, and perform the State of the Art (SOTA) experiments about SuperDiMP and ToMP.

Our improved trackers show comparative performances on six challenging benchmarks including OTB100(Yi et al., 2015), NFS (Galoogahi Hamed et al., 2017), UAV123 (Mueller et al., 2016), LaSOT (Fan et al., 2019), and TrackingNet (Matthias et al., 2018) and VOT2020 (Matej et al., 2020). Our approaches are implemented in Python using PyTorch, and all experiments are running in the GPU processor of GeForce RTX 4070. The code, raw tracking results, and trained models are available at https://github.com/hexdjx/VisTrack.

4.2. Ablation study

To demonstrate the effectiveness of our proposed CPE module and PCM method, we perform synthetical ablation studies on OTB100, NFS, and LaSOT.

Color Prompt Encoder: Due to the three prompt fusion schemes proposed by our CPE module, we first use BI_MLP to realize the CPE module for ablation analysis. Firstly, we replace the ClsFeat block of the SuperDiMP and ToMP50 with our CPE module to verify its validity. Since the color prompt features used during the training process are based on the given target states of the training and testing samples, but in the online tracking process, except for the first frame where the target state is provided, the remaining target states are predicted by the

tracking model. Therefore, it is extremely important to obtain color prompt features closer to the actual value to guide the generation of target-aware features. We discuss four approaches to solve the problem of color prompt feature generation in the tracking inference process. To explore how the Color Prompt (CP) should be obtained, we perform comparative experiments for four CP update methods on OTB100, NFS and LaSOT. ① Initial CP: using only the first frame CP to guide the obtainment of prompt features, ② Update CP per frame: generating CP directly from the previous frame, ③ Moving average update: when the *normal* and *hard negative* statues are given, we update CP with moving average method, that is $\mathscr{P}_c^u = (1-\eta)\mathscr{P}_c^u + \eta\mathscr{P}_c$, $\eta = 0.01$ is learning rate, $\mathscr{P}_c^u$ denotes the updated CP, $\mathscr{P}_c$ denotes the current CP, ④ Moving average update with replace: It's a bit similar to the combination of the **② and ③**, under normal tracking conditions, each frame is updated, and in other cases, the updated CP is used as a replacement. We can see from Table 1 that proper update has positive impacts to tracking results. Compared with SuperDiMP, the updated CP (*i.e.*, ③) acquires AUC gains of 0.7, 1.4 (%) on NFS and LaSOT datasets. Moreover, we can see that using the updated CP based on ToMP50 can also achieve good results, and especially will bring about obvious improvement with Precision and AUC gains of 1.1, and 1.2 (%) on NFS dataset.

In addition, we also conduct experiments on the effectiveness of the Hanning Window used in Fig. 2 for smoothing CTP to obtain color prompt features, which aims at demonstrating the impact of with and without the Hanning Window on tracking results. As shown in Table 2, we choose two CP generation methods (*i.e.*, ② and ③) as comparison baselines, and the results show that without the Hanning Window can also achieve not bad results, especially CPE for ③ choice can also obtain relative Precision and AUC increases of 1.0, and 0.9 (%) on NFS dataset.

Based on the results in Tables 1–2, we use the AUC score of LaSOT as the measure for the optimal selection of method combinations. Ultimately, we determine the moving average update CP (*i.e.*, ③) with the Hanning Window method as the optimal choice for further ablation test.

The above ablation experiments are conducted based on BI_MLP in Fig. 3. In order to further test the PE_MLP and PE_ATT methods, we directly compare the performance of diverse trackers based on the optimal combination obtained from BI-MLP. As shown in Fig. 7, we also use the *add* and *cat* operations commonly used in the network to explore the case of directly fusing the embedding vectors obtained from PE with deep features. Compared to the baseline SuperDiMP, most CPE_* trackers have shown improvement, except for a 0.6 % decrease in performance of CPE_pe_att on NFS dataset. From the results of the naive methods of *add* and *cat* on the OTB100 and NFS datasets, the improvement effect is more obvious, but the performance on LaSOT is not as good as the method introduced in Sec. 3.3. Based on LaSOT analysis, we believe that CPE_bi_mlp is an effective method for fusing color prompt features with deep features.

Position Constraint Mechanism: According to the above ablation experiments in the CPE module, we adopt the bilinear interpolation to down-sample the CP feature and use the moving average update CP method with Hanning Window to validate the feasibility of PCM. As shown in Table 3, PCM showed a significant Precision and AUC improvement compared to SuperDiMP on the OTB100 and NFS datasets, and a certain improvement on LaSOT, The Precision has been improved by 1.4, 2.2, and 0.4 (%), and the AUC has been improved by 0.7, 1.1, and 0.4 (%), respectively. However, we also found that the combination of PCM and CPE is not very effective, but it has an improvement effect on LaSOT. We speculate that this is because the tracking state results from PCM may interfere the CP inference updates of CPE.

