

CiteTracker: Correlating Image and Text for Visual Tracking

Xin Li^{1,†}, Yuqing Huang^{1,2,†}, Zhenyu He^{2,*}, Yaowei Wang^{1,*}, Huchuan Lu³, and Ming-Hsuan Yang^{4,5}
¹Peng Cheng Laboratory ²Harbin Institute of Technology, Shenzhen
³Dalian University of Technology ⁴UC Merced ⁵Yonsei University

Abstract

Existing visual tracking methods typically take an image patch as the reference of the target to perform tracking. However, a single image patch cannot provide a complete and precise concept of the target object as images are limited in their ability to abstract and can be ambiguous, which makes it difficult to track targets with drastic variations. In this paper, we propose the CiteTracker to enhance target modeling and inference in visual tracking by connecting images and text. Specifically, we develop a text generation module to convert the target image patch into a descriptive text containing its class and attribute information, providing a comprehensive reference point for the target. In addition, a dynamic description module is designed to adapt to target variations for more effective target representation. We then associate the target description and the search image using an attention-based correlation module to generate the correlated features for target state reference. Extensive experiments on five diverse datasets are conducted to evaluate the proposed algorithm and the favorable performance against the state-of-the-art methods demonstrates the effectiveness of the proposed tracking method. The source code and trained models will be released at <https://github.com/NorahGreen/CiteTracker>.

1. Introduction

Visual object tracking aims to estimate the state (location and extent) of an arbitrary target in a video sequence based on a specified region of the target in the initial frame as a reference point. It is challenging to locate a target that undergoes drastic appearance variations (*e.g.*, changes in pose, illumination, or occlusions) using only one target image sample since the target appearances can be significantly different. To successfully track the target with appearance changes, it is crucial to acquire a comprehensive representation of the target for establishing associations between the target exemplar and the target in test frames.

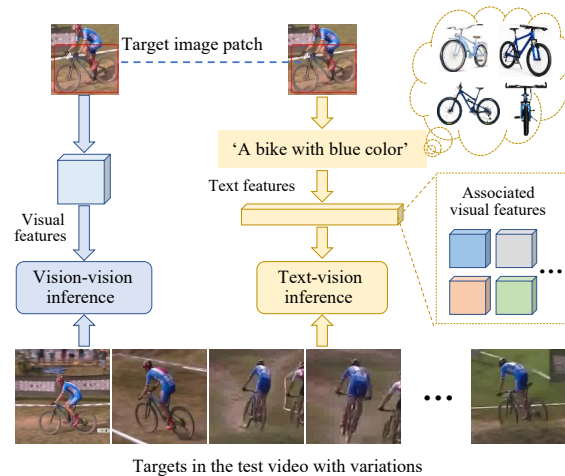


Figure 1. Comparison of the proposed algorithm and existing tracking methods in terms of target modeling and association. The left and right parts depict the typical visual tracking framework and the proposed one, respectively. Our approach first generates a text description of the target object and then takes the feature of the text to estimate the target state in the test image, enabling a more comprehensive target modeling and association.

Most existing deep trackers [2, 19, 36, 7, 39] learn an embedded feature space, where target samples with different appearances are still close to each other, to generate a robust representation to target variations. To build a more comprehensive target representation and better associate the target exemplar with the test target, several recent trackers [6, 40, 23] perform interaction of the target template and search region in every block of their feature extraction backbone, achieving state-of-the-art tracking performance. However, these methods do not perform well when the target changes drastically or the given target exemplars are of low-quality (*i.e.*, the appearance of the target exemplar is uncommon or the exemplar contains background noise).

The following issues arise when using an image patch as the target reference for tracking. First, the visual representation of a target is insufficient to provide a comprehensive understanding for recognizing the target with appear-

[†] Equal contribution, * corresponding author

ance variations, since images are limited in their ability to abstract. An image patch of a target only captures its appearance from a particular angle, but its shape, texture, and surface features can vary significantly when viewed from different angles, resulting in a completely different appearance that makes it difficult to track the target. Second, as images can be ambiguous and open to interpretation, a random target image patch can mislead tracking models by causing them to overemphasize certain unstable appearance features and ignore the more essential and stable features of the target, resulting in drifting to the background and tracking failures. For example, when tracking a circular object, the target patch may include a lot of the background, which causes the tracker to drift to the background.

We note that the human-created language signal provides a more abstract and precise concept of an object compared to the image signal, which has the potential to solve the aforementioned issues. In addition, the study [30] on connecting language and images shows that text and image features can be well-aligned and transferred to each other, allowing for using the advantages of both language and image signals for visual tracking. Motivated by these insights, we study correlating text and images for visual tracking.

In this paper, we propose a new tracking framework that uses an adaptive text description of the target as the reference point and correlates it with test image features to perform tracking, named as CiteTracker. Specifically, we first develop a text generation model via prompt learning with a pre-defined open vocabulary including class and attribute labels, enabling generating the text description of the target based on a target image patch. The generation model is built using the CLIP model as the baseline, which already connects text with rich image features. To adapt to target variations over time, we develop a dynamic text feature model that generates adaptive text features along the change of the target. Finally, we associate the features of the target text description with test image features to generate the correlated features for further target state estimation. We conduct extensive experiments on a variety of public datasets including GOT-10K [18], LaSOT, TrackingNet, OTB100, and TNL2K to evaluate the proposed algorithm. The favorable performance against the state-of-the-art methods on all the datasets demonstrates the effectiveness of correlating images and text for visual tracking.

