DPTracker: DYNAMIC PROMPTER FOR RGB-D Tracking

Junzhe Zhao[a], Jintao Su[a], Ye Liu[b,*], Jun Liu[c] and Miaohui Wang[d]

[a]School of Automation, Nanjing University of Posts and Telecommunications, Nanjing, P.R. China
[b]School of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Nanjing, P.R. China
[c]School of Computing and Communications, Lancaster University, Lancaster, UK
[d]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen University, Shenzhen, P.R. China

ARTICLE INFO

Keywords:
RGB-D tracking
Visual prompt learning
Dynamic prompting
Modality effectiveness prediction

ABSTRACT

RGB-D object tracking aims to accurately localize a target across video frames by jointly leveraging RGB and depth modalities. However, existing RGB-D trackers often suffer from limited adaptability to varying scene conditions and inconsistent reliability of modalities. To address these challenges, we propose DPTracker (Dynamic Prompt Tracker), a novel RGB-D tracking framework that leverages visual prompt learning and dynamic modality prompting. Specifically, DPTracker introduces a Modality Effectiveness Predictor (MEP) to estimate the validity of each modality, and a Dynamic Prompter (DP) that adaptively adjusts the fusion intensity of RGB and depth information during prompting. By dynamically reweighting modal contributions, DPTracker effectively suppresses low-quality depth cues while enhancing reliable RGB information. Built upon a frozen pre-trained RGB tracker, the proposed framework only fine-tunes lightweight prompt-related parameters, substantially reducing computational cost and data requirements. Extensive experiments on three benchmark datasets, including DepthTrack, CDTB, and VOT-RGBD2022, demonstrate that DPTracker achieves state-of-the-art performance with superior robustness and generalization capability, thereby validating the effectiveness of dynamic modality prompting in RGB-D tracking. Our code is available at: https://github.com/neilzhao996/DPTracker.

1. Introduction

Visual object tracking is a fundamental task in computer vision that aims to continuously localize a target across video frames. While conventional RGB-based trackers have achieved remarkable progress with the advent of deep learning [1, 2, 3, 4], their performance still degrades under challenging conditions such as illumination changes, occlusions, and background clutter. To overcome these limitations, RGB-D tracking has emerged as a promising direction by incorporating complementary depth information, which provides valuable three-dimensional cues for robust target localization [5, 6, 7].

However, effectively integrating RGB and depth information remains a major challenge. In real-world scenarios, the reliability of depth sensors fluctuates due to lighting variations, reflective surfaces, or measurement noise, leading to inconsistent modality quality. Many existing RGB-D trackers employ static fusion strategies that assign fixed importance to each modality [8, 9], which often results in the propagation of unreliable depth features through the network. This lack of adaptability significantly limits their robustness and generalization capability, especially in dynamically changing environments.

With the increasing popularity of large-scale pre-trained models, parameter-efficient transfer learning has emerged an appealing alternative to full fine-tuning. Instead of updating all model parameters, visual prompt learning adapts frozen pre-trained networks by optimizing a small number of learnable prompt tokens [10, 11, 12], This approach not only reduces computational cost but also retains the general knowledge embedded in foundation models. Inspired by its success in both natural language processing and visual recognition, recent studies such as ViPT [9] have introduced visual prompt tuning to RGB-D tracking, achieving competitive performance. Nevertheless, these methods typically adopt fixed prompting strategies and ignore the varying effectiveness of different modalities, which restricts their adaptability to complex and dynamic scenes. RGB-D tracking faces limited training data, as multi-modal datasets are much smaller than RGB counterparts. Parameter-efficient transfer learning [13] thus motivates our prompt-tuning design.

Figure1 illustrates common training paradigms in RGB-D object tracking. As shown in Figure 1(a), a base model is trained on large-scale RGB datasets. Figure 1(b) shows transfer learning, where the pre-trained RGB model is fully fine-tuned for RGB-D tracking, which is effective but computationally costly and may cause knowledge forgetting. Figure 1(c) depicts visual prompt learning, which freezes the backbone and adapts to RGB-D tasks by tuning prompt parameters. Building on this, Figure 1(d) presents our DPTracker, which integrates RGB cues and dynamically adjusts prompt intensity based on modality validity, bridging the gap between pre-trained RGB and downstream RGB-D tracking.

To address the aforementioned challenges, we propose DPTracker (Dynamic Prompt Tracker), a novel RGB-D tracking framework that integrates visual prompt learning with dynamic modality prompting. Specifically, DPTracker introduces a Modality Effectiveness Predictor (MEP) to

*Corresponding author
1224056328@njupt.edu.cn (J. Zhao); 1022051501@njupt.edu.cn (J. Su); yeliu@njupt.edu.cn (Y. Liu); j.liu81@lancaster.ac.uk (J. Liu); mhwang@szu.edu.cn (M. Wang)
ORCID(s):