In addition, we also conduct ablation experiments for the constraint radius $\boldsymbol{\pi}$ in Eq. (11) and test which strategy is more feasible. These strategies include the fixed threshold (*i.e.*, $\boldsymbol{\pi} = 20$), which is derived from the precision plot adopting a reported threshold of 20 pixels, the threshold counted in the initial frame, and the dynamic threshold based on the target scale of the previous frame. As shown in Table 4, we test the comparison of three threshold strategies based on the CPE method. The results show that the fixed and dynamic thresholds are not as good as the initial threshold. Especially for the performance on the LaSOT dataset, the fixed threshold results in an AUC decrease of 0.9 %, while the dynamic threshold results in an AUC decrease of 2.1 %.

The PCM described above is based on a single-frame position constraint, only considering the position offset between the previous frame and the current frame. We also discuss the impact of multi-frame positions, such as 3 and 5 frames, on tracking performance, as shown in Table 5. We use three different frames to construct PCMs, which are embedded into SuperDiMP and ToMP50 trackers with integrated CPE module. The experimental results show that using 1-frame and 3-frame PCMs are relatively better. In addition, we also test whether using average or maximum position offset for 3-frame PCM is more appropriate compared to the constraint radius $\boldsymbol{\pi}$, as shown in Table 6. We can see that the performance on all three datasets indicates that using average position offset results is relatively optimal.

Overall, the tracking performance of PCM can achieve certain increases by using 1-frame or 3-frame position offset averaged and initial frame threshold on the OTB100 and LaSOT datasets, but that of NFS is poor.

In order to present the improvements of CPE and PCM on SuperDiMP and ToMP50 more concisely, we visualize the Precision plot and Success plot of LaSOT, where the legend lists the Precision and AUC scores of the trackers. As shown in Fig. 8, it can be seen that the proposed CPE has a significant improvement on SuperDiMP, achieving a relative gain of 2.1 % and 1.4 % in terms of Precision and AUC. The further improvement of CPE by PCM is relatively small, achieving a relative gain of 0.2 % Precision and 0.1 % AUC based on ToMP50+CPE. In a word, the proposed CPE module and PCM methods are feasible and have the effect of improving tracking performances.

Table 1
Analysis of the generation methods of Color Prompt (CP) in the inference stage and their impacts on the SuperDiMP and ToMP50 trackers in terms of precision (Prec.) and area under the curve (AUC) scores. The two best results are shown in bold **red** and **blue** fonts.

	OTB100		NFS		LaSOT	
	Prec.	AUC	Prec.	AUC	Prec.	AUC
SuperDiMP	90.8	70.1	78.1	65.2	65.3	63.1
①	**92.7**	**71.1**	78.3	65.1	66.3	63.7
②	91.8	70.6	**79.1**	**65.6**	66.2	63.7
③	91.2	70.1	**79.7**	**65.9**	**67.4**	**64.5**
④	**92.4**	**70.8**	77.0	63.9	**67.0**	**64.3**
ToMP50	90.8	70.1	80.5	66.9	**72.2**	**67.6**
①	90.7	70.2	**81.6**	**67.5**	70.3	66.0
②	**91.6**	**70.7**	80.0	66.4	**72.1**	67.3
③	90.5	69.9	**81.6**	**68.1**	**72.2**	**67.4**
④	**91.3**	**70.4**	79.9	66.3	71.7	66.8

Table 2
Comparative analysis about with (w/) and without (w/o) Hanning Window in Fig. 2 based on two CP generation methods using ② Update CP per frame and ③ Moving average update, and its impact on the proposed CPE embedded into SuperDiMP tracker in terms of Prec. and AUC scores.

	Hanning Window	OTB100		NFS		LaSOT	
		Prec.	AUC	Prec.	AUC	Prec.	AUC
②	w/	91.8	70.6	79.1	**65.6**	66.2	63.7
	w/o	**91.9**	**70.7**	**79.2**	65.4	**66.7**	**64.0**
③	w/	**91.2**	**70.1**	79.7	65.9	**67.4**	**64.5**
	w/o	90.5	69.7	**80.7**	**66.8**	66.1	63.5

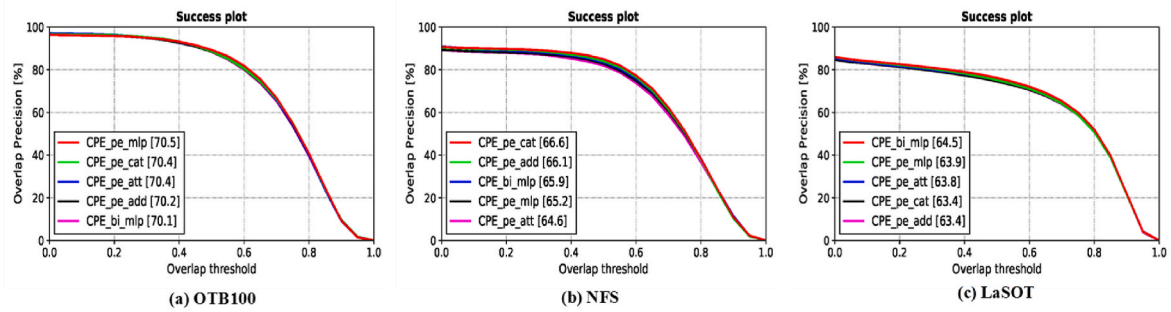

Fig. 7. Illustration of Success plot on OTB100, NFS, and LaSOT. The legend displays the AUC score for diverse improved CPE_* trackers.