We make the following contributions in this paper:

- We propose a text-image correlation based tracking framework. We use a text description to provide a more comprehensive and precise concept of the target and correlate the text with the test image for inferring the target location, enabling a more powerful ability to handle the target variation issue.
- We develop an adaptive feature model for target descriptions to better adapt to target variations in test

videos, contributing to more precise target features and more accurate tracking performance.

- We achieve state-of-the-art performance on numerous tracking datasets. We conduct extensive experiments including ablation studies to demonstrate the effectiveness of the proposed method and the effect of every component.

2. Related Work

We discuss closely related studies including deep visual tracking methods, language-based tracking approaches, and language-image association models.

Deep visual-based trackers. Deep visual tracking methods can be broadly categorized as correlation-based or model-prediction-based, depending on how they model and infer the target during tracking. The correlation-based trackers [20, 42, 8, 19, 41, 35] associate the target exemplar with the test image in embedded feature space to generate the correlated features and estimate the target state upon the correlated features using classification and box-regression prediction heads. As the correlated features show the similarity of every location of the test image to the target exemplar, their quality directly affects tracking performance. By using more powerful features [34, 12, 5, 23], the transformer-based trackers [7, 36, 39] significantly improve the tracking performance of previous CNN-based trackers. In addition, several recent transformer-based tracking methods [40, 6, 9] improve the association way by interacting the target exemplar and the test image in every transformer block for a more comprehensive correlation, achieving state-of-the-art performance. The model-prediction-based trackers [10, 3, 11, 27] learn to generate a classifier model based on the given target image patch and the backgrounds in the initial frame. These methods are effective at distinguishing the target from the background by specifically learning the difference between them, resulting in a strong discriminative ability. Different from the above deep visual trackers, the proposed method models the target also with language and infers the target via the language-image correlation, which enables a more comprehensive target modeling and robust target inference.

Language-based trackers. Several methods [21, 37, 14, 26, 16] explore utilizing language signals to facilitate visual object tracking. Some of them use the language signal as an additional cue and combine it with the commonly used visual cue to compute the final tracking result. The SNLT tracker [14] first exploits visual and language descriptions individually to predict the target state and then dynamically aggregates these predictions for generating the final tracking result. In [37], Wang *et al.* propose an adaptive switch-based tracker that switches to a visual grounding module when the target is lost and switches back to a

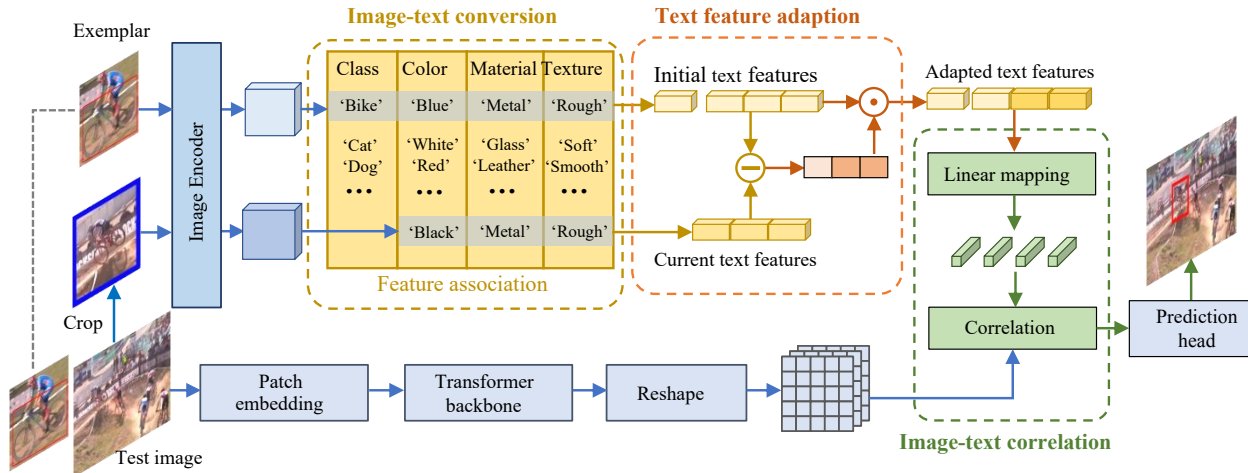


Figure 2. **Overall framework of the proposed CiteTracking algorithm.** It contains three modules: 1) the image-text conversion module, which generates text features of the target object based on an image exemplar; 2) the text feature adaption module, which adjusts the weights of attribute descriptions according to the current target state; 3) the image-text correlation module, which correlates the features of the target descriptions and the test image to generate correlated features for target state estimation.

visual tracking module when the target is found, ensuring robust and precise tracking. Another type of method focuses on integrating the visual and textual signals to get an enhanced representation for visual tracking. The CapsuleTNL [26] tracker develops a visual-textual routing module and a textual-visual routing module to promote the relationships within feature embedding space of query-to-frame and frame-to-query for object tracking. In [16], a modality mixer module is developed to learn unified-adaptive visual-language representations for robust vision-language tracking. Despite also utilizing both language and visual information for tracking, our method differs significantly from the above methods in terms of how to generate text descriptions of the target and associate them with the search image to perform tracking. The method proposed develops a CLIP-based model to generate text descriptions from a target image example, which eliminates the need for language annotations and expands the range of potential applications. In addition, we design a dynamic feature reweighting module that adjusts language features based on target appearance changes, leading to more accurate tracking performance.