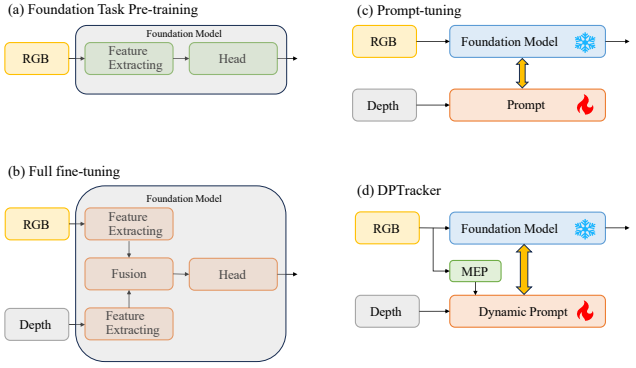

Figure 1: Training Methods of Existing RGB-D Object Tracking Algorithms.

estimate the reliability of each modality from RGB cues and a Dynamic Prompter (DP) to adaptively regulate the fusion intensity between RGB and depth features. By dynamically reweighting the contributions of different modalities, DPTracker effectively suppresses low-quality depth information while enhancing the utilization of reliable cues. Built upon a frozen pre-trained RGB tracker (OSTrack [2]), the proposed framework only fine-tunes lightweight prompt-related parameters, achieving an excellent balance between efficiency and accuracy.

The main contributions of this work are summarized as follows:

- We propose DPTracker, a parameter-efficient RGB-D tracking framework that combines visual prompt learning with dynamic modality prompting.
- We design a Dynamic Prompter (DP) guided by a Modality Effectiveness Predictor (MEP) to achieve adaptive fusion of RGB and depth information.
- Extensive evaluation is performed on several RGB-D datasets such as DepthTrack [5], VOT-RGBD2022 [14] and CDTB [6], demonstrating the effectiveness and superiority of our method.

2. Related work

RGB-D tracking extends traditional RGB object tracking by integrating depth information to improve robustness under challenging conditions such as occlusion, illumination changes, and background clutter. Early works such as CA3DMS [15] and OTR [16] leveraged handcrafted depth features or 3D representations to enhance target localization. Later, deep learning-based trackers like DeT [5], DMTracker [1], and DAL [17] incorporated CNN architectures to jointly learn multi-modal representations. However, these methods often rely on full-parameter fine-tuning, leading to high computational cost and limited generalization when training data are scarce.

The success of Transformer in visual recognition [18] has motivated their adoption in object tracking. OSTrack [2] unified feature extraction and relation modeling within a one-stream Transformer backbone, achieving competitive performance in RGB tracking. Subsequently, ViPT [9] introduced visual prompt tuning for RGB-D tracking, adapting pre-trained RGB models by optimizing prompt tokens rather than updating all parameters. Although efficient, fixed prompting strategies in these models cannot adapt to varying modality reliability, limiting their performance in complex dynamic scenes.

Prompt learning, initially popularized in NLP [11], has been extended to vision tasks as Visual Prompt Tuning (VPT) [10], enabling efficient adaptation of large-scale pre-trained models. In RGB-D tracking, prompting mechanisms have been used in ProTrack [8] to fuse modal cues via learnable prompts. Nonetheless, most existing visual prompting methods ignore the quality variability of input modalities, treating all sources equally. Such static strategies often propagate noise from unreliable modalities through the network.

Unlike prior works that employ fixed fusion weights or static prompts, DPTracker introduces dynamic modality prompting guided by the Modality Effectiveness Predictor (MEP). The MEP assesses the reliability of RGB and depth modalities at each frame, while the Dynamic Prompter (DP) adjusts their contributions accordingly. This design not only mitigates the negative impact of invalid modalities (e.g., depth corruption or missing data) but also enhances the adaptability of the model to diverse visual environments. The proposed framework thus bridges the gap between efficient prompt tuning and adaptive multimodal fusion, achieving strong performance with minimal additional parameters.

3. Method

3.1. Overall Framework of DPTracker

DPTracker (Dynamic Prompt Tracker) is built upon ViPT [9], a parameter-efficient framework that adapts a frozen pre-trained RGB backbone for RGB-D tracking without requiring full fine-tuning. To enhance this framework for RGB-D tracking, DPTracker introduces a Modality Effectiveness Predictor (MEP) to predict modality weights and replaces the first MCP with a Dynamic Prompter (DP). The overall network structure of DPTracker is illustrated in Figure 2. RGB and depth images first pass through an embedding layer to generate token sequences; additionally, RGB images are fed into the MEP to generate modality validity weights. The DP receives the two modality token sequences and the predicted validity weights to generate dynamically modulated input features. Subsequently, a stack of L-layer Transformer encoders is employed for feature extraction, with a Modality-Complementary Prompter (MCP) inserted between every two adjacent Transformer encoders. Finally, the output of the last Transformer encoder is passed to a prediction head to obtain the tracking results. Based on the frozen pre-trained model OSTrack [2], DPTracker dynamically adjusts the proportion of auxiliary modalities participating in prompting. With only a small number of additional parameters introduced by the MEP, DP, and