Table 3
Analysis of the CPE and PCM modules embedded into the SuperDiMP tracker and their impacts on the tracking performance in terms of Prec. and AUC scores.

	OTB100		NFS		LaSOT	
	Prec.	AUC	Prec.	AUC	Prec.	AUC
SuperDiMP	90.8	70.1	78.1	65.2	65.3	63.1
CPE	91.2	70.1	**79.7**	**65.9**	**67.4**	**64.5**
PCM	**92.2**	**70.8**	**80.3**	**66.3**	**65.7**	63.5
CPE+PCM	**91.6**	**70.4**	78.2	64.9	**67.4**	**64.7**

Table 4
Analysis of PCM with three thresholds for the constraint radius π in Eq. (9) based on the CPE method and their impacts on the tracking performance in terms of Prec. and AUC scores.

		OTB100		NFS		LaSOT	
		Prec.	AUC	Prec.	AUC	Prec.	AUC
CPE		**91.2**	70.1	**79.7**	**65.9**	**67.4**	**64.5**
PCM	fixed	91.0	69.8	**79.2**	**65.6**	**66.3**	63.6
	initial	**91.6**	**70.4**	78.2	64.9	**67.4**	**64.7**
	dynamic	**91.6**	**70.5**	78.5	65.3	64.6	62.4

4.3. Comparison to the state of the art

In Sec. 4.2, we conduct a series of ablation experiments on three datasets to verify the availability of the CPE and PCM modules. The experimental results show that our proposed approaches have favorable improvement effects, we uniformly embed the two components into SuperDiMP and ToMP50 for State of the Art (SOTA) experiment, which is called ProDiMP and ProToMP trackers successively. We compare our ProDiMP and ProToMP trackers with the SOTA trackers on six challenging tracking benchmarks.

LaSOT (Fan et al., 2019)**:** Table 7 shows the comparison results in terms of Precision and AUC scores for various trackers. LaSOT dataset contains most of the longer video sequences, averaging 2500 frames for each video sequence, which makes this dataset more challenging than other datasets. Therefore, our proposed color prompt encoder and position constraint mechanism is to enhance and maintain the discriminative ability of the target classifier. This is crucial for long video sequences, because the longer the video sequence, the more likely the model will be polluted by the erroneous background information. We choose the recent DiMP(Bhat et al., 2019), PrDiMP (Martin et al., 2020), SuperDiMP, TrDiMP (Wang et al., 2021), TransT (Chen et al., 2021), STARK-ST50 (Yan et al., 2021), Sim-B/32 (Chen et al., 2022b), CSWinTT (Song et al., 2022) and ToMP50 (Christoph et al., 2022) are considered for comparison. Our proposed CPE and PCM approaches integrated into

Table 5
Analysis of PCM using the multi-frame position constraint based on the CPE method and their impacts on the tracking performance in terms of Prec. and AUC scores.

		OTB100		NFS		LaSOT	
		Prec.	AUC	Prec.	AUC	Prec.	AUC
SuperDiMP + CPE		91.2	70.1	**79.7**	**65.9**	**67.4**	**64.5**
PCM	1-frame	**91.6**	**70.4**	**78.2**	**64.9**	**67.4**	**64.7**
	3-frame	**92.3**	**70.9**	78.0	64.5	**67.1**	64.4
	5-frame	**91.6**	**70.4**	77.5	64.1	66.6	63.9
ToMP50 + CPE		90.5	69.9	**81.6**	**68.1**	**72.2**	**67.4**
PCM	1-frame	91.1	**70.2**	**81.1**	**67.5**	**72.4**	**67.5**
	3-frame	**92.6**	**71.1**	80.9	67.3	**72.2**	**67.4**
	5-frame	**91.3**	70.1	79.9	66.5	71.4	66.6

Table 6
Analysis of the PCM with 3-frame position constraint, considering whether to use the average or max position offset result as comparison with threshold π, and their impacts on the tracking performance in terms of Prec. and AUC scores.

