Latent Spectral Prompting for Multimodal Hyperspectral Object Tracking

Wangshu Cai[1] Jun Zhou[2*] Lei Wang[1*]
Mohammad Aminul Islam[3] Xu Han[1] Kuldip Paliwal[1]
[1]Griffith University, Australia, [2]East China Normal University, China,
[3]Bangladesh Agricultural University, Bangladesh
{wangshu.cai, xu.han2}@griffithuni.edu.au, {l.wang4, k.paliwal}@griffith.edu.au,
amin@bau.edu.bd, jzhou@ee.ecnu.edu.cn

Abstract

Hyperspectral videos provide rich spectral signatures for robust tracking under challenging conditions such as camouflage and low illumination. However, limited hyperspectral annotations and the large RGB-hyperspectral domain gap hinder the direct use of modern temporally pretrained RGB trackers. We propose SProTrack, a multimodal framework that transfers temporal knowledge from pretrained RGB trackers to the hyperspectral domain through latent spectral prompting. The key idea is to decouple temporal modeling from spectral adaptation, allowing temporal dynamics to be inherited from RGB foundation models while hyperspectral information is learned from unlabeled data in a self-supervised manner. Unlike band-selection methods that discard most spectral information or prompt-based approaches that rely on local-search crops, SProTrack learns a global, scene-level spectral representation from the full hyperspectral frame using an autoencoder. This latent representation is used to guide spectral prompt extraction while a pretrained RGB tracker preserves strong temporal modeling on RGB-compatible inputs. By encoding spectral-spatial structure into compact latent prompts, SProTrack bridges the RGB-hyperspectral modality gap without extensive retraining or dense supervision. Experiments on HOT2020 and HOT2023 demonstrate that SProTrack achieves the best overall performance, with strong results across all spectral domains.

1. Introduction

Visual object tracking is a fundamental problem in computer vision, aiming to localize a target object across video frames given its initial state. Recent advances, driven by large-scale RGB datasets and transformer-based architectures, have significantly improved tracking performance, particularly in modeling long-range temporal dynamics [4, 27, 34, 36, 53]. However, RGB imagery is inherently limited to three spectral channels, making it vulnerable to challenging scenarios such as low illumination, camouflage, and appearance ambiguity. To compensate for this spectral limitation, Hyperspectral imaging offers a compelling alternative by capturing dense spectral information across dozens or hundreds of wavelength bands. These rich spectral signatures encode intrinsic material properties that are invariant to appearance changes, providing strong discriminative cues for robust tracking. Despite this advantage, hyperspectral object tracking (HOT) remains underexplored compared to its RGB counterpart [19, 33]. The primary bottleneck lies in the scarcity of annotated hyperspectral video data, which makes it difficult to train modern deep tracking models without severe overfitting.

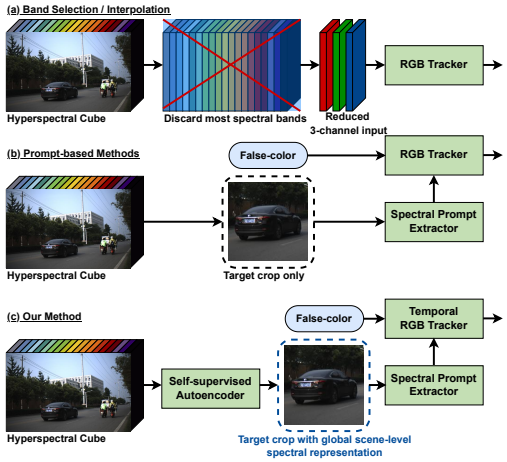


Figure 1. Hyperspectral tracking paradigms. (a) Band selection discards most bands, using a reduced 3-channel input. (b) Prompt-based methods keep all bands but extract spectral features only from local crops. (c) Ours encodes the full frame into global scene-level spectral prompts for a temporal RGB tracker.