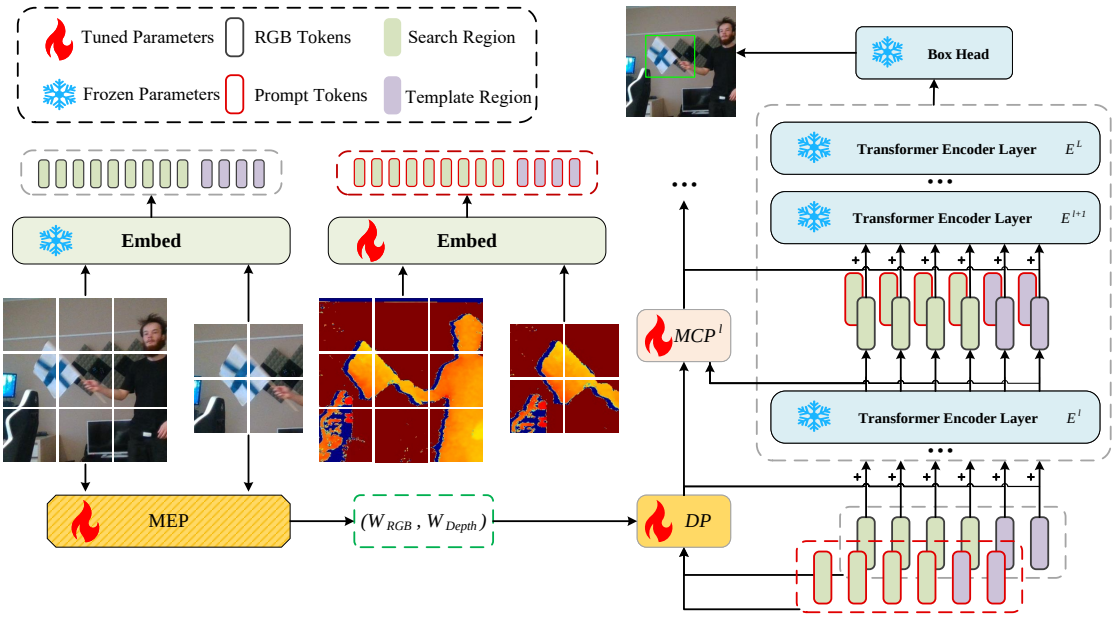

Figure 2: Overall Framework of DPTracker.

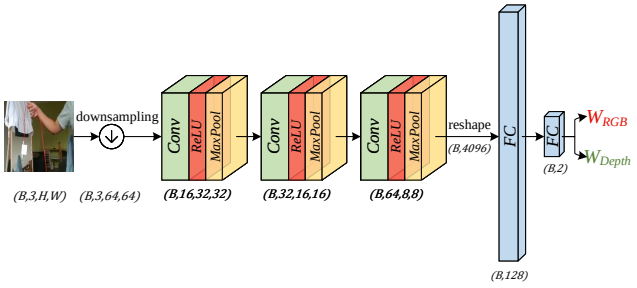

Figure 3: Network architecture of MEP.

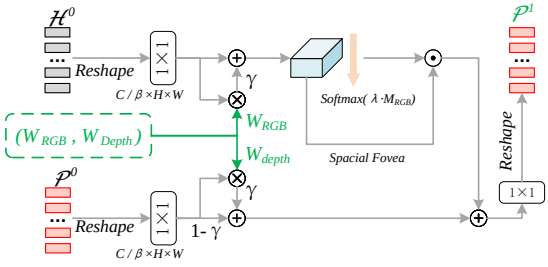

Figure 4: Network structure of DP.

MCP modules, it significantly improves the robustness and accuracy of the tracker.

3.2. Modality Effectiveness Predictor

In RGB-D tracking tasks, the relative contributions of RGB and depth information may vary under different scenes and conditions. To address this issue, the proposed Modality Effectiveness Predictor (MEP) dynamically estimates the effectiveness weights of different modalities.

As shown in Figure3, the MEP uses RGB images alone as input, since RGB is treated as the primary modality and provides richer semantic and appearance information. Many challenging factors that affect depth reliability, such as illumination changes, occlusions, and background clutter, are directly reflected in RGB appearance. Under such conditions, depth degradation is not random but highly correlated with observable RGB appearance changes (e.g., loss of contrast, disrupted object boundaries, and structural ambiguity), enabling the MEP to reliably infer depth effectiveness from RGB cues. Thus, the MEP estimates the effectiveness of the depth modality by modeling the correlation between RGB appearance patterns and cross-modality reliability. First, the RGB image is downsampled to reduce computational cost. Subsequently, three stacked blocks, each consisting of a 1×1 convolution, a ReLU activation, and a max-pooling layer, are used to extract compact appearance features. These features are then fed into two fully connected layers to produce a two-dimensional latent variable (h_1, h_2), where each element represents the unnormalized effectiveness score of the RGB and depth modalities, respectively. Specifically, (h_1, h_2) encodes the relative reliability of appearance cues for evaluating the contribution of RGB and depth modalities at the current frame.

The modality validity weights are obtained by normalizing (h_1, h_2) as:

$$\begin{aligned} W_{RGB} &= h_1/(h_1 + h_2), \\ W_{Depth} &= h_2/(h_1 + h_2). \end{aligned} \tag{1}$$

This normalization ensures that $W_{RGB} + W_{Depth} = 1$, thereby enforcing a relative weighting scheme between modalities and stabilizing modality contribution estimation under varying scene conditions.