	Average	OTB100		NFS		LaSOT	
		Prec.	AUC	Prec.	AUC	Prec.	AUC
3-frame	w/	**92.3**	**70.9**	**78.0**	**64.5**	**67.1**	**64.4**
	w/o	91.1	70.1	77.7	64.4	66.0	63.6

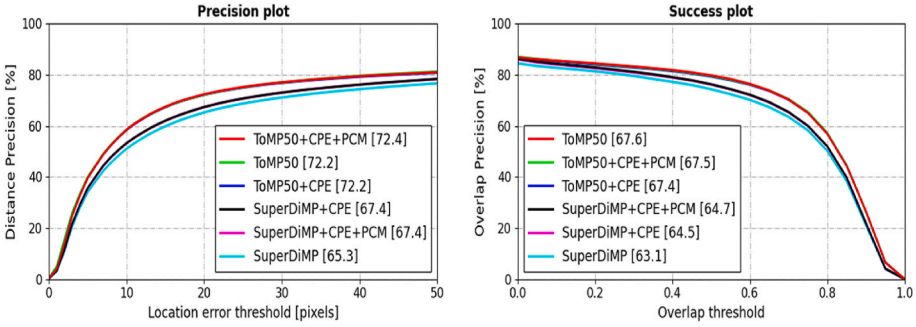

Fig. 8. Illustration of Precision plot and Success plot on LaSOT. The legend displays the Precision and AUC scores for diverse improved trackers and baseline trackers.

Table 7
Comparison results of our ProDiMP and ProToMP trackers compared with state-of-the-art trackers on the LaSOT, TrackingNet, OTB100, NFS, and UAV123 datasets in terms of Precision (Prec.), Normalized Precision (Norm. Prec.) and AUC scores (%).

	LaSOT		TrackingNet			OTB100	NFS	UAV123
	Prec.	AUC	Prec.	Norm. Prec.	AUC	AUC	AUC	AUC
DiMP[5]	56.7	56.9	68.7	80.1	74.0	68.4	62.0	65.3
PrDiMP[25]	60.8	59.8	70.4	81.6	75.8	69.6	63.5	68.0
TrDiMP[3]	66.3	63.9	73.1	83.3	78.4	**71.1**	66.5	67.5
TransT[9]	69.0	64.9	**80.3**	**86.7**	81.4	69.4	65.7	69.1
STARK-ST50[8]	71.2	66.4	-	86.1	81.3	68.5	65.2	69.1
Sim-B/32[24]	-	66.2	**83.9**	-	79.1	-	-	-
CSWinTT[51]	70.9	66.2	79.5	**86.7**	**81.9**	-	-	**70.5**
SuperDiMP[5, 25]	65.3	63.1	73.3	83.5	78.1	70.1	65.2	67.7
ToMP50[2]	**72.2**	**67.6**	78.6	**86.2**	81.2	70.1	**66.9**	69.0
ProDiMP	67.4	64.7	74.0	83.9	78.5	**70.4**	64.9	**69.2**
ProToMP	**72.4**	**67.5**	79.6	**86.7**	**81.8**	70.2	**67.5**	68.9

SuperDiMP and ToMP50 achieve competitive performance, outperforming SuperDiMP with a relative gain of 2.1 % and 1.4 % (%) respectively. In addition, compared with Transformer-based trackers, suach as TrDiMP, TransT, STARK-ST50 and ToMP50, ProToMP exhibits competitive performances. From the results in Table 7 and it can be seen that there is a significant improvement on the pure CNN-based tracker, such as SuperDiMP, while the Transformer-based tracker does not show significant improvement on this dataset. However, ProToMP can still achieve good results. Overall, it can be seen that the proposed methods have a certain significance in improving tracking performances.

TrackingNet (Matthias et al., 2018): The test set of the TrackingNet dataset also is used to test our method, which is comprised of 511 video sequences. The experimental results are provided in Table 7. Our ProToMP tracker achieves AUC score of 81.8 %, outperforming the state-of-the-art trackers such as TrDiMP(Wang et al., 2021), TransT (Chen et al., 2021), STARK-ST50 (Yan et al., 2021), Sim-B/32 (Chen et al., 2022b)and ToMP50 (Christoph et al., 2022). In addition, compared with SuperDiMP, we can see that our ProDiMP promotes the SuperDiMP with a relative AUC gain of 0.4 %, and increasing by 0.7 and 0.4 (%) in Precision and Normalized Precision. Moreover, our ProToMP also obtains competitive performance compared to ToMP50(Christoph et al., 2022), which promotes the ToMP50 with relative increases of 1.0, 0.5, and 0.6 (%) in Precision, Normalized Precision, and AUC respectively. The competitive results on this dataset further demonstrate the effectiveness of our proposed methods.

OTB100(Yi et al., 2015)**:** OTB100 is comprised of 98 video sequences with 100 tracked targets. Table 7 provides the AUC scores of our ProDiMP and ProToMP compared with state-of-the-art trackers. As can be seen from the table contents, our ProDiMP and ProToMP achieve an AUC score of 70.4 and 70.2 (%), surpassing SuperDiMP and ToMP50 with a 0.3 and 0.1 (%) AUC gain. Moreover, ProDiMP outperforms the recent SOTA trackers (*e.g.*, STARK-ST50 (Yan et al., 2021), TransT (Chen et al., 2021)). Moreover, DiMP, PrDiMP, STARK and ToMP use a ResNet50 as backbone by default for a fair comparison. The experimental results further prove the effectiveness of the proposed methods.