A natural solution is to transfer knowledge from pre-

*Corresponding authors.

trained RGB trackers [22, 41]. However, this introduces a fundamental challenge: *the spectral domain gap*. Existing approaches bridge this gap either by reducing hyperspectral data to three channels via band selection or interpolation [9, 39, 42, 55], or by directly learning spectral representations through fusion or prompting [5, 12, 18, 46] and integrating them with pre-trained RGB trackers in an end-to-end manner. The overviews of the two methods are shown in Fig. 1 (a) and (b). Band selection methods inevitably discard valuable inter-band correlations, while end-to-end approaches jointly learn spectral representations and temporal dynamics from limited data, leading to suboptimal generalization. More recent prompt-based methods mitigate this issue by injecting auxiliary modality information into pretrained RGB backbones [26, 40, 59]. Nevertheless, these methods typically operate on local search regions and lack explicit temporal modeling, limiting their ability to fully exploit both spectral and temporal information [6, 44].

A common thread runs through these failures: band selection, end-to-end fusion, and local prompting all entangle spectral adaptation with temporal modeling, forcing both to be learned jointly from scarce data. We therefore revisit hyperspectral tracking from a multimodal perspective, where the key challenge is not merely adapting hyperspectral inputs but decoupling and integrating complementary information across modalities: inheriting strong temporal priors from RGB foundation models while learning robust spectral representations from limited hyperspectral data (see Fig. 1(c)). Building on this insight, we propose SProTrack, a multimodal tracking framework that transfers temporal knowledge from pretrained RGB trackers through latent spectral prompting in two phases. First, a Spectral-Spatial AutoEncoder (SSAE) is trained in a self-supervised manner on full hyperspectral frames, exploiting abundant unlabeled data to learn compact latent representations that capture global scene context beyond the target box. Second, these representations are injected as visual prompts into a frozen, temporally-aware RGB backbone, inheriting its temporal dynamics while adapting to hyperspectral inputs through lightweight LoRA [13] modulation.

These three components address the limitations identified above in a unified manner. First, decoupling spectral representation learning from temporal modeling avoids the overfitting that arises when end-to-end methods learn both jointly from scarce annotations. Second, injecting latent spectral prompts enables deep, layer-wise multimodal fusion without modifying the backbone, in contrast to the shallow, input-level fusion of prior prompt-based methods. Third, full-frame self-supervised learning lets the model exploit the global scene context that band-selection and local-crop approaches discard. Our main **contributions** are:

i. We formulate hyperspectral object tracking as a *multimodal adaptation problem* and propose a decoupled learning paradigm that separates temporal modeling from spectral representation learning, mitigating overfitting under limited annotations.
ii. We introduce a *Spectral-Spatial AutoEncoder (SSAE)* that learns compact latent representations from full hyperspectral frames in a self-supervised manner, capturing global scene-level context and preserving inter-band correlations beyond local regions.
iii. We propose a *latent spectral prompting* that injects layer-wise representations into a frozen, temporally-aware RGB backbone, achieving deep multimodal fusion while retaining strong temporal priors, and yielding strong performance across all spectral domains.

2. Related Work

Hyperspectral object tracking. Early HOT methods are primarily based on generative models and handcrafted spectral features, such as spectral angle mapper (SAM) and radiative transfer modeling. These approaches explicitly model reflectance properties but often fail to capture complex spatial-temporal dynamics, leading to limited robustness in challenging scenarios [3, 29, 48]. More recent works adopt discriminative learning frameworks. For instance, channel selection strategies reduce spectral redundancy by selecting informative bands, improving efficiency but discarding inter-band correlations [20, 54, 56]. Attention-based models further model spectral dependencies through hierarchical or hybrid attention, but still suffer from the trade-off between spectral richness and noise redundancy [28–30]. Unlike these approaches that either compress spectral information heuristically or entangle spectral-spatial modeling within a single network, our method learns a *compact latent spectral representation* from full hyperspectral frames in a self-supervised manner, preserving global spectral structure without manual band selection or heavy architectural coupling.