Vision-language models. Recently, the CLIP model [30], trained on a large and diverse dataset of images and their associated captions, maps images and their corresponding text descriptions into a shared feature space, where the similarity between the two modalities can be measured. This shared feature space allows the model to perform various tasks, such as image generation [31], few-shot learning [33], and image captioning [1]. CLIP-based methods achieve state-of-the-art performance on several benchmarks

for these tasks, demonstrating their ability to generalize to new domains and languages. Based on the CLIP model, we develop a dynamic text feature generation module to enable more comprehensive target modeling with more accurate and informative representations for visual tracking. In addition, as text and image features are well aligned in the CLIP model, we correlate the text features of the target with the search image features for reasoning about the location of the target, achieving more robust tracking performance.

3. Proposed Algorithm

The goal of our approach is to construct a robust association between a given target image patch and a search image in a tracking sequence by formulating it as an image-text correlation allowing a more comprehensive understanding of the target state, which helps cope with various appearance variations of the target object. To this end, our CiteTracker first generates text features of the target based on the given target image patch via the proposed image-text conversion module, then adjusts the text features according to the latest state of the target, and finally associates the features of the text and the search image for robust tracking.

3.1. Overall Framework

Figure 2 shows the overall framework of the CiteTracker, consisting of three core modules: an image-text conversion module, a text feature adaption module, and an image-text correlation part.

Taken as input an exemplar image and a search image in a test sequence, our approach processes them with a text branch (the upper part in Figure 2) and a visual branch (the

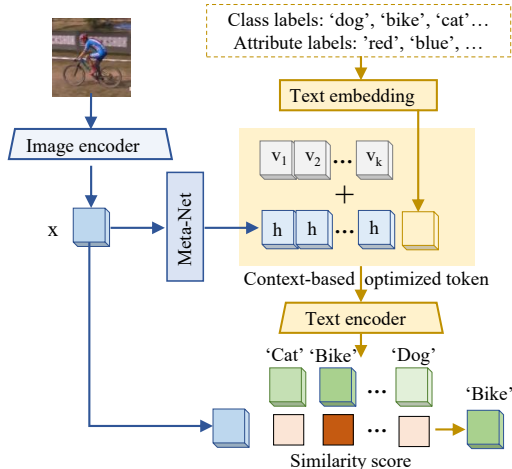


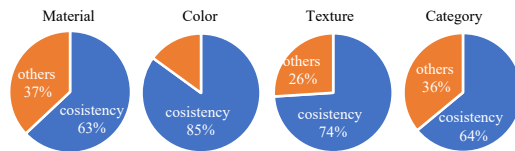
Figure 3. **Structure of the image-text conversion module.** It takes a target image and pre-defined class attribute text vocabularies as input and outputs text descriptions (features) of the target.

lower part in Figure 2). The text branch first uses an image encoder to extract the visual features of the given exemplar image and a target image patch cropped from the test image at the target location in the previous frame. Then, it converts the visual features of the target to text features with the image-text conversion module and adjusts the text features using the text feature adaption module based on the difference between the text features of the initial target state and the current target state. The visual branch adopts the same processing flow as OSTRack [40] that takes both the exemplar and search images as input and outputs a feature map of the test image. Finally, the image-text correlation component associates the outputs of the text and visual branches to generate the correlated features for target state prediction via a commonly used prediction head [39].

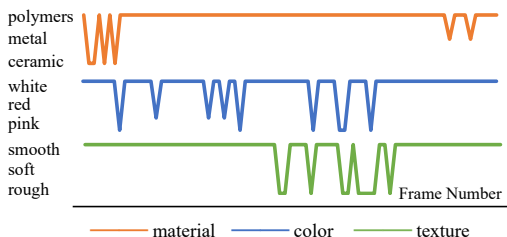
3.2. Image-Text Conversion

To generate the text feature of a tracking target from a given image exemplar, we construct an image-text conversion model that connects images and text based on the CLIP model [30] via prompt learning.

Image-text association learning. Figure 3 shows the structure of the image-text conversion model. It takes the target image and vocabulary of object categories and attributes as input. The target image is processed by the image encoder of a CLIP model to generate the image feature x and x is then fed into a lightweight neural network $h_\theta(\cdot)$ (Meta-Net) to generate target tokens $h_\theta(x)$ that contain the target information. The input vocabulary is processed by a text embedding module to generate the word embeddings c_i . In addition, K learnable vectors v_1, v_2, \dots, v_K , where v_i have the same dimensions as c_i , are introduced as prompt tokens for a specific prediction task. Given the target tokens $h_\theta(x)$



(a) Consistency of predictions on the GOT-10K training dataset.



(b) Temporal consistency on a video sequence.

