Prompting for Multi-Modal Tracking

Jinyu Yang
Southern University of Science and
Technology
University of Birmingham
jinyu.yang96@outlook.com

Zhe Li
Southern University of Science and
Technology
liz8@mail.sustech.edu.cn

Feng Zheng*
CSE & RITAS, Southern University of
Science and Technology
f.zheng@ieee.org

Aleš Leonardis
University of Birmingham
a.leonardis@cs.bham.ac.uk

Jingkuan Song
University of Electronic Science and
Technology of China
jingkuan.song@gmail.com

ABSTRACT

Multi-modal tracking gains attention due to its ability to be more accurate and robust in complex scenarios compared to traditional RGB-based tracking. Its key lies in how to fuse multi-modal data and reduce the gap between modalities. However, multi-modal tracking still severely suffers from data deficiency, thus resulting in the insufficient learning of fusion modules. Instead of building such a fusion module, in this paper, we provide a new perspective on multi-modal tracking by attaching importance to the multi-modal visual prompts. We design a novel multi-modal prompt tracker (ProTrack), which can transfer the multi-modal inputs to a single modality by the prompt paradigm. By best employing the tracking ability of pre-trained RGB trackers learning at scale, our ProTrack can achieve high-performance multi-modal tracking by only altering the inputs, even without any extra training on multi-modal data. Extensive experiments on 5 benchmark datasets demonstrate the effectiveness of the proposed ProTrack.

CCS CONCEPTS

• **Computing methodologies → Tracking**.

KEYWORDS

Prompt learning, multi-modal tracking, multi-modal data

---

*Corresponding author.

Table 1: Dataset comparison between RGB tracking datasets and multi-modal ones. "M" denotes million. "Resolution" indicates the maximum resolution.

Dataset	Year	Modality	Videos	Frames	Resolution
LaSOT[10]	2019	RGB	1,400	3.52M	1280 × 720
GOT-10k[14]	2019	RGB	10,000	1.5M	1920 × 1080
TrackingNet[30]	2018	RGB	31,000	14M	1280 × 720
DepthTrack[37]	2021	RGB+D	200	294,600	640 × 480
VisEvent[34]	2021	RGB+E	820	371,127	346 × 260
LasHeR[23]	2021	RGB+T	1,224	734,800	960 × 576

1 INTRODUCTION

Object tracking is to localize an arbitrary object in a video sequence given the object description in the first frame. Recent years witness a great spur of object tracking research, especially in color mode. Large-scale datasets [10, 13, 30], worldwide challenges [18–20], and flourishing novel algorithms [2, 6, 9, 29] boost this area. At the same time, multi-modal tracking, *e.g.*, Visible+Depth (RGB-D), Visible+Thermal (RGB-T), and Visible+Event (RGB-E), opens up opportunities for more accurate and robust tracking in complex scenarios. For example, depth cameras provide spatial information to distinguish the object from the background, thermal cameras have a strong penetration ability to particulate matters like smog and fog, and event cameras are free from motion blurs and low illumination. Therefore, in complex scenarios, especially when trackers fail in color views, multi-modal information can strongly complementary.

However, unlike the success in color-based object tracking, we note that there exists a significant development gap in multi-modal tracking. High-performing data-driven models are lacking for multi-modal tracking, due to the deficiency of large-scale training datasets. For example, a comparison among RGB tracking datasets and multi-modal tracking ones is shown in Table 1, clearly showing that the multi-modal tracking datasets are orders of magnitude smaller than the RGB counterparts. Also, the resolution of multi-modal data is relatively smaller than the RGB ones. In addition, multi-modal datasets also suffer from low-quality problems like desynchronization and misalignment between different modalities or sensors.

To handle the data-hungry bottleneck, compared to manually collecting the multi-modal data for deep model designing, a more efficient way is to best exploit the state-of-the-art pre-trained RGB
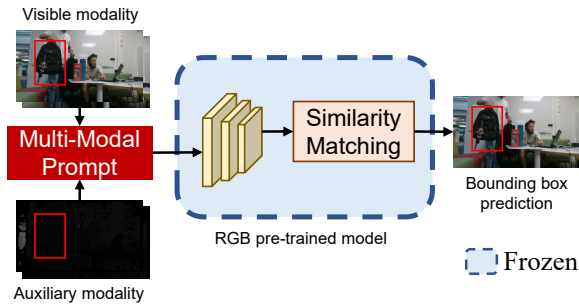
Figure 1: How our multi-modal prompt works. Given a frozen, pre-trained RGB tracker, we expect it can perform well on multi-modal tracking tasks with only a modality-agnostic prompt on the test videos.

tracking models. Current state-of-the-arts in multi-modal tracking already have many attempts on it mostly with the "pre-trained RGB baselines + multi-modal data fine-tuned" paradigm. In RGB-D tracking, trackers use pre-trained RGB trackers as strong baselines and show promising performance [37]. In the VisEvent benchmark, Wang et al. [34] proposed a series of baselines by embedding event information into RGB trackers and re-trained on its training set. However, these models can only learn a small amount of new data with modality gaps, resulting in incomplete learning of the cross-modal fusion mechanism. Further, inevitably, such a modality gap will confuse models that were originally well-learned on large-scale RGB datasets. Even though we can have large-scale cross-modal datasets in the future, current research clearly illustrates that over-training on a new dataset also potentially causes a knowledge forgetting problem. Therefore, a question is raised: how can we effectively utilize both the large-scale RGB knowledge and the complementary information from non-color modalities?