NFS(Galoogahi Hamed et al., 2017)**:** We evaluate our tracker on the 30 fps (frames per second) version of the dataset. This dataset is composed of 100 video sequences with fast-moving target objects. As is shown in Table 7, our ProToMP acquires SOTA performance, which gets an AUC score of 67.5 %, surpassing ToMP50 with a 0.6 % gain. Moreover, our method also outperforms the recent TransT and STARK.

UAV123 (Mueller et al., 2016)**:** The UAV dataset is comprised of 123 aerial videos, which are aimed at testing trackers applicable to UAVs. Among the compared methods, Table 7 reveals that our ProDiMP and ProToMP trackers achieve an AUC score of 69.2 and 68.9 (%), which obtains competitive performance compared with the other state-of-the-art approaches (*e.g.*, STARK). Although the tracking performance of ProToMP is not as good as CSWinTT and ToMP50, our ProDiMP surpasses SuperDiMP with a gain of 1.5 %.

VOT2020 (Matej et al., 2020): we evaluate our ProDiMP and ProToMP trackers on the 2020 edition of the Visual Object Tracking (VOT) short-term challenge. Although the dataset contains 60 videos with segment mask annotation, we only compare it with the tracker that predicts the bounding box, because our tracker will predict the bounding box. The trackers are evaluated in the multi-start protocol and are ranked based on the EAO metric that considers tracking accuracy (average overlap over successful frames) and robustness (failure rate). The results in Table 8 show that ProDiMP achieves the first accuracy and EAO. Our ProToMP acquires the first robustness. Especially, ProDiMP achieves a 0.798 robustness and 0.311 EAO, surpassing SuperDiMP in terms of robustness (+7 %) and EAO (+0.6 %). And ProToMP achieves a 0.815 robustness and 0.305 EAO, which is superior to ToMP50 in terms of robustness (+2.6 %) and EAO (+0.8 %).

4.4. Qualitative analysis

The above ablation and SOTA experiments mainly demonstrate the effectiveness of the proposed methods from a quantitative perspective, and can effectively improve tracking results. Further, we also qualitatively demonstrate the tracking effect of representative video sequences in Fig. 9. We visualize the tracking results of SuperDiMP, ToMP50 and our trackers (*i.e.*, ProDiMP and ProToMP) on the 4 challenging sequences from OTB100 and LaSOT. We select two video sequences (*i.e.*, Basketball and Skating2_1) from the OTB100 short-term dataset and two video sequences (*i.e.*, dog-15 and zebra-17) from the LaSOT long-term dataset for qualitative analysis. As shown in Fig. 9, the number of video frame for the four video sequences are 725, 473, 5009, and 2964 respectively. We display four representative frames, among which the first frame shows the bounding box for the tracked target. We can see from Fig. 9 that our ProDiMP and ProToMP trackers can accurately predict the target objects despite appearing occlusion, deformation and background clutter. In frame #0470 of Basketball sequence, our trackers mistake the similar distractor as the target, but SuperDiMP and ToMP50 can also obtain improper bounding boxes. In Skating2_1 sequence, we note that our trackers can basically track the female skater in the presence of distractor, yet our SuperDiMP and ToMP50 method will be disturbed by background clutter. In frame #0110 of dog-15 sequence, SuperDiMP will lose the tracked target and mistakes the dog in the mirror for the target. In zebra-17 sequence, SuperDiMP and ToMP50 is not robust enough and can be affected by similar objects and self-deformation. Moreover, in frame #1258, SuperDiMP gradually tracks the background information, while ToMP50 only tracks the part of the target. However, although our trackers are closer to the tracked area for Ground Truth, updating the model as usual at this time will also cause contamination of the tracking model. From frame #2934, it can be seen that the compared trackers still maintain the correct tracking of the target. We have found that the tracking bounding box is not accurate enough and surrounding objects are erroneously prone to be identified as the target in complex scenarios. Although our tracker has meritorious robustness to complex situations such as deformation, distractor, and background clutter, we also found that there is a gap from the labeled bounding box.

4.5. Discussion

Sec. 4.2 and 4.3 mainly focus on quantitative experimental

Table 8
State-of-the-art comparison of bounding box-only methods on VOT2020ST in terms of Accuracy (A), Robustness (R), and expected average overlap (EAO).