3.3. Dynamic Prompter

In ViPT [9], feature fusion through the MCP module fails to account for modality-specific effectiveness. To fully and properly exploit modality validity weights, a Dynamic

Prompter (DP) is introduced to adaptively regulate the contribution of auxiliary modalities in the initial input. In multi-modality-related tasks, the quality and reliability of different modalities often vary dynamically with scene conditions. In the original ViPT framework, multiple cascaded MCPs are employed to generate prompt information at various levels; however, the validity of auxiliary modalities in the initial input is overlooked. When depth information is missing in the current frame, such a fixed fusion strategy leads to the accumulation of erroneous information across network layers, thereby deteriorating tracking performance. The proposed Dynamic Prompter dynamically adjusts the information fed into the initial Transformer encoder, reinforces the contribution of valid modalities, and suppresses that of invalid ones, thus effectively mitigating the adverse effects of low-quality modal data.

The detailed network structure of the DP module is illustrated in Figure 4. The initial RGB modal Token sequence and depth modal Token sequence are denoted as $\mathcal{H}^0$ and $\mathcal{P}^0$, respectively, and are projected to a low-dimensional space via 1×1 convolutional layers:

$$\mathcal{M}_{RGB} = g_1(\mathcal{H}^0), \quad \mathcal{M}_{Depth} = g_2(\mathcal{P}^0), \tag{2}$$

where $\mathcal{M}_{RGB}$ and $\mathcal{M}_{Depth}$ are projected to the dimension $\left[\frac{C}{\beta} \times H \times W\right]$. In DPTracker, the number of input channels C is set to 768, and the scaling factor β is configured as $\frac{768}{8}$. Here, g_1 and g_2 represent 1×1 convolutional layers. This projection reduces computational overhead while preserving the spatial structure required for subsequent modality-aware modulation.

Subsequently, modal validity weighting is applied to the RGB modal features, followed by the Spatial Fovea operation. The specific process can be expressed using Equations 3 and 4:

$$\mathcal{M}^e_{RGB} = \left(\mathcal{M}_{RGB} + \gamma \cdot W_{RGB} \cdot \mathcal{M}_{RGB}\right) \odot \mathcal{M}_{fovea}, \tag{3}$$

$$\mathcal{M}_{fovea} = \left\{ \lambda \cdot \frac{e^{\mathcal{M}^{[:,i,j]}_{RGB}}}{\sum e^{\mathcal{M}^{[:,i,j]}_{RGB}}} \right\}, \tag{4}$$

where $i = 1, 2, \ldots, H$ and $j = 1, 2, \ldots, W$. λ denotes a learnable parameter in the Dynamic Prompter, while γ is a fixed weight that controls the intensity of the modal validity effect.

Next, modal validity weighting is performed on the depth modal features $\mathcal{M}_{Depth}$:

$$\mathcal{M}^e_{Depth} = (1 - \gamma) \cdot \mathcal{M}_{Depth} + \gamma \cdot W_{Depth} \cdot \mathcal{M}_{Depth}. \tag{5}$$

Finally, the fused features obtained by summation are restored to the input dimension through a 1×1 convolutional layer:

$$\mathcal{P}^1 = g_3\left(\mathcal{M}^e_{RGB} + \mathcal{M}^e_{Depth}\right), \tag{6}$$

where g_3 represents a 1×1 convolutional layer. The dynamically prompted Token sequence $\mathcal{P}^1$ is then used as the input to the first Transformer encoder, enabling adaptive multi-modal fusion.

4. Experiment and Analysis

4.1. Datasets and Experiment Settings

Datasets. To assess the effectiveness of DPTracker, we perform experiments on three widely used RGB-D datasets, namely Depthtrack [5], VOT-RGBD2022 [14] and CDTB [6]. CDTB and DepthTrack comprise 80 and 50 long-term tracking sequences, respectively, featuring diverse challenges such as occlusion, darkness, and outdoor scenarios. VOT-RGBD2022 [14] is a widely adopted benchmark that contains 127 short-term RGB-D sequences for exploiting the role of depth in RGB-D tracking.

Evaluation Metrics. Both DepthTrack and CDTB are longterm RGB-D tracking datasets, thus the overall evaluation is based on the precision (Pr), recall (Re) and F-score metrics [19]. Tracking precision measures the accuracy of target localization when the target is marked as visible. Tracking recall assesses the accuracy of classifying visible marked targets. F-score, as the harmonic mean of precision and recall, reflects the overall performance of the tracker.

Implementation details. All experiments are conducted on a workstation equipped with a single NVIDIA GeForce RTX 3090 GPU and an Intel i9-10900K CPU, running Ubuntu 22.04. The experimental environment is configured with Python 3.7 and PyTorch 1.11. In addition, the toolkit provided by the Visual Object Tracking (VOT) challenge is employed to compute the performance metrics of the tracking results, where the specific versions used are `vot-toolkit 0.5.3` and `vot-trax 3.0.3`.