To achieve high-performing multi-modal tracking more efficiently, we explore a different route in this paper. Firstly, we observe that there are visible and auxiliary modalities in multi-modal tracking. Tracking performance highly depends on the models' discriminative ability in color views, which can be gained from large-scale data training, while auxiliary modalities are effective in specific scenarios. Moreover, due to the data sources/sensors, the auxiliary modality is less informative. Inappropriately importing auxiliary modalities may receive more kicks than halfpence. Drawing inspiration from the recent advances in prompting in Natural Language Processing (NLP), we therefore ask, can we still use auxiliary modalities but without the inappropriate fine-tuning process? To this end, we propose a new simple but efficient prompt method for multi-modal object tracking tasks, namely multi-modal Prompt Tracker (ProTrack). Instead of altering or fine-tuning the pre-trained model itself, we directly modify multi-modal inputs to adapt the state-of-the-art RGB trackers as shown in Fig. 1. The comparison between our ProTrack and the existing three types of fusion strategies for multi-modal tracking methods is shown in Fig. 2. Regardless of the stage at which classical methods fuse multiple data, there are two main differences compared to our methods. On one hand, these models have to be fine-tuned on new datasets for new tasks, while our model can be merely trained once on RGB videos. On the other

hand, since these models are fine-tuned for only one type of data, they cannot be used in different multi-modal scenarios, while our one model can handle multiple tasks, simultaneously.

By selecting the appropriate prompts, we can manipulate the model behavior so that the pre-trained model itself can be used to predict the desired output, without any additional task-specific or modality-specific training. Unlike efforts on vision-and-language tasks with textual prompts, we introduce the tracking applications of this promising paradigm with multi-modal visual prompts. By transferring the multi-modal data to colored ones, the invisible auxiliary modality information can be leveraged by RGB trackers as color-based referential markers. We adopt the prompt engineering in 5 multi-modal object tracking benchmarks. Enormous experiments verify its effectiveness. Our contributions are three-fold:

- We present a novel prompt learning paradigm for multi-modal tracking, in which both the large-scale RGB knowledge and the complementary information are effectively utilized. To the best of our knowledge, this is the first attempt on prompt learning in multi-modal tracking areas.
- We present a principled solution to cross-modal prompt configurations for various kinds of multi-modal tracking but without the inappropriate fine-tuning process.
- We unify different multi-modal object tracking tasks into a prompting framework and conduct comprehensive experiments on different scenarios which demonstrate the effectiveness of ProTrack.

2 RELATED WORK

2.1 Multi-Modal Tracking

Tracking through joint information from different modalities is an intuitive idea, whose effectiveness has been verified by many existing works. Preliminary explorations focused on auxiliary modality representation to assist the trackers on specific occasions. For example, early methods use depth and thermal maps for occlusion handling [26] and scale estimation [21]. Similarly, event features are extracted for target re-detection after model drifting [5]. With the development of deep learning, data-driven models appeared, which devote to fusing different modalities with deep features. Wang *et al.* [34] construct a series of baseline trackers by fusing visible frames and event flows on RGB tracking baselines and re-train them on the proposed training dataset. Li *et al.* [7] concatenate deep features of visible and thermal frames and adaptively fuse them with the proposed FusionNet to get robust feature representations. Qian *et al.* [31] propose embedded depth-aware convolution in RGB tracking baselines to obtain robust target estimation. At the same time, a rising problem in those areas is the deficiency of large-scale training datasets. To deal with the data-hungry bottleneck, besides collecting new datasets, Zhang *et al.* [39] generate a large-scale synthetic RGB-T dataset from 8,335 videos, and Yan *et al.* [37] use synthetic RGB-D data generated from LaSOT [10] and COCO [24].

2.2 Prompt Learning

"Pre-train, prompt, and predict" paradigm is replacing the "pre-train, fine-tune" one in NLP area. In the prompt paradigm, downstream tasks are reformulated to look more like the solved ones with the
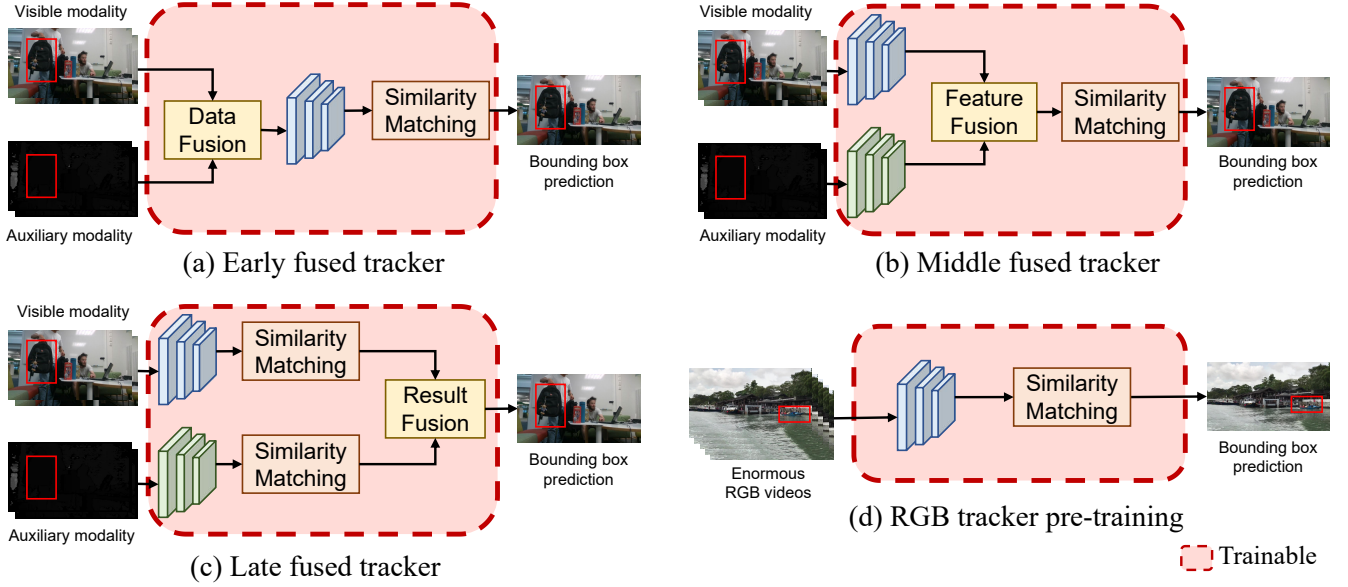
Figure 2: Comparison between our proposed method (d) and the existing ones (a, b, and c) during the training process.

help of textual prompts [25]. Instead of adapting pre-trained language models to downstream tasks via fine-tuning, prompting learning can reach high performance even in the few-shot or zero-shot settings. Extending from NLP, Pre-trained vision-language models also show promising capabilities in many tasks. CLIP [32] is trained from scratch on a dataset of 400 million (image, text) pairs collected from the internet, and evaluated on 30 different existing computer vision datasets. Despite CLIP, multiple works in various research fields have emerged. For example, Zhou *et al.* [41] proposed context optimization (CoOp) which learns continuous soft prompts that perform well for downstream tasks. CLIP-Adapter [11] is then proposed to conduct fine-tuning with feature adapters on both visual and language branches.