Figure 4. **Consistency of the predicted descriptions in terms of target category and the selected attributes on tracking videos.**

and the prompt tokens v_1, v_2, \dots, v_K , each context-based optimized token can be obtained by $v_k(x) = v_k + h_\theta(x)$, where $k \in \{1, 2, \dots, K\}$. The prompt for the i -th class label is thus conditioned on the image features, i.e. $m_i(x) = \{v_1(x), v_2(x), \dots, v_K(x), c_i\}$. Let $t(\cdot)$ denote the original CLIP text encoder and the prediction probability of the i -th class label is computed as

$$p(c_i|x) = \frac{\exp(S_{im}(x, t(m_i(x)))/\tau)}{\sum_{j=1}^N \exp(S_{im}(x, t(m_j(x)))/\tau)}, \quad (1)$$

where $S_{im}(\cdot, \cdot)$ computes the cosine similarity score, τ is a learned temperature parameter, and N is the number of class labels. The target description is predicted as the label corresponding to the max probability computed with Equation 1. In this work, we implement the Meta-Net using a two Linear-ReLU-Linear structure with the hidden layer reducing the input dimension by 16 times.

Tracking-related vocabulary construction. In order to accurately describe tracking targets, we choose 80 category labels in the MS COCO [25] dataset as the category vocabulary, which contains the most frequently occurring objects in daily life. In addition, we select three kinds of object attributes including color, texture, and material from the OVAD [4] dataset to caption detailed target states. We evaluate the consistency of the predicted descriptions in terms of the class and attribute labels on the GOT-10k dataset. Figure 4(a) shows the proportions of cases where the predicted results are consistent and Figure 4(b) presents the predicted values of a target object in video frames. They demonstrate that the predicted text descriptions of tracking objects in terms of class and attribute values are consistent in video sequences, which can be used as features for target

localization.

3.3. Dynamic Text Feature Generation

In a video, the category of tracking target remains consistent but its states may vary. Therefore, we divide text feature generation into category feature generation and attribute feature generation. For category feature T_c , let T_i is the text feature of i -th class label generated by the CLIP text encoder, T_c can be computed as

$$T_c = \sum_{i=1}^N p_i * T_i, \quad (2)$$

where p_i is the prediction probability of every category label using Equation 1. For each attribute feature T_a , which has the highest prediction probability, can be computed as

$$\begin{aligned} index &= \operatorname{argmax}(p_i), i \in (1, N), \\ T_a &= T_{index}. \end{aligned} \quad (3)$$

As the attribute values of a tracking target may change, we adjust the weights of different attribute features based on their changes. The changes in terms of color, material, and texture, denoted as D_{color} , $D_{material}$, and $D_{texture}$, respectively, are computed as

$$\begin{aligned} D_{color} &= |R_{color} - S_{color}|, \\ D_{material} &= |R_{material} - S_{material}|, \\ D_{texture} &= |R_{texture} - S_{texture}|, \end{aligned} \quad (4)$$

where $R_{attribute}$ and $S_{attribute}$ represent the probabilities of the reference target and the current test target to be with a specific attribute value computed using Equation 1. The lower the $R_{attribute}$ value, the more similar the target and the search image are on that attribute. Therefore, the attention weights for different attributes are formulated as:

$$W_{att} = \operatorname{Softmax}(-D_{color}, -D_{material}, -D_{texture}). \quad (5)$$

After that, the dynamic text features for different attributes are adjusted as

$$T_{att} = W_{att} * T_a, \quad (6)$$

where T_a is the text feature generated by using Equation 3.

3.4. Image-Text Correlation

The joint visual features $V \in \mathbb{R}^{H \times W \times C}$ of the target and the search image are extracted by the Vision Transformer (ViT-base) [34] model pre-trained with the MAE [17] method. The text features $T \in \mathbb{R}^{1 \times 1 \times C_T}$ are adapted by a linear layer to align with the visual features in the channel dimension. Then the correlation between these two kinds of features is achieved by a convolution operation where the text features $T' \in \mathbb{R}^{1 \times 1 \times C}$ are used as the

kernel weights. The correlated features between the image features and all the text features are added up as the final correlated features for state prediction, which are computed as

$$\begin{aligned} C_{orr}(V, T_c, T_{co}, T_m, T_t) &= \\ (1 + L_c(T_c) + L_{co}(T_{co}) + L_m(T_m) + L_t(T_t)) \odot V, \end{aligned} \quad (7)$$

where \odot denotes the convolutional operation, L_{att} is a linear projection layer for channel adaptation, T_c denotes the category feature, while T_{co} , T_m , and T_t represent the dynamic color, material, and texture feature, respectively.

3.5. State Estimation and Training Objective

State estimation. Based on the correlated features generated by the image-text correlation, our CiteTracker estimates the target state via a commonly used prediction head [40] comprising 4 stacked Conv-BN-ReLU layers. The prediction head outputs a classification score map C , offset maps O for compensating for reduced resolution, and size maps B . Then, the target state is computed as

$$(x, y, w, h) = (x_c + O_x, y_c + O_y, B_w, B_h), \quad (8)$$

where (x_c, y_c) is the target center computed as $(x_c, y_c) = \operatorname{argmax}_{(x,y)} C_{xy}$, (O_x, O_y) denotes the shifts to (x_c, y_c) from O , and (B_w, B_h) is the predicted box size from B .