For model training, DPTracker leverages cue-word fine-tuning to effectively reduce its dependence on large-scale annotated datasets for downstream tasks. Therefore, only the DepthTrack dataset is used during training. DPTracker adopts OSTrack as its pre-trained base model, and all parameters of the base model remain frozen throughout the training process. The model is trained for 70 epochs, with the learning rate initialized at $4{\times}10^{-4}$ and decayed to $4{\times}10^{-5}$ after the 56th epoch. The batch size is set to 32, and the AdamW optimizer is employed with a weight decay of 0.0001. The fixed weight γ in the DP module is set to 0.05.

4.2. State-of-the-art Comparison

For RGB-D tracking, we evaluate the proposed DPTracker on three widely adopted benchmark datasets, namely DepthTrack [5], VOT-RGBD2022 [14], and CDTB [6], and compare its performance against a comprehensive set of state-of-the-art RGB-D tracking methods.To intuitively present the multi-metric performance comparison across different datasets, we plot a radar chart that integrates key evaluation indicators of DPTracker, ViPT, and OSTrack, as shown in Fig. 5.

Table 1 presents the comparative experimental results of DPTracker on the CDTB dataset. DPTracker achieves a

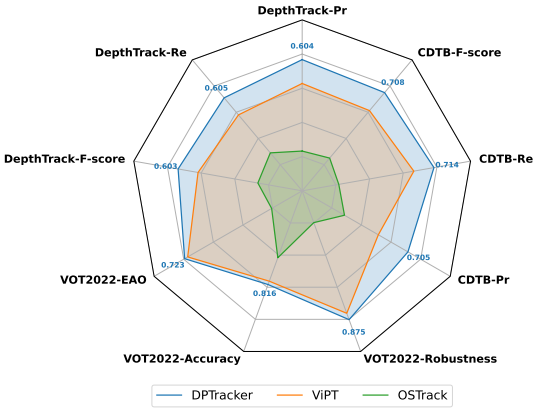

Figure 5: Multi-metric Performance Comparison of Tracking Algorithms Across Benchmarks.

Table 1
Comparative experimental results of DPTracker on the CDTB dataset.

Method	Pr↑	Re↑	F-score↑	Type	Year
DiMP50 [1]	0.546	0.549	0.547	RGB	2019
DAL [17]	0.661	0.565	0.592	RGB-D	2021
TSDM [20]	0.578	0.541	0.559	RGB-D	2021
DeT [5]	0.651	0.633	0.642	RGB-D	2021
DMTracker [21]	0.662	0.658	0.660	RGB-D	2022
ViPT [9]	0.691	0.703	0.697	RGB-D	2023
SSLTrack [22]	0.650	0.620	0.635	RGB-D	2024
FECD [23]	0.637	0.624	0.630	RGB-D	2024
CDAAT[24]	0.665	0.737	0.699	RGB-D	2025
OSTrack [2]	0.675	0.662	0.668	RGB	2022
DPTracker	0.705	0.714	0.708	RGB-D	

Table 2
Comparative experimental results of DPTracker on the VOT-RGBD2022 dataset.

Method	EAO↑	Acc↑	Rob↑	Type	Year
DRefine [25]	0.592	0.775	0.760	RGB-D	2021
STARK_RGBD [25]	0.647	0.803	0.798	RGB-D	2021
DeT [5]	0.657	0.760	0.845	RGB-D	2021
DMTracker [21]	0.658	0.758	0.851	RGB-D	2022
SBT_RGBD [14]	0.708	0.809	0.864	RGB-D	2022
ProTrack [8]	0.651	0.801	0.802	RGB-D	2022
SPT [26]	0.651	0.798	0.851	RGB-D	2023
ViPT [9]	0.721	0.815	0.871	RGB-D	2023
Un-Track [27]	0.721	0.820	0.869	RGB-D	2024
RDT-TEF [28]	0.710	0.807	0.877	RGB-D	2025
TABBTrack [29]	0.722	0.821	0.874	RGB-D	2025
OSTrack [2]	0.666	0.808	0.814	RGB	2022
DPTracker	0.723	0.816	0.875	RGB-D	

Precision of 0.705, a Recall of 0.714, and an F-score of 0.708. Compared with ViPT, DPTracker achieves improvements of 2.0%, 1.6%, and 1.6% in Precision, Recall, and F-score, respectively. This fully verifies that modal validity information plays an important role in different types of tracking models.

Table 2 lists the comparative experimental results of DPTracker on the VOT-RGBD2022 dataset. DPTracker achieves an EAO value of 0.723, a Accuracy (Acc) of 0.816, and a Robustness (Rob) of 0.875. Among these metrics, both the EAO value and Robustness reach leading

Table 3
Comparative experimental results of DPTracker on the DepthTrack dataset.