2.3 Prompting for Vision Tasks

There have been preliminary attempts on prompt with images in vision tasks. CPT [38] converts visual grounding into a fill-in-the-blank problem by creating visual prompts with color-based co-referential markers in both images and text. Very recently, Jia *et al.* [15] proposed VPT which applied visual prompts to vision backbones on 24 classification tasks. Bahng *et al.* [1] proposed visual prompting which learns a task-specific image perturbation to adapt the pre-trained models to downstream tasks. With only changing a few pixels, the visual prompting shows surprising effectiveness. To the best of our knowledge, there are no prompt paradigms designed for semantic-agnostic multi-modal vision tasks.

3 METHOD

3.1 Prompt Formulation

In the classical paradigm of NLP, in general, models are first pre-trained on large-scale datasets and then fine-tuned on downstream data to adapt to new tasks. The data and supervision are often

Table 2: Terminology and notation of our proposed multi-modal prompting methods.

Name	Notation	Description
Input	$X = \{V, A\}$	Multi-modal frames.
Prompt function	$f(X)$	A function that converts the input into a specific form.
Prompt	$X' = f(X)$	Prompted RGB frames generated by prompt function.
Model	$tracker\{X'\}$	A pre-trained model for object tracking with input X'.
Output	B	Predicted target bounding box.

different between the two stages, thus leading to forgetting or under-fitting problems. Instead of updating the model, a new prompt paradigm is proposed, using text templates to modify the data to fit the input of the pre-trained model. The reason to fix these models is that they have learned at scale, seeing more concepts that cannot be provided by downstream datasets. The key component of prompting is how to design the templates (textual prompts), which can reduce the distribution gap between the pre-trained data and downstream data.

3.2 Intuition

Similarly, the key to our multi-modal prompts is to reformulate multi-modal tracking into a single-modal tracking problem. To this end, our ProTrack establishes fine-grained connections between single-modal videos and multi-modal videos. Table 2 shows how we define the prompting method in multi-modal tracking. The multi-modal input $X = \{V, A\}$ is converted to $X' \in \mathcal{R}^{3 \times H \times W}$ after the prompt function $f(X)$, where H and W are the height and width

of the input image, respectively. Here we denote visible modality as V and auxiliary modality as A.

In ProTrack, the pre-trained model is from the RGB tracking area while the downstream tasks are multi-modal object tracking tasks. Specifically, the ProTrack framework consists of two components: 1) a multi-modal prompt that transfers the multi-modal video sequences into visible single-modal ones; 2) a pre-trained model $tracker\{X\}$ that has a strong discriminative ability in the visible tracking area.

We then illuminate how we generate multi-modal prompts and reformulate the tracking problems.

3.3 Multi-Modal Prompt Design

Given that the prompt specifies the task, choosing a proper prompt has a large effect on not only the accuracy but also which task the model performs in the first place.

Pre-trained tracker. In this work, given a multi-modal video in multi-modal tracking, we usually have the Visible Modality (RGB) and Auxiliary Modality (*e.g.*, depth, event, or thermal). The original multi-modal tracking process can be formulated as:

$$tracker : \{X_t, X_1, B_1\} \rightarrow B_t \tag{1}$$

where B_t denotes the predicted bounding box and B_1 is the bounding box supervision in the first frame. Thus, as shown in Fig. 2 (a,b,c), the model $tracker$ will be trained using multi-modal data $X = \{V, A\}$.

Instead, as shown in Fig. 2(d), our model $tracker$ is trained merely using existing large-scale RGB data to associate the objects in different frames. Thus, the input size of the tracker is fixed to $3 \times H \times W$. Therefore, no fusion module is required, neither in the training stage nor in the testing stage. We consider a variant of the spatial-temporal transformer [35] as the baseline tracker. For more details, please refer to Sec. 4.2. The intuitive consideration is that RGB videos are more readily available than collecting multi-model datasets, so we have already had more RGB videos. The immediate benefit of using an RGB tracker is that the tracker has seen many challenges in learning from big data. If we can transfer other challenges to these already-seen challenges by introducing new modality data, the association performance will be improved further.

Prompt. Obviously, if we want to use the pre-trained models without any updates, we need to transfer the multi-modal data $X = (V, A)$ into the solution $3 \times H \times W$. With only modifying the trackers' input, our multi-modal prompt function can be formulated as:

$$f(V_t, A_t) = \lambda * Color(A_t) + (1 - \lambda) * Color(V_t), \tag{2}$$

where λ is a parameter. $Color(*)$ denotes dyeing function of different modalities. If the data V has one channel, the result $Color(V)$ will have three channels. But if the data V is the RGB image, the operation $Color(V)$ does nothing. Equipped with multi-modal prompts, it is then straightforward to apply pre-trained RGB trackers on prompted videos, without any extra training:

$$tracker : \{f(V_t, A_t), f(V_1, A_1), B_1\} \rightarrow B_t. \tag{3}$$

It is worth noting that the pre-trained RGB tracker has not seen any multi-modal data.
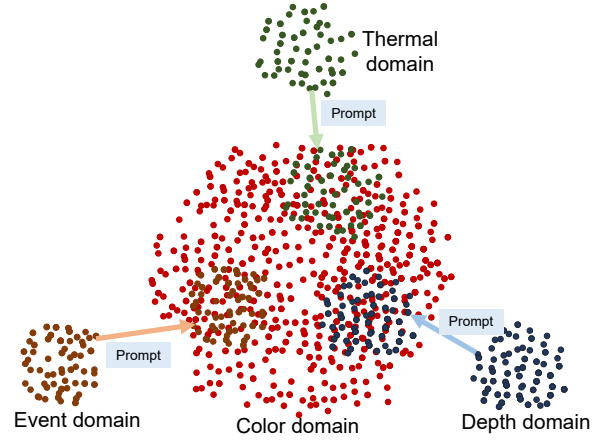
Figure 3: Multi-modal prompts in the data domain reduce the distribution gap between the pre-trained data (red) and downstream data.