Cross-modal transfer & RGB-based tracking adaptation. Given the scarcity of hyperspectral annotations, a dominant paradigm is to transfer knowledge from RGB trackers pretrained on large-scale datasets [7, 16, 31, 35]. Early approaches adopt full fine-tuning, directly adapting RGB models to hyperspectral inputs, but often suffer from overfitting under insufficient data. To address this, recent works explore parameter-efficient adaptation, hybrid architectures that bridge spectral and spatial features, and domain adaptation modules that learn domain-invariant representations across sensors and environments [38, 45]. Existing transfer-based methods typically *jointly learn spectral representation and temporal dynamics*, which leads to suboptimal learning under limited data. In contrast, we explicitly *decouple temporal modeling from spectral adaptation*: a frozen RGB backbone preserves strong temporal priors, while spectral information is injected through

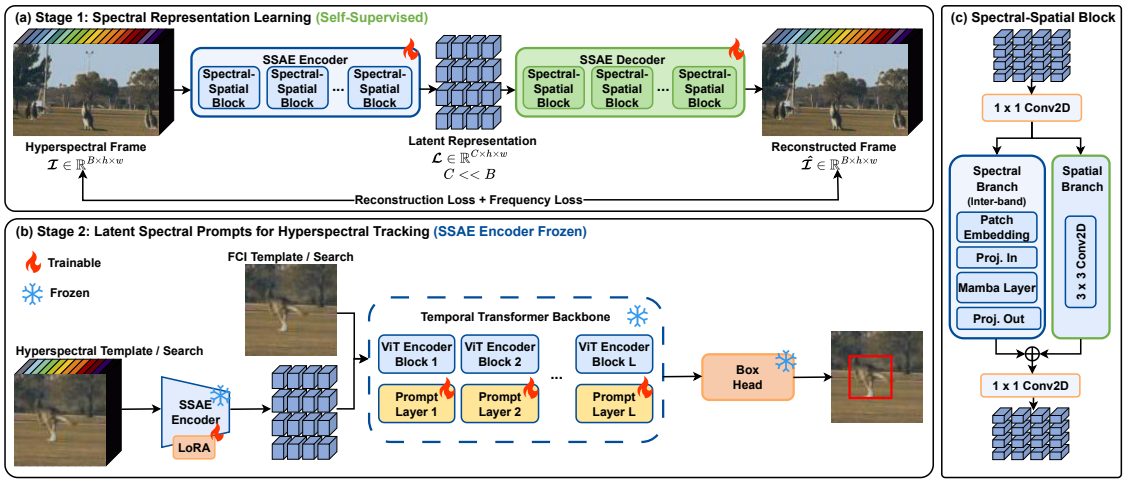


Figure 2. Overview of SProTrack. (a) Stage 1: a Spectral-Spatial AutoEncoder (SSAE) is trained self-supervisedly to encode hyperspectral frames into compact latent representations. (b) Stage 2: the frozen SSAE encoder generates latent spectral prompts, injected into a transformer-based RGB backbone for layer-wise modulation. (c) Spectral-Spatial block: a spectral branch models inter-band dependencies via Mamba and a spatial branch captures local patterns via convolution; their fused outputs, stacked across blocks, form the encoder, with a symmetric decoder for reconstruction. Detailed implementations are provided in the supplementary material.

lightweight latent prompts, improving data efficiency and avoiding catastrophic forgetting.

Prompt learning & multimodal representation. Prompt learning has recently emerged as an effective paradigm for adapting large pretrained models with minimal parameters. In visual tracking, early works such as ProTrack [51] introduce prompt-based modulation, and subsequent methods extend this to hyperspectral tracking by learning modality-specific prompts while keeping the backbone frozen [2, 11, 12, 52, 58]. A parallel line explicitly models spectral-spatial-temporal interactions with transformer or state-space architectures [25, 43, 50], but often requires training complex models from scratch, which limits its scalability under data scarcity. Despite their success, existing prompt-based methods exhibit three key limitations: (i) *Locality*: prompts are typically learned from local search regions, limiting global spectral context; (ii) *Shallow fusion*: prompts are often injected at the input level, lacking deep integration across layers; (iii) *Lack of explicit representation learning*: spectral information is not modeled in a structured latent space. Our method addresses these limitations through a *latent spectral prompting mechanism* learned via a self-supervised autoencoder on full frames, enabling global, compact representations injected into the backbone for deeper multimodal fusion.