Method	Pr↑	Re↑	F-score↑	Type	Year
ATCAIS [30]	0.500	0.455	0.476	RGB-D	2020
DDiMP [30]	0.503	0.469	0.485	RGB-D	2020
DeT [5]	0.560	0.506	0.532	RGB-D	2021
ProTrack [8]	0.583	0.573	0.578	RGB-D	2022
SPT [26]	0.527	0.549	0.538	RGB-D	2023
ViPT [9]	0.592	0.596	0.594	RGB-D	2023
SSLTrack [22]	0.565	0.491	0.525	RGB-D	2024
RDT-TEF [28]	0.615	0.538	0.574	RGB-D	2025
M^3Track[31]	0.566	0.588	0.577	RGB-D	2025
CDAAT[24]	0.578	0.603	0.590	RGB-D	2025
OSTrack [2]	0.558	0.576	0.567	RGB	2022
DPTracker	0.604	0.605	0.603	RGB-D	

levels. When compared with ProTrack, a multi-modal tracker that also adopts the prompt concept, DPTracker demonstrates that learnable prompt modules can enable trackers to achieve better generalization performance. In contrast to ViPT, DPTracker based on dynamic prompting achieves improvements of 0.5% and 0.3% in Robustness and EAO value, respectively.

Table 3 presents the comparative experimental results of DPTracker on the DepthTrack test set. Specifically, DPTracker achieves a Precision of 0.604, a Recall of 0.605, and an F-score of 0.603. respectively. Compared with OSTrack, DPTracker improves the F-score by 6.3%. This indicates that DPTracker not only well inherits most of the prior knowledge from the RGB modality but also effectively leverages the supplementary information from the depth modality. When compared with ViPT, DPTracker achieves a 1.5% improvement in F-score, which verifies the effectiveness of the dynamic prompting approach based on modal validity.

4.3. Visualization

Figure 6 illustrates the visual tracking results of DPTracker on the CDTB dataset, which includes four RGB-D video sequences representing typical challenging scenarios such as dark scenes, fast motion, similar targets, and long-distance tracking.

In Sequence 1, under dark scenes, the RGB image quality deteriorates while depth information remains stable. The MEP module effectively predicts modality validity, guiding the DP module to dynamically adjust feature fusion. Consequently, DPTracker achieves notably superior performance compared with ViPT and other trackers.

Sequence 2 involves fast-moving targets where both RGB and depth images suffer from motion blur and partial occlusion. While most trackers fail, DPTracker continues to localize the target accurately in most frames.

In Sequence 3, multiple similar moving targets cause repeated occlusions. DPTracker successfully distinguishes the true target from distractors, maintaining accurate tracking.

Sequence 4 corresponds to a long-distance tracking scenario. When the target moves into the depth of the corridor, partial depth loss occurs due to sensor limitations.

Table 4
The impact of full-parameter fine-tuning and prompt tuning methods on tracking results.

		DepthTrack			CDTB		
Methods	Parameters	Pr↑	Re↑	F-score↑	Pr↑	Re↑	F-score↑
full fine-tuning	93,906,419	0.579	0.570	0.575	0.681	0.690	0.685
prompt-tuning	1,387,886	0.604	0.605	0.603	0.705	0.714	0.708

Table 5
The impact of MEP and DP on tracking results.

		DepthTrack			CDTB		
MEP	DP	Pr↑	Re↑	F-score↑	Pr↑	Re↑	F-score↑
-	-	0.567	0.559	0.563	0.688	0.694	0.691
✓	-	0.589	0.582	0.584	0.696	0.702	0.698
-	✓	0.574	0.572	0.573	0.696	0.705	0.701
✓	✓	0.604	0.605	0.603	0.705	0.714	0.708

Table 6
Influence of fixed weight γ in dynamic prompter on tracking results.

	DepthTrack			CDTB		
γ	Pr↑	Re↑	F-score↑	Pr↑	Re↑	F-score↑
0.00	0.677	0.588	0.582	0.691	0.699	0.696
0.01	0.592	0.597	0.595	0.699	0.706	0.701
0.05	0.604	0.605	0.603	0.705	0.714	0.708
0.10	0.579	0.591	0.584	0.701	0.708	0.705
0.50	0.559	0.569	0.565	0.684	0.690	0.689

Benefiting from its dynamic multi-modal fusion mechanism, DPTracker maintains stable tracking performance.

Overall, the dynamic prompting strategy based on modality validity enables adaptive fusion of depth cues across levels, thereby enhancing the robustness and flexibility of DPTracker in complex tracking environments.

4.4. Ablation Experiment Analysis

To verify the effectiveness of dynamic prompting based on modality validity, we conducted ablation studies on the MEP and DP modules in DPTracker. When removing MEP, a fixed weight of $(0.5, 0.5)$ was used to replace $(W_{\text{RGB}}, W_{\text{Depth}})$, assigning equal importance to both modalities. When removing DP, the MEP-predicted weights were directly applied to the RGB and depth tokens. As shown in Table 5, removing either module leads to a noticeable performance drop, confirming that both MEP and DP are essential to the framework.

DPTracker follows the visual prompt learning paradigm of ViPT, training only a small set of prompt parameters. We further compare full-parameter fine-tuning and prompt-based fine-tuning. As shown in Table 4, prompt tuning requires only 1.5% of the parameters of full fine-tuning yet achieves better results. Fully unfreezing the base model (about 94M parameters) even degrades performance, as global optimization overwrites the pre-trained RGB knowledge and increases the training burden. Hence, prompt-based fine-tuning enables efficient and effective adaptation to downstream tracking tasks.