	DiMP	TrDiMP	STARK-ST50	CSWinTT	SuperDiMP	ToMP	**ProDiMP**	**ProToMP**
A	0.457	0.471	0.477	**0.480**	0.477	0.453	**0.482**	0.452
R	0.734	0.782	**0.799**	0.787	0.728	0.789	0.798	**0.815**
EAO	0.274	0.300	**0.308**	0.304	0.305	0.297	**0.311**	0.305

Fig. 9. Visualization results of the proposed trackers (*i.e.*, ProDiMP and ProToMP) and Baselines (*i.e.*, SuperDiMP and ToMP50) on several challenge sequences from OTB100 and LaSOT datasets. We can see that our trackers can achieve a more accurate tracking effect.

comparison and analysis. Sec. 4.4 visualizes the tracking effect from a qualitative perspective and analyzes the results. We conduct a comprehensive discussion based on the experimental results of three Sections. Our main contributions in this article are to propose a Color Prompt Encoder (CPE) to improve insufficient feature expression and a Position Constraint Mechanism (PCM) to improve inaccurate updates.

In the aspect of CPE, mainly including the acquisition of color prompts and how to encode color prompts into deep networks, our ablation experiments on OTB100, NFS, and LaSOT datasets are conducted by integrating the proposed CPE and PCM methods into SuperDiMP and ToMP50 trackers. In the acquisition of Color Prompt (CP), we experiment with four different methods of updating CP and whether or not there was a Hanning Window. The experimental results show that appropriate updates have a positive effect on tracking performance. Based on the results in Tables 1 and 2, we have decided to use the moving average update CP method with the Hanning Window method as the optimal color prompt. Immediately after, we compare the multiple encoding methods: BI_MLP, PE_MLP, PE_ATT, PE_*cat*, and PE_*add*, and the first three methods are elaborated in Sec. 3.3. Based on the LaSOT result of Fig. 7, we believe that BI_MLP is a more effective method for fusing color prompt features with deep features.

In the aspect of PCM, we propose a low-cost and portable position constraint mechanism, our experimental results in Table 4 show that it can effectively improve the tracking performance of SuperDiMP, but when combined with CPE, it will result in lower tracking performance on NFS dataset. In addition, the hyperparameter experiment of the constraint radius π also indicates that the selection of constraint radius has a significant impact on the effectiveness of PCM. Specifically, Table 5 also conducts experiments on the multi-frame position offset constraint, while Table 6 examines the impact of the experimental average and maximum aggregation methods on the 3-frame PCM. In fact, the results of 3-frame are also acceptable, but the LaSOT method is ultimately chosen for single frame PCM method.

From the results of various ablation experiments, it can also be seen that our CPE and PCM modules using different settings on OTB100, NFS, and LaSOT will produce diverse results, making it difficult to guarantee which configuration can achieve ideal results on all datasets. However, overall, the proposed method undoubtedly improves tracking performance. Based on the results of the ablation experiments, further SOTA experiments are conducted on six datasets. Since our tracking frameworks is based on CNN (*i.e.*, SuperDiMP) and the combination of CNN and Transformer (*i.e.*, ToMP50), we mainly compared our trackers with this type of SOTA methods. The results presented in Tables 7 and 8 indicate that our proposed ProDiMP and ProToMP exhibit superior performance compared to SuperDiMP and ToMP50. Moreover, compared with TrDiMP, TransT, STARK, CSWinTT, etc. methods, our trackers can obtain SOTA results.

Normally, the effectiveness of tracking performance is mainly measured by the Precision and AUC results on mainstream tracking benchmarks, Fig. 9 also presents the visual tracking effect of our proposed trackers and baseline trackers. Experimental visualization and analysis show that our trackers can effectively alleviate the interference of complex scenes such as similar objects and deformations. Although our tracker can effectively improve the accuracy of bounding boxes, the PCM method has shortcomings for occlusion situations. In the future, we will further consider the method of comparing appearance feature similarity for joint decision-making. In addition, the proposed CPE currently mainly enhances the target-aware ability of CNN, and it is worth exploring how to use Transformer architecture to achieve the fusion of color prompt features.

5. Conclusions

In this paper, we are devoted to enhancing the discriminative ability of the recent DCF-based trackers from the perspective of the feature extraction and model update. To this end, we construct color prompt features and propose three encoding methods, including BI_MLP, PE_MLP, and PE_ATT, to implement the construction of the color prompt encoder. The color prompt encoder in the feature extraction stage can yield the target-aware feature expression for training the robust target model. Moreover, we propose a practical position constraint method for storing samples with high confidence and updating the model properly. The position constraint method experiments with constraint radius and multi-frame position offset cases. Ultimately, the ablation study and SOTA comparison indicate that our proposed methods are feasible, but we also found that there are differences in performance gains between different scheme settings and method combinations. Overall, our methods achieve state-of-the-art performances on six benchmarks, showing the potential ability of our approaches.

In the future, in terms of feature extraction, we will discuss replacing CNN with ViT backbone and improving the color prompt encoding method to achieve stronger feature fusion and performance improvement. In terms of online updates, the position offset constraint can acquire not satisfactory outcome when there is occlusion, so we plan to propose a joint update decision based on appearance feature matching. Moreover, we will further explore the improvement and application of color prompt and position constraint approaches in pure Transformer-based trackers to get stronger feature representation.