Training objective. We adopt a similar training process as that of OSTrack [40], which trains the three tasks jointly. We use the weighted focal loss [24], l_1 Loss, and the GIoU [32] loss to train the classification, offset, and box size branches, respectively. The overall loss function is defined as

$$L = L_{cls} + \lambda_{iou} L_{iou} + \lambda_{L1} L_1, \quad (9)$$

where $\lambda_{iou} = 2$ and $\lambda_{L1} = 5$ are used in our experiments.

4. Experiments

In this section, we present the experimental results of the proposed CiteTracker. We first show the overall performance on four large-scale datasets with comparisons against the state-of-the-art trackers. We then investigate the contribution of each component with an exhaustive ablation study. A robustness evaluation is conducted to study the robustness of our tracker to initialization. Finally, the visualized results on a number of challenging sequences are given for providing a comprehensive qualitative analysis.

4.1. Implementation Details

Our experiments are conducted with 4 NVIDIA Tesla V100 GPUs. We adopt the Vision Transformer (ViT-base) [34] model pre-trained using the MAE [17] method as the backbone for extracting visual features. We use the

Table 1. **State-of-the-art comparisons on the datasets of TNL2K, LaSOT, TrackingNet, and GOT-10k.** The best two results are shown in red and blue color. Our approach performs favorably against the state-of-the-art methods on all datasets.

Method	TNL2K [37]		LaSOT [13]			TrackingNet [29]			GOT-10k [18]		
	P	SUC	AUC	P_{Norm}	P	AUC	P_{Norm}	P	AO	$SR_{0.75}$	$SR_{0.5}$
SiamFC [2]	28.6	29.5	33.6	42.0	33.9	57.1	66.3	53.3	34.8	9.8	35.3
RPN++ [19]	41.2	41.3	49.6	56.9	49.1	73.3	80.0	69.4	51.7	32.5	61.6
Ocean [41]	37.7	38.4	56.0	65.1	56.6	-	-	-	61.1	47.3	72.1
TransT [7]	51.7	50.7	64.9	73.8	69.0	81.4	86.7	80.3	67.1	60.9	76.8
KeepTrack [28]	-	-	67.1	77.2	70.2	-	-	-	-	-	-
STARK [39]	-	-	67.1	77.0	-	82.0	86.9	-	68.8	64.1	78.1
OSTrack [40]	-	55.9	71.1	81.1	77.6	83.9	88.5	83.2	73.7	70.8	83.2
SimTrack [6]	55.7	55.6	70.5	79.7	-	83.4	87.4	-	-	-	-
MixFormer [9]	-	-	70.1	79.9	76.3	83.9	88.9	83.1	71.2	65.8	79.9
AiATrack [15]	-	-	69.0	79.4	73.8	82.7	87.8	80.4	69.6	63.2	80.0
SwinTrack [23]	57.1	55.9	71.3	-	76.5	84.0	-	82.8	72.4	67.8	80.5
SNLT [14]	41.9	27.6	54.0	-	57.6	-	-	-	43.3	22.1	50.6
VLT [16]	53.3	53.1	67.3	-	72.1	-	-	-	69.4	64.5	81.1
Ours	59.6	57.7	69.7	78.6	75.7	84.5	89.0	84.2	74.7	73.0	84.3

fine-tuned version of the CLIP model [30] as the backbone to construct the proposed image-text conversion model. We crop the search image that is 4 times the area of the target box from the test frame and resize it to a resolution of 384×384 pixels. While crop only 2 times of that from the reference frame and resize it to 192×192 pixels. The open vocabulary class labels and attribute labels are derived from the MS COCO [25] dataset and OVAD [4] dataset. We train our CiteTracker on the training splits of the TrackingNet [29], COCO2017 [25], LaSOT [13], and GOT-10K [18] datasets, except for the evaluation on GOT-10K, where CiteTracker is only trained on the GOT-10K training set.

4.2. State-of-the-art Comparison

We compare our tracker with the state-of-the-art methods on four diverse datasets including TNL2K, LaSOT, TrackingNet, and GOT-10K. Table 1 shows the results.

TNL2K [37]. TNL2K is a benchmark designed for evaluating natural language-based tracking algorithms. The benchmark introduces two new challenges, *i.e.* adversarial samples and modality switching, which makes it a robust benchmark for tracking algorithm assessment. Although the benchmark provides both bounding boxes and language descriptions, we only use the bounding box for evaluation. Our approach achieves the best performance compared to the state-of-the-art methods including the language-based VLT tracker. Compared to the second-best tracker OSTRack [40], the proposed method improves the performance by gains of 1.8% and 2.5% in terms of success rate (SUC) and precision, respectively. The favorable performance demonstrates the promising potential of our tracker to deal

with adversarial samples and modality switch problems, which benefits from the use of text descriptions to model and inference the tracking target.

LaSOT [13]. LaSOT is a high-quality long-term single object tracking benchmark, with an average video length of more than 2,500 frames. Although our method does not employ any updating mechanisms which play a critical role in long-term tracking, it still achieved a result close to the best method SwinTrack. The proposed CiteTracker focuses on handling drastic target variations by formulating target inference as a robust image-text correlation.