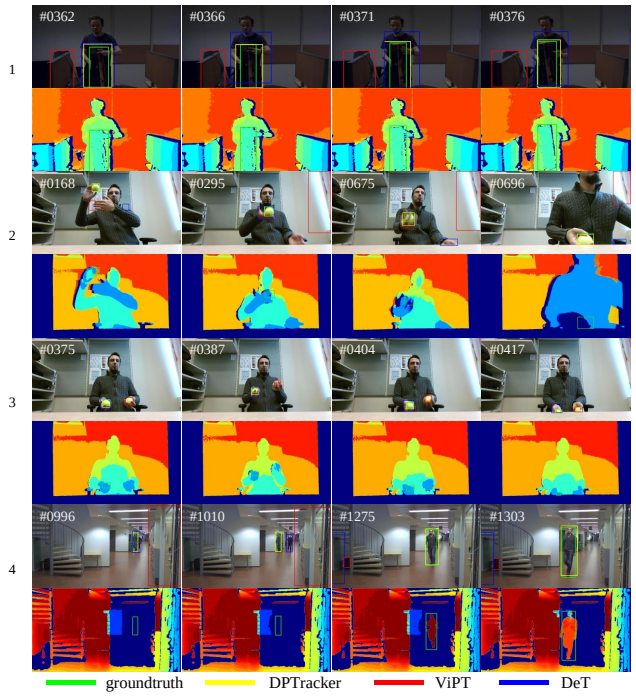

Figure 6: Visual Tracking Results of DPTracker on the CDTB Dataset.

Finally, to analyze the influence of the fixed weight γ in the DP module, five groups of γ values were tested (Table 6). When γ is too large, modality prediction errors are amplified; when too small, the model lacks adaptability. Experiments show that $\gamma = 0.05$ provides the best balance, maximizing the dynamic adjustment capability of modality effectiveness.

4.5. Computational Efficiency.

DPTracker runs at 32.5FPS on an NVIDIA RTX 3090 GPU with an inference-time memory footprint of 3.5GB. The proposed method introduces only 1.39M trainable parameters, as the backbone network is frozen and only lightweight MEP and Dynamic Prompter modules are trained. This design achieves a favorable balance between tracking accuracy and resource utilization, making DPTracker suitable for real-world, resource-constrained scenarios.

5. Conclusion

This paper presented DPTracker, an RGB-D tracking framework that integrates visual prompt learning with dynamic modality prompting. By introducing a Modality Effectiveness Predictor (MEP) to assess modality reliability and a Dynamic Prompter (DP) to adaptively fuse RGB and depth information, DPTracker effectively enhances tracking

robustness under varying scene conditions. Built upon a frozen pre-trained RGB model, the proposed DPTracker achieves efficient adaptation with minimal additional parameters. Experimental results on multiple benchmarks demonstrate that DPTracker outperforms existing RGB-D trackers in both accuracy and stability, validating the effectiveness of dynamic prompting guided by modality validity.

Limitation: MEP relies solely on RGB features to predict depth modality quality. In scenarios where both RGB and depth modalities are severely degraded, the prediction accuracy may decrease, affecting dynamic fusion performance.

Future Work: DPTracker predicts modality reliability via global RGB features, which is suboptimal under severe modality degradation. Future work will incorporate cross-modal and depth-aware cues into the MEP, extend global weighting to spatial and token-wise adaptive schemes, and investigate more flexible learnable fusion in the Dynamic Prompter for enhanced robustness in complex scenarios.

References

[1] G. Bhat, M. Danelljan, L. V. Gool, R. Timofte, Learning discriminative model prediction for tracking, in: IEEE International Conference on Computer Vision (ICCV), 2019, pp. 6182–6191.

[2] B. Ye, H. Chang, B. Ma, S. Shan, X. Chen, Joint feature learning and relation modeling for tracking: A one-stream framework, in: European Conference on Computer Vision (ECCV), 2022, pp. 341–357.

[3] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, P. H. Torr, Fully-convolutional siamese networks for object tracking, in: European conference on computer vision, Springer, 2016, pp. 850–865.

[4] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, H. Lu, Transformer tracking, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 8126–8135.

[5] S. Yan, J. Yang, J. Käpylä, F. Zheng, A. Leonardis, J.-K. Kämäräinen, Depthtrack: Unveiling the power of rgbd tracking, in: IEEE International Conference on Computer Vision (ICCV), 2021, pp. 10725–10733.

[6] A. Lukezic, U. Kart, J. Kapyla, A. Durmush, J.-K. Kamarainen, J. Matas, M. Kristan, Cdtb: A color and depth visual object tracking dataset and benchmark, in: IEEE International Conference on Computer Vision (ICCV), 2019, pp. 10013–10022.

[7] Y. Liu, X.-Y. Jing, J. Nie, H. Gao, J. Liu, G.-P. Jiang, 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.

[8] J. Yang, Z. Li, F. Zheng, A. Leonardis, J. Song, Prompting for multi-modal tracking, in: ACM International Conference on Multimedia (ACM MM), 2022, pp. 3492–3500.

[9] J. Zhu, S. Lai, X. Chen, D. Wang, H. Lu, Visual prompt multi-modal tracking, in: IEEE International Conference on Computer Vision (CVPR), 2023, pp. 9516–9526.