CRediT authorship contribution statement

Xuedong He: Writing – original draft, Visualization, Methodology, Data curation, Conceptualization. **Huiying Xu:** Writing – review & editing, Validation, Supervision, Conceptualization. **Xinzhong Zhu:** Writing – review & editing, Project administration, Funding acquisition. **Hongbo Li:** Visualization, Validation, Data curation. **Xiao Huang:** Writing – review & editing, Software, Methodology. **Yunliang Jiang:** Writing – review & editing, Resources, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 62376252, 62473338), Natural Science Foundation of Zhejiang Province (Grant No. LZ22F030003, LQN25F030016), Zhejiang Province Leading Geese Plan (2024C02G1123882, 2024C01SA100795), Jinhua Science and Technology Bureau (Grant No. 2024-4-006), Young Doctoral Program (Grant No. 2023QB019) and research start-up funds from Zhejiang Normal University. We also thank the editors and the anonymous reviewers for their valuable comments to improve this article.

Data availability

I have shared the link to my code/data

References

Alexey, Dosovitskiy, Lucas, Beyer, Alexander, Kolesnikov, et al., 2021. An image is worth 16x16 words: transformers for image recognition at scale. In: International Conference on Learning Representations, pp. 1–21.

Ashish, Vaswani, Noam, Shazeer, Niki, Parmar, et al., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 5998–6008.

Bertinetto, Luca, Valmadre, Jack, Henriques, João F., et al., 2016a. Fully-convolutional siamese networks for object tracking. In: Proceedings of the European Conference on Computer Vision Workshops, pp. 850–865.

Bertinetto, L., Valmadre, J., Golodetz, S., et al., 2016b. Staple: complementary learners for real-time tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1401–1409.

Bhat, Goutam, Joakim, Johnander, Martin, Danelljan, et al., 2018. Unveiling the power of deep tracking. In: Proceedings of the European Conference on Computer Vision, pp. 493–509.

Bhat, Goutam, Martin, Danelljan, Luc, Van Gool, et al., 2019. Learning discriminative model prediction for tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6182–6191.

Bhat, Goutam, Martin, Danelljan, Luc, Van Gool, et al., 2020. Know your surroundings: exploiting scene information for object tracking. In: Proceedings of the European Conference on Computer Vision, pp. 205–221.

Bin, Yan, Zhang, Xinyu, Dong, Wang, et al., 2021. Alpha-refine: boosting tracking performance by precise bounding box estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5289–5298.

Bo, Li, Junjie, Yan, Wu, Wei, et al., 2018. High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8971–8980.

Bolme, David S., Beveridge, J. Ross, Draper, Bruce A., et al., 2010. Visual object tracking using adaptive correlation filters. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2544–2550.

Cai, Wenrui, Liu, Qingjie, Wang, Yunhong, 2024. HIPTrack: visual tracking with historical prompts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19258–19267.

Chen, Xin, Bin, Yan, Zhu, Jiawen, et al., 2021. Transformer tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8126–8135.

Chen, Z., Zhong, B., Li, G., et al., 2022a. SiamBAN: target-aware tracking with siamese box adaptive network. IEEE Trans. Pattern Anal. Mach. Intell. 5158–5173.

Chen, Boyu, Peixia, Li, Lei, Bai, et al., 2022b. Backbone is all your need: a simplified architecture for visual object tracking. In: European Conference on Computer Vision, pp. 375–392.

Christoph, Mayer, Martin, Danelljan, Pani, Paudel Danda, et al., 2021. Learning target candidate association to keep track of what not to track. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 13444–13454.

Christoph, Mayer, Martin, Danelljan, Bhat, Goutam, et al., 2022. Transforming model prediction for tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8731–8740.

Cui, Yutao, Song, Tianhui, Wu, Gangshan, et al., 2023. MixFormerV2: efficient fully transformer tracking. Adv. Neural Inf. Process. Syst. 58736–58751.

Cui, Yutao, Cheng, Jiang, Wu, Gangshan, et al., 2024. MixFormer: end-to-end tracking with iterative mixed attention. IEEE Trans. Pattern Anal. Mach. Intell. 46 (6), 4129–4146.

Dai, Kenan, Zhang, Yunhua, Dong, Wang, et al., 2020. High-performance long-term tracking with meta-updater. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6298–6307.

Danelljan, M., Häger, G., Khan, F.S., et al., 2015. Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4310–4318.

Fan, Heng, Lin, Liting, Fan, Yang, et al., 2019. LaSOT: a high-quality benchmark for large-scale single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5369–5378.

Galoogahi Hamed, Kiani, Ashton, Fagg, Chen, Huang, et al., 2017. Need for speed: a benchmark for higher frame rate object tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1125–1134.