TrackingNet [29]. TrackingNet is a large-scale short-term benchmark for object tracking in the wild, which contains 511 testing videos that sequester the ground truth annotation. Table 1 shows the performance on the TrackingNet dataset. Our tracker achieves 84.4% in area-under-the-curve (AUC), surpassing all previously released trackers. It depicts that our tracker is highly competitive in tracking short-term scenarios in the wild with various changes.

GOT-10k [18]. GOT-10k is a large-scale tracking dataset that contains over 560 classes of moving objects and 87 motion patterns, emphasizing class agnosticism in the test set. The ground truths of the test set are withheld and we use the test platform provided by the authors to evaluate our results. We follow the one-shot protocol training rule that the tracker is trained only on the training set of GOT-10k. As shown in Table 1, our tracker improves all metrics, *e.g.* 1.5% in AUC score compared with OSTRack [40] and SwinTrack [23]. The good performance shows that our tracker has a good generalization ability to track class-agnostic targets. We attribute this to the proposed robust target model-

Table 2. **Ablation study of the proposed algorithm on the OTB, GOT-10k, and TNL2K datasets.** The best results in each part of the table are marked in **bold**.

		Base model	Lang Traker	w/o FT	w/o DDG	w/o attr.	Cite Tracker
OTB	AUC	67.5	68.1	69.1	69.3	69.3	69.6
	P	88.8	90.2	90.4	90.6	90.6	92.2
	NP	82.7	83.5	84.1	84.2	84.4	85.1
GOT-10k	AO	72.9	-	71.4	74.5	74.2	74.7
	SR _{0.75}	70.2	-	68.8	72.5	72.4	73.0
	SR _{0.5}	82.4	-	80.4	83.8	83.5	84.3
TNL2K	P	57.0	58.8	57.5	59.3	59.1	59.6
	SUC	55.9	57.1	56.1	57.6	57.4	57.7

ing approach using text descriptions.

Comparisons with Vision-Language trackers. In addition to the comparisons with the SOTA visual trackers, we compare the proposed method with the SOTA vision-language trackers to verify the effectiveness of the description generation ability. Our tracker improves the tracking performance in all benchmarks by a large margin, *e.g.* 5% in the success rate of TrackingNet benchmark and 5.3% in that of GOT-10k compared with the recently published vision-language tracker VLT [16]. Although we do not use the manually annotated text description, the proposed method with a description generation module can still achieve considerable tracking performance.

4.3. Ablation Study

To evaluate the contribution of each component in our tracker, we conduct the ablation studies with six variants of the CiteTracker:

Base visual model, which only employs the backbone to extract the joint visual features of the target and test images, and a prediction head to predict the final tracking results. Herein, the prediction head is constructed on the feature maps of the joint visual features.

LangTraker, which uses manually annotated target descriptions to track. It extracts description features by the CLIP text encoder and performs a correlation between the extracted description features and visual features obtained from the backbone to acquire the associated features.

W/O attribute (attr.), which only generates category descriptions from the template frame using the image-text conversion model, and then correlates these descriptions with visual features extracted from the backbone to obtain associated features.

W/O dynamic description generation (DDG), which extracts category and attribute descriptions only from the template frame using the image-text conversion model.

W/O fine-tune (FT), which employs the original CLIP model to extract category descriptions and attribute descriptions for the tracking targets.

Table 3. **Robustness evaluation against the OTrack method on the OTB dataset.** Our CiteTracker is more robust to initialization compared to the OTrack method.

AUC (%)	TRE	TRE-worst	SRE-shift	SRE-scale
OTrack	69.3	57.5	45.9	63.1
Ours	(+0.9) 70.2	(+3.6) 61.1	(+1.4) 47.3	(+2.7) 65.8

CiteTracker, our intact model uses an image-text conversion model to obtain category and attribute descriptions for both the template and search frames. Then, these descriptions are correlated with visual features to generate the correlated features for target state estimation.

Table 2 presents the experimental results of these variants on the OTB2015, GOT-10K, and TNL2K datasets. The manually annotated target descriptions of the OTB dataset are from the OTB-lang dataset [22].

Effect of the vision and text feature correlation. The performance gap between the base model and LangTraker clearly demonstrates the advantages of correlating visual and text-based features for tracking.

Effect of the prompt-tuning on the CLIP model. With the prompt-tuning process, CiteTracker achieves performance gains of 0.5% and 3.3% in AUC on OTB2015 and GOT-10K, while 1.6% in SUC on TNL2K, respectively. These improvements validate the benefits of the prompt-tuning of the CLIP model, which generates a more robust representation by exploiting the content-based optimization tokens.

Effect of using the attribute description. Without using the attribute description (w/o attr.), CiteTracker decreases by 1.6% and 0.5% in precision on OTB2015 and TNL2K respectively. It validates the superiority of using the attribute description to model tracking objects.

Effect of the dynamic text-feature generation module. By comparing our CiteTracker with w/o DDG, it is clear that the proposed dynamic text-feature generation module improves tracking performance by 0.3%, 0.2%, and 0.3% in terms of AUC on OTB2015, GOT-10K, and TNL2K, respectively. This mechanism successfully enables the tracker to focus more on the differences between the reference and search frames, leading to improved results.