[10] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, S.-N. Lim, Visual prompt tuning, in: European Conference on Computer Vision (ECCV), 2022, pp. 709–727.

[11] B. Lester, R. Al-Rfou, N. Constant, The power of scale for parameter-efficient prompt tuning, arXiv preprint arXiv:2104.08691 (2021).

[12] H. Bahng, A. Jahanian, S. Sankaranarayanan, P. Isola, Exploring visual prompts for adapting large-scale models, arXiv preprint arXiv:2203.17274 (2022).

[13] L. Alzubaidi, J. Bai, A. Al-Sabaawi, J. Santamaría, A. S. Albahri, B. S. N. Al-Dabbagh, M. A. Fadhel, M. Manoufali, J. Zhang, A. H. Al-Timemy, et al., A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications, Journal of Big Data 10 (1) (2023) 46.

[14] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, J.-K. Kämäräinen, H. J. Chang, M. Danelljan, L. Č. Zajc, A. Lukežič, et al., The tenth visual object tracking vot2022 challenge results, in: European Conference on Computer Vision (ECCV), 2022, pp. 431–460.

[15] Y. Liu, X.-Y. Jing, J. Nie, H. Gao, J. Liu, G.-P. Jiang, 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.

[16] U. Kart, A. Lukezic, M. Kristan, J.-K. Kamarainen, J. Matas, Object tracking by reconstruction with view-specific discriminative correlation filters, in: IEEE International Conference on Computer Vision (CVPR), 2019, pp. 1339–1348.

[17] Y. Qian, S. Yan, A. Lukežič, M. Kristan, J.-K. Kämäräinen, J. Matas, Dal: A deep depth-aware long-term tracker, in: 2020 25th International conference on pattern recognition (ICPR), 2021, pp. 7825–7832.

[18] A. Dosovitskiy, An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929 (2020).

[19] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kamarainen, L. ˇCehovin Zajc, O. Drbohlav, A. Lukezic, A. Berg, et al., The seventh visual object tracking vot2019 challenge results, in: Proceedings of the IEEE/CVF international conference on computer vision workshops, 2019, pp. 0–0.

[20] P. Zhao, Q. Liu, W. Wang, Q. Guo, Tsdm: Tracking by siamrpn++ with a depth-refiner and a mask-generator, in: International Conference on Pattern Recognition (ICPR), 2021, pp. 670–676.

[21] S. Gao, J. Yang, Z. Li, F. Zheng, A. Leonardis, J. Song, Learning dual-fused modality-aware representations for rgbd tracking, in: European Conference on Computer Vision (ECCV), 2022, pp. 478–494.

[22] X.-F. Zhu, T. Xu, S. Atito, M. Awais, X.-J. Wu, Z. Feng, J. Kittler, Self-supervised learning for rgb-d object tracking, Pattern Recognition 155 (2024) 110543.

[23] X.-F. Zhu, T. Xu, X.-J. Wu, J. Kittler, Feature enhancement and coarse-to-fine detection for rgb-d tracking, Pattern recognition letters 179 (2024) 130–136.

[24] X.-F. Zhu, T. Xu, X.-J. Wu, Adaptive colour-depth aware attention for rgb-d object tracking, IEEE Signal Processing Letters (2024).

[25] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kämäräinen, H. J. Chang, M. Danelljan, L. Cehovin, A. Lukežič, et al., The ninth visual object tracking vot2021 challenge results, in: IEEE International Conference on Computer Vision Workshops (ICCVW), 2021, pp. 2711–2738.

[26] X.-F. Zhu, T. Xu, Z. Tang, Z. Wu, H. Liu, X. Yang, X.-J. Wu, J. Kittler, Rgbd1k: A large-scale dataset and benchmark for rgb-d object tracking, in: AAAI Conference on Artificial Intelligence (AAAI), Vol. 37, 2023, pp. 3870–3878.

[27] Z. Wu, J. Zheng, X. Ren, F.-A. Vasluianu, C. Ma, D. P. Paudel, L. Van Gool, R. Timofte, Single-model and any-modality for video object tracking, in: IEEE International Conference on Computer Vision (CVPR), 2024, pp. 19156–19166.

[28] L. Gao, Y. Ke, W. Zhao, Y. Zhang, Y. Jiang, G. He, Y. Li, Rgb-d visual object tracking with transformer-based multi-modal feature fusion, Knowledge-Based Systems (2025) 113531.

[29] G. Ying, D. Zhang, Z. Ou, X. Wang, Z. Zheng, Temporal adaptive bidirectional bridging for rgb-d tracking, Pattern Recognition 158 (2025) 111053.

[30] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, J.-K. Kämäräinen, M. Danelljan, L. Č. Zajc, A. Lukežič, O. Drbohlav, et al., The eighth visual object tracking vot2020 challenge results, in: European Conference on Computer Vision Workshops (ECCV), 2020, pp. 547–601.

[31] Z. Tang, T. Xu, X.-J. Wu, J. Kittler, M 3 track: Meta-prompt for multi-modal tracking, IEEE Signal Processing Letters (2025).