Gao, Shenyuan, Chunluan, Zhou, 2023. Zhang Jun generalized relation modeling for transformer tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18686–18695.

He, Xuedong, Chen, Calvin Yu-Chian, 2022. Exploring reliable visual tracking via target embedding network. Knowl. Base Syst. 244, 108584.

He, Xuedong, Huang, Jiehui, 2025. Color attention tracking with score matching. International Journal of Machine Learning and Cybernetics 16 (2), 983–997.

He, Kaiming, Zhang, Xiangyu, Shaoqing, Ren, et al., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 770–778.

He, Xuedong, Yu-Chian, Chen Calvin, 2022. Learning object-uncertainty policy for visual tracking. Inf. Sci. 582, 60–72.

He, Xuedong, Lu, Zhao, Chen, Calvin Yu-Chian, 2023a. Variable scale learning for visual object tracking. J. Ambient Intell. Hum. Comput. 14, 3315–3330.

He, Xuedong, Yu-Chian, Chen Calvin, 2023b. Attention fusion and target-uncertain detection for discriminative tracking. Knowl. Base Syst. 278, 110860.

Henriques, João F., Caseiro, Rui, Martins, Pedro, et al., 2015. High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. 37 (3), 583–596.

Huang, Lianghua, Xin, Zhao, Huang, Kaiqi, 2021. GOT-10k: a large high-diversity benchmark for generic object tracking in the wild. IEEE Trans. Pattern Anal. Mach. Intell. 43 (5), 1562–1577.

Javed, Sajid, Danelljan, Martin, Khan, Fahad Shahbaz, et al., 2023. Visual object tracking with discriminative filters and siamese networks: a survey and outlook. IEEE Trans. Pattern Anal. Mach. Intell. 45, 6552–6574.

Jun, Wang, Lai, Changwang, Wang, Yuanyun, et al., 2024. EMAT: efficient feature fusion network for visual tracking via optimized multi-head attention. Neural Netw. 172, 106110.

Kang, Ben, Chen, Xin, Dong, Wang, et al., 2023. Exploring lightweight hierarchical vision transformers for efficient visual tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9612–9621.
Liu, Ze, Lin, Yutong, Cao, Yue, et al., 2021. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 10012–10022.
Ma, C., Huang, J., Yang, X., et al., 2015. Hierarchical convolutional features for visual tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3074–3082.
Martin, Danelljan, Goutam, Bhat, Shahbaz, Khan Fahad, et al., 2017. ECO: efficient convolution operators for tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6638–6646.
Martin, Danelljan, Goutam, Bhat, Shahbaz, Khan Fahad, et al., 2019. ATOM: accurate tracking by overlap maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4660–4669.
Martin, Danelljan, Van, Gool Luc, Radu, Timofte, 2020. Probabilistic regression for visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7183–7192.
Matej, Kristan, Aleš, Leonardis, Jiří, Matas, et al., 2020. The eighth visual object tracking VOT2020 challenge results. In: Proceedings of the European Conference on Computer Vision Workshops, pp. 547–601.
Matthias, Müller, Adel, Bibi, Silvio, Giancola, et al., 2018. TrackingNet: a large-scale dataset and benchmark for object tracking in the wild. In: European Conference on Computer Vision, pp. 310–327.
Mueller, Matthias, Smith, Neil, Bernard, Ghanem, 2016. A benchmark and simulator for UAV tracking. In: European Conference on Computer Vision, pp. 445–461.
Song, Zikai, Junqing, Yu, Chen, Yi-Ping Phoebe, et al., 2022. Transformer tracking with cyclic shifting window attention. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8791–8800.
Tian, Zhi, Shen, Chunhua, Chen, Hao, et al., 2019. FCOS: fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 9627–9636.
Wang, Mengmeng, Liu, Yong, 2017. Zeyihuang. Large margin object tracking with circulant feature maps. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4021–4029.
Wang, Ning, Zhou, Wengang, Wang, Jie, et al., 2021. Transformer meets tracker: exploiting temporal context for robust visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1571–1580.
Wang, Jun, Yang, Shuai, Wang, Yuanyun, 2024. Dynamic region-aware transformer backbone network for visual tracking. Eng. Appl. Artif. Intell. 133, 108329.
Xie, Jinxia, Zhong, Bineng, Zhiyi, Mo, et al., 2024. Autoregressive queries for adaptive tracking with spatio-temporal transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 19300–19309.
Yan, Bin, Peng, Houwen, Fu, Jianlong, et al., 2021. Learning spatio-temporal transformer for visual tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10448–10457.
Ye, Botao, Hong, Chang, Ma, Bingpeng, et al., 2022. Joint feature learning and relation modeling for tracking: a one-stream framework. In: European Conference on Computer Vision, pp. 341–357.
Yi, Wu, Lim, Jongwoo, Ming-Hsuan, Yang, 2015. Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. 37 (9), 1834–1848.