4.4. Robustness Evaluation

We evaluate the robustness of the proposed method on the OTB dataset by adopting the temporal robustness evaluation (TRE) and the spatial robustness evaluation (SRE) [38]. We additionally report the worst score (TRE-worst) among all sequence segments, which measures the robustness of a tracker to bad initial target samples. SRE-shift and SRE-scale denote the evaluations using shifted ground truth and scaled ground truth, respectively. Table 3 shows that the proposed tracker achieves more robust per-

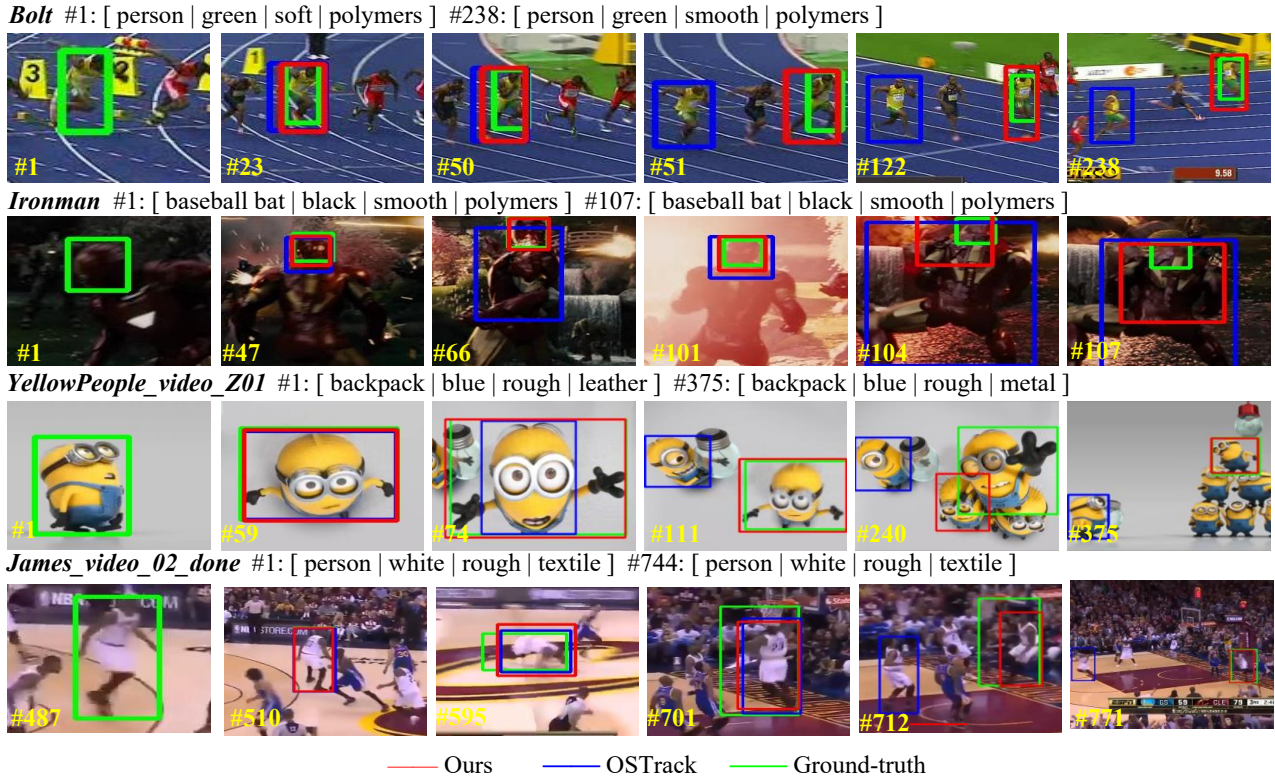


Figure 5. Visualized results of the proposed algorithm and the OSTRack method on four challenging sequences with drastic changes. It shows that our CiteTracker performs well with the aid of the generated text descriptions (shown above each row of pictures), while the OSTRack method with solely visual cues struggles with these sequences.

formance compared to OSTRack, especially in terms of TRE-worst with a gain of 3.6%, which demonstrates that our approach performs significantly well in the cases when the initial target examples are with very poor quality.

4.5. Qualitative Study

To obtain more insights from our proposed tracking algorithm, we visualize the tracking results of several challenging sequences compared with OSTRack.

The *Bolt* sequence is characterized by a swiftly moving target and adversarial examples that closely resemble the reference target. Our tracking algorithm performs accurately in tracking the target, whereas OSTRack fails to track the target at the 51st frame. In the *Ironman* series, our tracker accurately tracks the target despite significant lighting changes, whereas OSTRack does not. In addition, our CiteTracker accurately locates the target and distinguishes it from similar distractors, even in the presence of adversarial samples and target appearance variations in the *YellowPeople* sequence. Despite frequent changes in viewpoint in the *James* sequence, our tracking algorithm still performs well.

Figure 5 additionally shows the generated text descriptions of the targets for each sequence, including category,

color, material, and texture. Most descriptions of the same target are consistent in different frames with drastic changes, which demonstrates the robustness of textual descriptions for tracking. The predicted category may differ from the real object class due to the limited 80 categories from the COCO dataset used for training, but it remains consistent in most frames of a video (is also supported by the statistical results in Figure 4), benefiting target identification and localization.