3. Method

Problem setting. Let $\{\mathcal{I}_t\}_{t=1}^{T}$ denote a hyperspectral video of length T, where each frame $\mathcal{I}_t \in \mathbb{R}^{B\times h\times w}$ contains B spectral bands. Given the initial bounding box $\mathbf{b}_0 \in \mathbb{R}^4$, the objective is to estimate the target state $\mathbf{b}_t$ for each subsequent frame t. Following the tracking-by-detection paradigm, each frame is represented by a *search region* $\boldsymbol{\mathcal{X}}_t$ and a *template set* $\boldsymbol{\mathcal{Z}}_t$ constructed from historical target appearances. The tracking model is formulated as: $\mathcal{F}: (\boldsymbol{\mathcal{X}}_t, \boldsymbol{\mathcal{Z}}_t, \boldsymbol{\Theta}_{t-1}) \rightarrow (\mathbf{b}_t, \boldsymbol{\Theta}_t)$, where $\boldsymbol{\Theta}_t$ denotes the temporal state encoding trajectory information across frames. We adopt a transformer-based RGB tracker as the temporal backbone. The search and template images are divided into non-overlapping patches and embedded into token sequences $\boldsymbol{S}_x \in \mathbb{R}^{D\times N_x}$ and $\boldsymbol{S}_z \in \mathbb{R}^{D\times N_z}$. These tokens are concatenated with a temporal token $\boldsymbol{\Theta}_t$:

$$\boldsymbol{\Sigma}_0^t = \boldsymbol{S}_x \oplus \boldsymbol{S}_z \oplus \boldsymbol{\Theta}_t, \tag{1}$$

and processed by a transformer encoder $\mathcal{T}$ with L layers:

$$\boldsymbol{\Sigma}_l^t = \mathcal{T}_l(\boldsymbol{\Sigma}_{l-1}^t), \quad l = 1, \ldots, L. \tag{2}$$

The temporal token is updated and propagated across frames to maintain trajectory consistency.

Overview. As shown in Fig. 2, SProTrack decouples temporal modeling from spectral adaptation in two stages: a self-supervised SSAE learns compact spectral representations (Sec. 3.2), which are then injected as layer-wise prompts into a temporal RGB tracking backbone (Sec. 3.3).

3.1. Decoupled Multimodal Framework

HOT involves modeling both temporal dynamics across frames and spectral characteristics across wavelength bands. Jointly learning these components from limited hyperspectral data often leads to overfitting due to data scarcity. To address this limitation, we propose a *decou-*

pled multimodal framework that separates temporal modeling from spectral representation learning, allowing each component to be optimized independently and more effectively. The tracking process is decomposed into:

$$\mathbf{p}_t = \mathcal{P}(\boldsymbol{\mathcal{L}}_t), \quad \mathbf{b}_t = \mathcal{T}_{\text{RGB}}\big(\boldsymbol{\Sigma}_t;\, \mathbf{p}_t\big), \tag{3}$$

where $\mathcal{T}_{\text{RGB}}$ denotes a pretrained RGB tracking backbone and $\mathcal{P}$ denotes the prompt module that generates spectral prompts $\mathbf{p}_t$ from the latent representation $\boldsymbol{\mathcal{L}}_t$. This formulation explicitly decouples temporal modeling from spectral adaptation. We use a pretrained transformer-based tracker to model temporal dynamics and keep its parameters frozen. This preserves strong trajectory modeling capability learned from large-scale RGB data and avoids overfitting under limited hyperspectral supervision. Spectral information is captured independently through a compact latent representation learned from full hyperspectral frames. This representation encodes global inter-band correlations and serves as the basis for cross-modal adaptation.

Multimodal fusion is achieved via latent prompting, where the spectral representation is transformed into prompts that modulate intermediate features of the RGB backbone. This enables deep, layer-wise interaction between spectral and temporal information without altering the backbone architecture. Decoupling representation learning from temporal modeling improves data efficiency and preserves pretrained temporal knowledge, treating spectral cues as complementary guidance rather than a replacement for RGB temporal modeling.