3.4 Why Multi-Modal Prompt Works?

Compared with a special design to read and understand different channels, our solution is more straightforward. By dyeing salient colors or highlighting the auxiliary modality on brightness, the prompting in our ProTrack can reach state-of-the-art performance. In summary, the following three aspects clearly explain why the prompting strategy is very effective for such types of tasks.

Color image is more informative. Conventional cameras can capture informative images and videos. Multi-modal information is proposed for vision tasks at first due to their sensitivity to specific activities. At the same time, the information provided by the sensors is more focused and less informative. In existing works, researchers expect the visible modality and auxiliary modality from two sensors can cooperate with each other to achieve more reliable object tracking. However, inappropriate fusion or combination between different modalities may damage the information volume, which finally has side effects on tracking performance. The key to exploiting multi-modal information is preserving the visible information at most and embedding the useful parts of the auxiliary information at the same time. Thus, the prompt paradigm then provides us a new perspective to adapt the tasks to the pre-trained models, in which our design will not damage the discriminative ability of the original frameworks. In other words, to make multi-modal prompt work, we have to guarantee the parameter λ is relatively very small.

Prompting reduces the gap between distributions. Obviously, the data distribution of auxiliary modality is very different from that of the counterpart RGB videos. Thus, naturally, feeding this data directly into a model that learns only from RGB videos fails to achieve good performance. As shown in Fig. 3, the purpose of prompting is to transfer the data field of auxiliary modality to the main field of RGB images. The newly generated samples capture the inherent characteristics of both modalities, and since they lie within the distribution region learned by the model, the model can make better predictions. The essential benefit is that the distribution gap between the different modalities has narrowed considerably.
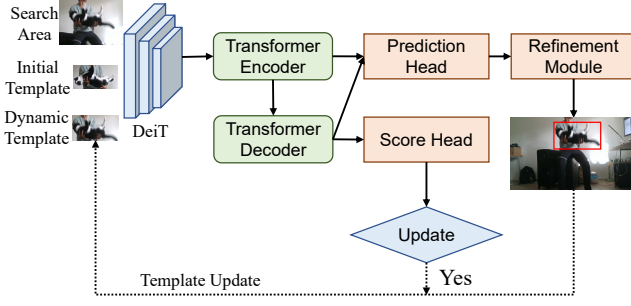
Figure 4: Architecture of the pre-trained model STARK.

Large-scale data plays a vital role. The original purpose of learning new fused data is to enhance the ability to meet new challenges. However, due to limited data volumes, not only this goal is not achieved, but the fusion module is not fully learned. Furthermore, fine-tuning somewhat destroys the associating ability learned from RGB videos. That's why the main problem we have is that the current fusion models cannot work well or even match the performance of using only RGB trackers. In contrast, our prompting strategy can successfully avoid such a paradox. On the one hand, auxiliary data is used, and thus new challenges can be solved. On the other hand, trackers pre-trained on large-scale data remain unchanged, thus avoiding the knowledge forgetting problem. Thus, we believe our prompting is a very good compromise under the condition that currently we have no large-scale multi-modal data. Nevertheless, in the future, we also believe that collecting more multi-modal data can better address this issue.

4 EXPERIMENTS

We evaluate the proposed ProTrack for a wide range of downstream multi-modal tracking tasks with pre-trained RGB trackers. We first describe our experimental settings in Sec. 4.1, including the downstream tasks and parameter settings. Then a brief introduction to the pre-trained model is given in Sec. 4.2. Then we demonstrate the effectiveness of our method on the downstream multi-modal tracking tasks in Sec. 4.3. We also systematically study how different design choices would affect performance in Sec. 4.4, which leads to an improved understanding of our approach.

4.1 Experimental Settings

To verify the effectiveness of our proposed ProTrack, we select the following downstream tracking tasks:

- **RGB-Depth object tracking.** We compare trackers on the popular CDTB [28] and DepthTrack [37].
- **RGB-Thermal object tracking.** We compare trackers on the large-scale LasHeR [23] and RGBT234 [22].
- **RGB-Event object tracking.** We compare trackers on the largest VisEvent [34].

In experiments, we empirically set $\lambda = 0.05$. For color choices, we use JET colormaps for depth maps and thermal images by default, while for event data, we simply use the event images transformed from event flows. All experiments are run on a single 32GB Tesla V100 GPU.

4.2 Pre-trained Model Selection

Since we aim to fully employ the discriminative ability of RGB pre-trained models, the choice of pre-trained models is of vital importance. Here we choose a variant of spatial-temporal transformer [35], which leads the leaderboard of RGB object tracking benchmarks. In this paper, we specifically use "STARK" denoting the variant model we selected.

Here we briefly introduce the pre-trained model we selected. STARK is based on spatial-temporal transformer architecture. As shown in Fig. 4, STARK contains the following main components: backbone, encoder, decoder, prediction head, and score head. For feature extraction, STARK utilizes DeiT [33] to strengthen the deep features. The encoder-decoder is based on DETR [4]. With inputting both the initial template and dynamic template, the encoder extracts the spatial-temporal features by modeling the correlation in both spatial and temporal dimensions. The decoder takes a single target query to predict a bounding box. The prediction head firstly takes the search region features from the encoder's output, then computes the similarity between the search region features and the output embedding from the decoder. The similarity is used to enhance the search area features and then estimate the probability distribution of the box corners. With a refinement module based on AlphaRefine [36], we finally get the bounding box prediction. The output is also used as a dynamic template to enhance the target template in most cases. Since there are mostly long-term settings in multi-modal tracking tasks, which means the targets may disappear and reappear during the tracking process, STARK utilizes a score head to determine whether the dynamic template should be updated.