5. Conclusions

In this work, we present the CiteTracker that performs target modeling and target state inference in a more robust and accurate way by associating images and text. Specifically, the proposed algorithm first constructs an image-text conversion model to generate text-description features of the target from a given target image, enabling a more abstract and accurate target representation. In addition, we develop a text feature adaption model to generate dynamic text features and an image-text correlation to associate the target text and the search image for further target state prediction. Qualitative and quantitative evaluations demonstrate that our approach performs favorably against the state-of-the-art

methods, which suggests that incorporating language signals into visual tracking has a notable effect on improving tracking performance.

6. Acknowledgments

The paper is supported by the National Natural Science Foundation of China (62002241, U20B2052, 62172126), and the Shenzhen Research Council (No. JCYJ20210324120202006).

References

- [1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018. 3
- [2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *ECCVW*, 2016. 1, 6
- [3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *ICCV*, 2019. 2
- [4] María A. Bravo, Sudhanshu Mittal, Simon Ging, and Brox Thomas. Open-vocabulary attribute detection. In *arXiv*, 2022. 4, 6
- [5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2
- [6] Boyu Chen, Peixia Li, Lei Bai, Lei Qiao, Qihong Shen, Bo Li, Weihao Gan, Wei Wu, and Wanli Ouyang. Backbone is all your need: a simplified architecture for visual object tracking. In *ECCV*. Springer, 2022. 1, 2, 6
- [7] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *CVPR*, 2021. 1, 2, 6
- [8] Zedu Chen, Bineng Zhong, Guorong Li, Shengping Zhang, and Rongrong Ji. Siamese box adaptive network for visual tracking. In *CVPR*, 2020. 2
- [9] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *CVPR*, 2022. 2, 6
- [10] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Atom: Accurate tracking by overlap maximization. In *CVPR*, 2019. 2
- [11] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *CVPR*, 2020. 2
- [12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2
- [13] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*, 2019. 6
- [14] Vitaly Feng Qi and, Ablavsky, Qinxun Bai, and Stan Sclaroff. Siamese natural language tracker: Tracking by natural language descriptions with siamese trackers. In *CVPR*, 2021. 2, 6
- [15] Shenyan Gao, Chunlun Zhou, Chao Ma, Xinggang Wang, and Junsong Yuan. Aiatrack: Attention in attention for transformer visual tracking. In *ECCV*. Springer, 2022. 6
- [16] Mingzhe Guo, Zhipeng Zhang, Heng Fan, and Liping Jing. Divert more attention to vision-language tracking. In *NeurIPS*, 2022. 2, 3, 6, 7
- [17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 5
- [18] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE TPAMI*, 2019. 2, 6
- [19] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *CVPR*, 2018. 1, 2, 6
- [20] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *CVPR*, 2018. 2
- [21] Zhenyang Li, Ran Tao, Efstratios Gavves, Cees GM Snoek, and Arnold WM Smeulders. Tracking by natural language specification. In *CVPR*, 2017. 2
- [22] Zhenyang Li, Ran Tao, Efstratios Gavves, Cees G. M. Snoek, and Arnold W.M. Smeulders. Tracking by natural language specification. In *CVPR*, 2017. 7
- [23] Liting Lin, Heng Fan, Zhipeng Zhang, Yong Xu, and Haibin Ling. Swintrack: A simple and strong baseline for transformer tracking. In *NeurIPS*, 2022. 1, 2, 6
- [24] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 5
- [25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 4, 6
- [26] Ding Ma and Xiangqian Wu. Capsule-based object tracking with natural language specification. In *ACM MM*, 2021. 2, 3
- [27] Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *CVPR*, 2022. 2
- [28] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc Van Gool. Learning target candidate association to keep track of what not to track. In *ICCV*, 2021. 6
- [29] Matthias Muller, Adel Bibi, Silvio Giancola, Salman Alsubaihi, and Bernard Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *ECCV*, 2018. 6
- [30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 3, 4, 6

- [31] Mr D Murahari Reddy, Mr Sk Masthan Basha, Mr M Chinnaiahgari Hari, and Mr N Penchalaiah. Dall-e: Creating images from text. *UGC Care Group I Journal*, 8(14):71–75, 2021. [3](#)
- [32] Hamid Rezaatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. [5](#)
- [33] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *NeurIPS*, 2021. [3](#)
- [34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. [2](#), [5](#)
- [35] Paul Voigtlaender, Jonathon Luiten, Philip H.S. Torr, and Bastian Leibe. Siam r-cnn: Visual tracking by re-detection. In *CVPR*, 2020. [2](#)
- [36] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *CVPR*, 2021. [1](#), [2](#)
- [37] Xiao Wang, Xiujun Shu, Zhipeng Zhang, Bo Jiang, Yaowei Wang, Yonghong Tian, and Feng Wu. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In *CVPR*, 2021. [2](#), [6](#)
- [38] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *CVPR*, 2013. [7](#)
- [39] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *ICCV*, 2021. [1](#), [2](#), [4](#), [6](#)
- [40] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *ECCV*. Springer, 2022. [1](#), [2](#), [4](#), [5](#), [6](#)
- [41] Zhipeng Zhang, Houwen Peng, Jianlong Fu, Bing Li, and Weiming Hu. Ocean: Object-aware anchor-free tracking. In *ECCV*, 2020. [2](#), [6](#)
- [42] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking. In *ECCV*, 2018. [2](#)