Specifically, STARK is trained in two stages. In the first stage, the whole network, except for the score head, is trained end-to-end. In the second stage, only the score head is optimized with BCE loss. The training data consists of the training sets from the aforementioned LaSOT [10], GOT-10K [13], COCO2017 [24], and TrackingNet [30]. Top performance on multiple tracking benchmarks demonstrates the discriminative ability of this model. For more details, please refer to [35].

4.3 Main Results

CDTB dataset [28] consists of 80 long-term RGB-D video sequences for evaluation, covering a wide range of challenges in RGB-D tracking. In the CDTB dataset, target objects disappear and reappear frequently. Tracking Precision (Pr) and Recall (Re) are computed under a series of confidence thresholds. F-score is obtained by $F = \frac{2Re \times Pr}{Re + Pr}$. As shown in Table 3, we compare our ProTrack with existing stare-of-the-art RGB-D trackers on it. As reported, with the same ResNet-50 backbones, our ProTrack outperforms DeT [37] by 10%. Besides, ProTrack reaches a new state-of-the-art F-score of 75.7%, which also surpasses the STARK without prompting.

DepthTrack dataset [37] is a large-scale long-term RGB-D tracking benchmark. We evaluate RGBD trackers on the Depth-Track test set, which contains 50 long RGB-D video sequences. DepthTrack dataset uses the same protocols as the CDTB dataset. Table 4 presents that our ProTrack surpasses all previous state-of-the-art trackers, obtaining a new state-of-the-art F-score of 57.8%. Also, without multi-modal prompts, the F-score of STARK is 0.550,

Table 3: Overall performance on the CDTB dataset [28].

Method	DS-KCF[3]	CA3DMS[26]	CSR_RGBD++[16]	OTR[17]	DAL[31]	TSDM[40]	DeT[37]	STARK[35]	**ProTrack**
Pr	0.036	0.271	0.187	0.336	0.662	0.578	0.674	0.740	**0.747**
Re	0.039	0.284	0.201	0.364	0.565	0.541	0.642	0.765	**0.767**
F-score	0.038	0.259	0.194	0.312	0.592	0.559	0.657	0.752	**0.757**

Table 4: Overall performance on DepthTrack test set [37].

Method	DS-KCF-shape[12]	CA3DMS[26]	CSR_RGBD++[16]	DAL[31]	TSDM[40]	DeT[37]	STARK[35]	**ProTrack**
Pr	0.023	0.212	0.113	0.478	0.393	0.560	0.558	**0.583**
Re	0.023	0.216	0.115	0.390	0.376	0.506	0.543	**0.573**
F-score	0.023	0.214	0.114	0.421	0.384	0.532	0.550	**0.578**
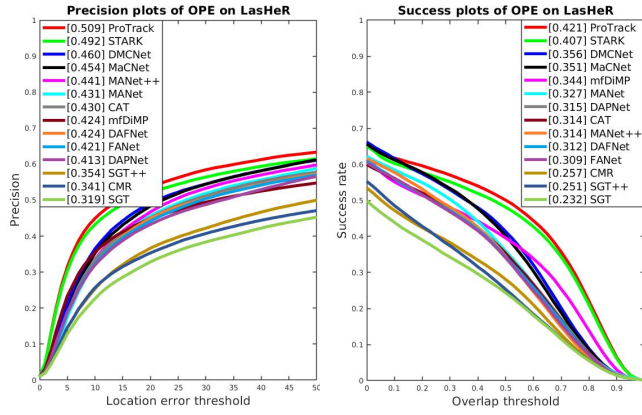
Figure 5: Overall performance on the LasHeR test set [23].

Figure 6: Overall performance on the RGBT234 dataset [22].
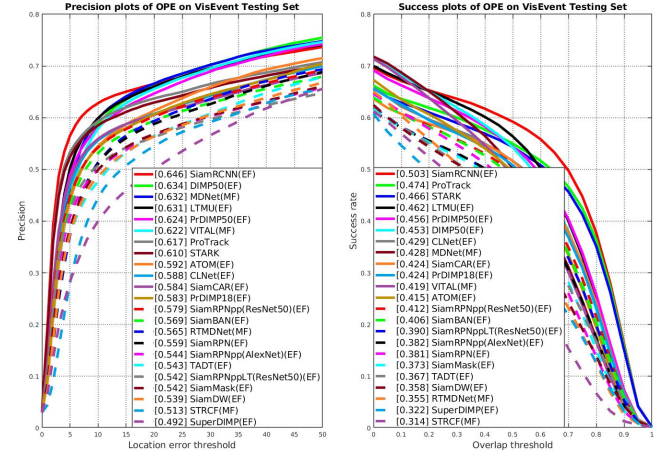
Figure 7: Overall performance on VisEvent test set [34].

while our ProTrack has an improvement of 2.8% thanks to the informative prompts.

LasHeR dataset [23] is a large-scale high-diversity benchmark for short-term RGB-T tracking. LasHeR equips the standard tracking performance metrics including Precision Plot and Success Plot. We evaluate trackers on the testing subset which contains 245 video sequences. Here we compare our ProTrack with existing stare-of-the-art RGB-T trackers. Note that the compared models are trained on the LasHeR training set. The results are reported in Fig. 5. As shown, our ProTrack shows the top performance of 50.9/41.9 (precision/success), outperforming the well-performing DMCNet [27] by 4.9%/6.3%.

RGBT234 dataset [22] is a large-scale RGB-T tracking dataset for performance evaluation, which contains 234 videos and 116.6k image pairs. Here we compare ProTrack with state-of-the-art RGB-T trackers. Comparison results are shown in Fig. 6. Without pre-training or adaptive learning on RGB-T data, our ProTrack can obtain a competitive precision rate of 78.6%. Moreover, ProTrack reaches the top success rate of 58.7%, which beats the well-designed RGB-T trackers. Compared to the STARK which is used for comparison, our ProTrack has improvements of 4.1% and 2.4% on precision and success rate, respectively. Till now, we observe that our Pro-Track shows better on the success plots compared to the precision

ones. Note that the precision plot requires manually setting thresholds for location error, which might be too tolerant for trackers to obtain higher precision. While the success plot reflects the real overlap ratio, which can be fairer than the precision one.

VisEvent dataset [34] is a large-scale Visible-Event benchmark. We evaluate trackers on the VisEvent test subset (320 videos). Two standard metrics are adopted for the evaluation of tracking performance, including Precision Plot and Success Plot. We involve the

multiple baseline methods [34] to compare, which keep state-of-the-art on VisEvent. Note that the compared trackers are dedicated to visible-event object tracking and fine-tuned on the VisEvent training subset. Our ProTrack still performs on par with current SOTAs. Except for without domain-specific training, another reason for our sub-optimal performance is that the event modality after prompting can only provide positive/negative (+1/−1) on some pixels, which is much less informative compared to the depth or thermal maps.

4.4 Ablation Study

We ablate different prompt design choices on the large-scale benchmarks for ablation study. For DepthTrack [37], we use F-score for comparison. For LasHeR [23] and VisEvent [34], we use precision rate and success rate.

Effect of different modalities. To investigate the contributions of different modalities, we evaluate the tracking performance with single-modal inputs. Results are reported in Table 5. "Default" denotes the input data we used with multi-modal prompts. Auxiliary information is transferred to colormaps for evaluation. As shown, the tracker performs better in visible modality compared to auxiliary modalities, and the performance is highly dependent on visible information. While they are not comparable to the ones with our default multi-modal prompts, which effectively work on multi-modal data.

Effect of color choices. As we choose colormaps for representation, colors choices are the key components in tracking the performance of ProTrack. Specifically, we compare different color choices as shown in Table 6. By default, the colormaps follow the JET style for covering more colors in RGB domain. Since event cameras only capture the changing pixels, there are only red or blue pixels by default in RGB-Event experiments. "RED" is the single-color colormap covering (0,0,0) to (255,0,0). "GRAY" indicates that we only use the grayscale maps. As reported, the default settings can achieve better performance compared to the single colored ones. It indicates that more colors bring more information and are more similar to the natural RGB image color distribution.

Effect of parameter λ. In practice, the hyperparameter λ, which indicates the trade-off between different modalities, is crucial in ProTrack. To investigate the effect of λ, we evaluate ProTrack with different λ, as shown in Table. 7. When $\lambda = 0$, there are no prompts and the input remains the visible modality only. As reported, the performance of ProTrack increases first and then decreases with λ improves on DepthTrack and VisEvent. This can be explained by the that, a tiny λ can preserve strong discriminative ability from visual appearances, but will undermine the visibility of the auxiliary modality, and vice versa. In addition, we observe that the performance on LasHeR keeps going higher with λ going larger, indicating that the thermal maps are more informative or more color-like, and thus they can provide more complementary compared to depth and event information. Thus, choosing a proper λ is essential in our ProTrack.

5 ANALYSIS AND DISCUSSION

In this section, we conduct a deep analysis of ProTrack for a better understanding of its working mechanism from various perspectives.

Table 5: Ablation study on different modalities.

	DepthTrack[37]	LasHeR[23]		VisEvent[34]	
Modality	F-score	Pre	Suc	Pre	Suc
Default	0.578	0.509	0.419	0.617	0.474
Visible	0.550	0.492	0.405	0.610	0.466
Auxiliary	0.297	0.349	0.289	0.411	0.277

Table 6: Ablation study on color choices.

	DepthTrack[37]	LasHeR[23]		VisEvent[34]	
Color	F-score	Pre	Suc	Pre	Suc
Default	0.578	0.509	0.419	0.617	0.474
RED	0.564	0.486	0.401	0.596	0.443
GRAY	0.561	0.501	0.410	0.546	0.396

Table 7: Ablation study on parameter λ. Bold denotes the highest score.

|	DepthTrack[37]	LasHeR[23]		VisEvent[34]	
λ	F-score	Pre	Suc	Pre	Suc
0	0.550	0.492	0.405	0.610	0.466
0.01	0.557	0.495	0.407	0.612	0.469
0.05(Default)	**0.578**	0.509	0.419	**0.617**	**0.474**
0.1	0.537	0.527	0.436	0.611	0.468
0.2	0.499	**0.531**	**0.439**	0.606	0.454

5.1 Visualization

To investigate how the multi-modal prompts work during the tracking process, we visualize the search regions and corresponding feature maps before and after prompting in two representative tracking frames, as shown in Fig. 8. As shown, we observe that our ProTrack has much cleaner and unambiguous target localization than the one without prompting and leads to an accurate bounding box estimation. While the one without prompts often produces multiple local maxima for distractors, our methods are able to almost fully suppress these. An important design enabling this is the prompts with auxiliary information. Although it is almost invisible to see the auxiliary information after prompting, the feature maps tend to distinguish the objects from background distractors due to the change in overall data distribution. The visualized examples verify the hypothesis shown in Fig. 3. Our multi-modal prompts can transfer the invisible auxiliary modality into the visible one by transferring the auxiliary data into the color domain.

5.2 ProTrack with More Pre-trained Models

As ProTrack aims to leverage the information learned from enormous RGB data, many alternatives can be used as pre-trained models in our ProTrack framework. Here we choose some RGB trackers to simply verify the effectiveness of multi-modal prompts. The representative trackers include ATOM [8], TransT [6], DiMP [2], PrDiMP [9], and KeepTrack [29], which are pre-trained on large-scale RGB datasets, e.g., LaSOT [10], GOT-10K [13], COCO [24], and TrackingNet [30]. We report the results of trackers' F-score on DepthTrack without and with prompting, respectively. Compared

Table 8: ProTrack with more pre-trained models. Performance on DepthTrack [37] is shown according to F-score. "_P" denotes tracking performance after prompting.

ATOM[8]	ATOM_P	DiMP[2]	DiMP_P	PrDiMP[9]	PrDiMP_P	TransT[6]	TransT_P	KeepTrack[29]	KeepTrack_P
0.313	0.349	0.377	0.431	0.392	0.402	0.489	0.504	0.509	0.542
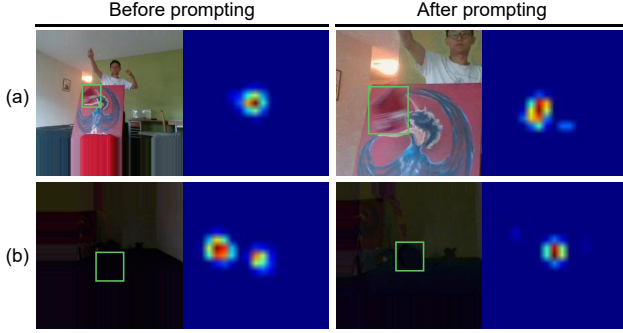
Figure 8: Visualized comparison of the score maps in search regions with/without prompting. The groundtruth bounding box is shown in green.

results are shown in Table 8. The RGB trackers show improved performance on multi-modal tracking after prompting. Specifically, DiMP and KeepTrack get improvements of 5.4% and 3.3% after prompting, respectively. Thus, ProTrack can continuously benefit from the strong discriminative ability of pre-trained models.

5.3 Beyond Dual-Modal tracking

As we claim that our ProTrack is modality-agnostic, its applications can be broader. For example, we can apply ProTrack to triple-modal tracking by modifying Eq. 2 to:

$$f(V, A_1, A_2) = \alpha * Color(A_1) + \beta * Color(A_2) + \gamma * Color(V), \quad (4)$$

where α, β, γ are parameters, satisfying $\alpha + \beta + \gamma = 1$. And, V, A_1, A_2 denote visible modality and two auxiliary modalities. With prompting, information from different modalities can be converged to the visible modality and get assistance from RGB pre-trained tracking models. Unfortunately, triple-modal tracking is not explored yet. We hope our work will provide a straightforward solution for this promising direction.

5.4 Failed Cases

Fig. 9 shows failed cases of our tracker. In particular, it shows the adjacent frames in two sequences containing the groundtruth in search regions and the corresponding score maps of the pre-trained tracker with or without prompting. We can see that the location of highest scores is not matching the groundtruth. Overall, our tracker typically fails due to two reasons: 1) The tracker's discriminative ability is limited to distinguishing the object in some challenging scenarios. 2) The input image is disturbed by the multi-modal prompts. For the former reason, complex sequences exist and challenge the trackers, as shown in Fig. 9(a). This can be solved by the improvements of pre-trained RGB trackers. For the latter, multi-modal prompts in ProTrack may wrongly guide the trackers as it
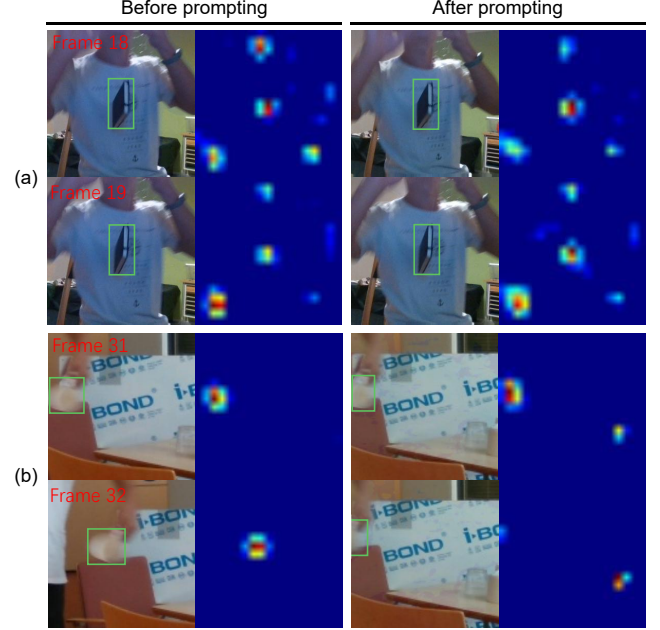
Figure 9: Visualization of failure cases of our tracker. The groundtruth bounding box is shown in green.

changes the data distribution. For this occasion when the tracking challenges can be solved by RGB trackers, the tracking performance after prompting may be disturbed by auxiliary information, as shown in Fig. 9(b). Thus, it will be necessary to design more robust prompts in the future, although our current approach can solve a considerable part of the problems.

6 CONCLUSION

In this paper, we propose multi-modal prompts for object tracking, a simple but effective approach to leverage large-scale RGB tracking models for a wide range of downstream multi-modal tracking tasks. Through applying prompting on multi-modal videos, we adapt the multi-modal tracking tasks to pre-trained RGB trackers. Thus, by solely modifying the trackers' input, we exploit the most of the discriminative ability from pre-trained RGB trackers to handle the multi-modal tracking challenges. Promising results on 6 benchmark datasets verify the effectiveness of our proposed ProTrack. We hope this work will spur further research on multi-modal tracking and provide inspiration to related areas.

ACKNOWLEDGMENTS

REFERENCES

[1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. 2022. Visual Prompting: Modifying Pixel Space to Adapt Pre-trained Models. *arXiv preprint arXiv:2203.17274* (2022).

[2] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. 2019. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE International Conference on Computer Vision*. 6182–6191.

[3] Massimo Camplani, Sion L Hannuna, Majid Mirmehdi, Dima Damen, Adeline Paiement, Lili Tao, and Tilo Burghardt. 2015. Real-time RGB-D Tracking with Depth Scaling Kernelised Correlation Filters and Occlusion Handling.. In *BMVC*, Vol. 4. 5.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.

[5] Haosheng Chen, Qiangqiang Wu, Yanjie Liang, Xinbo Gao, and Hanzi Wang. 2019. Asynchronous tracking-by-detection on adaptive time surfaces for event-based object tracking. In *Proceedings of the 27th ACM International Conference on Multimedia*. 473–481.

[6] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. 2021. Transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8126–8135.

[7] Chenglong, Xiaohao, Zhao, Nan, Cao, Xiaochun, Tang, and Jin. 2018. Fusing two-stream convolutional neural networks for RGB-T object tracking. *Neurocomputing* (2018).

[8] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. 2019. ATOM: Accurate Tracking by Overlap Maximization. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[9] Martin Danelljan, Luc Van Gool, and Radu Timofte. 2020. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7183–7192.

[10] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. 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*. 5374–5383.

[11] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. 2021. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544* (2021).

[12] Sion Hannuna, Massimo Camplani, Jake Hall, Majid Mirmehdi, Dima Damen, Tilo Burghardt, Adeline Paiement, and Lili Tao. 2016. DS-KCF: a real-time tracker for RGB-D data. *Journal of Real-Time Image Processing* 16, 5 (2016), 1–20.

[13] Lianghua Huang, Xin Zhao, and Kaiqi Huang. 2019. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019), 1–1. https://doi.org/10.1109/tpami.2019.2957464

[14] Lianghua Huang, Xin Zhao, and Kaiqi Huang. 2021. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 5 (2021), 1562–1577. https://doi.org/10.1109/TPAMI.2019.2957464

[15] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual Prompt Tuning. *arXiv preprint arXiv:2203.12119* (2022).

[16] Uğur Kart, Joni-Kristian Kämäräinen, and Jiří Matas. 2018. How to Make an RGBD Tracker?. In *ECCVW*.

[17] Uğur Kart, Alan Lukežič, Matej Kristan, Joni-Kristian Kämäräinen, and Jiří Matas. 2019. Object Tracking by Reconstruction with View-Specific Discriminative Correlation Filters. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[18] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Luka Čehovin Zajc, Martin Danelljan, Alan Lukezic, Ondrej Drbohlav, Linbo He, Yushan Zhang, Song Yan, Jinyu Yang, Gustavo Fernandez, and et al. 2020. The Eighth Visual Object Tracking VOT2020 Challenge Results.

[19] Matej Kristan, Jiri Matas, Ales Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Luka Cehovin Zajc, Ondrej Drbohlav, Alan Lukezic, Amanda Berg, et al. 2019. The seventh visual object tracking vot2019 challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 0–0.

[20] Matej Kristan, Jiri Matas, Ales Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Hyung Jin Chang, Martin Danelljan, Luka Cehovin, Alan Lukezic, et al. 2021. The Ninth Visual Object Tracking VOT2021 Challenge Results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2711–2738.

[21] Jiaxu Leng and Ying Liu. 2018. Real-time RGB-D Visual Tracking with Scale Estimation and Occlusion Handling. *IEEE Access* (2018), 1–1.

[22] Chenglong Li, Xinyan Liang, Yijuan Lu, Nan Zhao, and Jin Tang. 2019. RGB-T object tracking: Benchmark and baseline. *Pattern Recognition* 96 (2019), 106977.

[23] C. Li, W. Xue, Y. Jia, Z. Qu, B. Luo, and J. Tang. 2021. LasHeR: A Large-scale High-diversity Benchmark for RGBT Tracking. (2021).

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[25] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586* (2021).

[26] Ye Liu, Xiao-Yuan Jing, Jianhui Nie, Hao Gao, Jun Liu, and Guo-Ping Jiang. 2018. Context-aware three-dimensional mean-shift with occlusion handling for robust object tracking in RGB-D videos. *IEEE Transactions on Multimedia* 21, 3 (2018), 664–677.

[27] Andong Lu, Cun Qian, Chenglong Li, Jin Tang, and Liang Wang. 2022. Duality-gated mutual condition network for RGBT tracking. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[28] Alan Lukezic, Ugur Kart, Jani Kapyla, Ahmed Durmush, Joni-Kristian Kamarainen, Jiri Matas, and Matej Kristan. 2019. Cdtb: A color and depth visual object tracking dataset and benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10013–10022.

[29] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc Van Gool. 2021. Learning target candidate association to keep track of what not to track. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13444–13454.

[30] Matthias Müller, Adel Bibi, Silvio Giancola, Salman Al-Subaihi, and Bernard Ghanem. 2018. TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild. In *European Conference on Computer Vision*.

[31] Yanlin Qian, Song Yan, Alan Lukežič, Matej Kristan, Joni-Kristian Kämäräinen, and Jiří Matas. 2020. DAL : A Deep Depth-aware Long-term Tracker. In *International Conference on Pattern Recognition (ICPR)*.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*. PMLR, 8748–8763.

[33] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2020. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877* (2020).

[34] Xiao Wang, Jianing Li, Lin Zhu, Zhipeng Zhang, Zhe Chen, Xin Li, Yaowei Wang, Yonghong Tian, and Feng Wu. 2021. VisEvent: Reliable Object Tracking via Collaboration of Frame and Event Flows. *arXiv:2108.05015* (2021).

[35] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. 2021. Learning spatio-temporal transformer for visual tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10448–10457.

[36] Bin Yan, Xinyu Zhang, Dong Wang, Huchuan Lu, and Xiaoyun Yang. 2020. Alpha-Refine: Boosting Tracking Performance by Precise Bounding Box Estimation. *arXiv preprint arXiv:2012.06815* (2020).

[37] Song Yan, Jinyu Yang, Jani Kapyla, Feng Zheng, Ales Leonardis, and Joni-Kristian Kamarainen. 2021. DepthTrack: Unveiling the Power of RGBD Tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10725–10733.

[38] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2021. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797* (2021).

[39] Lichao Zhang, Martin Danelljan, Abel Gonzalez-Garcia, Joost van de Weijer, and Fahad Shahbaz Khan. 2019. Multi-modal fusion for end-to-end rgb-t tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 0–0.

[40] Pengyao Zhao, Quanli Liu, Wei Wang, and Qiang Guo. 2021. TSDM: Tracking by SiamRPN++ with a Depth-refiner and a Mask-generator. In *2020 25th International Conference on Pattern Recognition*. IEEE, 670–676.

[41] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2021. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134* (2021).