3.2. Latent Spectral Representation Learning

Spectral-spatial encoding. To support the decoupled framework, we learn a compact and expressive latent representation from hyperspectral data in a self-supervised manner. Given a hyperspectral frame $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{B\times h\times w}$ with B spectral bands, we aim to learn a low-dimensional latent representation $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{C\times h\times w}$, where $C \ll B$. This representation preserves spatial structures and inter-band correlations while remaining efficient for downstream integration. We adopt an *SSAE* consisting of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$:

$$\boldsymbol{\mathcal{L}} = \mathcal{E}(\boldsymbol{\mathcal{I}}), \quad \hat{\boldsymbol{\mathcal{I}}} = \mathcal{D}(\boldsymbol{\mathcal{L}}). \tag{4}$$

The encoder jointly models spectral dependencies and spatial structures through a sequence of Spectral-Spatial Blocks (see Fig. 2 (c)). Given feature maps $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{C\times H\times W}$, each block consists of two complementary branches. The spectral branch captures long-range inter-band dependencies by partitioning features into spatial patches and reshaping them into spectral token sequences, which are processed by a sequence modeling module. The spatial branch applies convolutional operations to capture local spatial patterns independent of spectral ordering. The outputs of both branches are fused through a learnable projection, enabling joint spectral-spatial representation learning. We instantiate the sequence modeling module using a Mamba block, which efficiently models long-range dependencies with linear complexity, making it suitable for high-dimensional hyperspectral data. The final latent representation is obtained via a 1×1 convolution followed by normalization.

Learning objective. The SSAE is trained without annotations by reconstructing the input hyperspectral image (see Fig. 2(a)). We combine a spatial-domain and a frequency-domain term ($\lambda_f = 0.1$):

$$\mathcal{L}_{\text{SSAE}} = \left\|\mathcal{I} - \hat{\mathcal{I}}\right\|_1 + \lambda_f \left\| \,|\mathcal{F}_b(\mathcal{I})| - |\mathcal{F}_b(\hat{\mathcal{I}})| \,\right\|_1, \tag{5}$$

$\mathcal{F}_b(\cdot)$ is the orthonormal 2-D DFT taken over the two spatial dimensions of each band independently; no transform is applied along the spectral axis. Concretely, we compute the DFT of the input and of the reconstruction separately, take the magnitude of each, and average the absolute difference over all bands and frequency bins. Inter-band structure is modelled by the spectral branch of the encoder rather than by this objective. The resulting latent representation captures global spectral structure from full, unlabeled hyperspectral frames, and serves as a compact interface for the subsequent multimodal adaptation via prompting.

3.3. Latent Spectral Prompting

Prompt generation. We integrate spectral information into the tracking backbone through a latent prompting mechanism that enables efficient and deep multimodal fusion. Given the latent representation $\boldsymbol{\mathcal{L}}_t \in \mathbb{R}^{C\times h\times w}$, we first crop the latent to search/template regions corresponding to their false color counterparts, then project it into the token space to obtain spectral prompts:

$$\mathbf{p}_0^t = \phi(\text{cropped}(\boldsymbol{\mathcal{L}}_t)), \tag{6}$$

where $\phi(\cdot)$ denotes a learnable projection function.

Prompt injection. The resulting tokens $\mathbf{p}_0^t \in \mathbb{R}^{D\times N}$ encode compact spectral cues aligned with the RGB backbone. To enable deep fusion, the prompts are progressively computed and injected into each transformer layer. Since the SSAE operates on the full hyperspectral frame and preserves spatial correspondence with its input, cropping the latent by the same box coordinates used for the RGB search and template regions yields features that align with the corresponding RGB token grids. At layer l, the prompts are updated as:

$$\mathbf{p}_l^t = \mathcal{P}_l\big(\boldsymbol{\Sigma}_{l-1}^t, \mathbf{p}_{l-1}^t\big), \tag{7}$$

where $\mathcal{P}_l$ denotes a lightweight learnable prompt module that maps the previous backbone features and prompts to the layer-l prompt of dimension $D \times N$. The prompt is then used to modulate the corresponding backbone features:

$$\boldsymbol{\Sigma}_l^t = \mathcal{T}_l\big(\boldsymbol{\Sigma}_{l-1}^t + \mathbf{p}_l^t\big). \tag{8}$$

Table 1. State-of-the-art comparison on HOT2020 and HOT2023. RGB trackers use RGB/FCI inputs, while hyperspectral methods use FCI + HSI. Best and second-best results are in **bold** and underline.

	Method	HOT2020		HOT2023							
		RGB/HSI		FCI/NIR		FCI/RedNIR		FCI/VIS		ALL	
		DP	AUC	DP	AUC	DP	AUC	DP	AUC	DP	AUC
RGB trackers	SeqTrack [4]	0.908	0.668	0.834	0.643	0.631	0.470	0.831	0.603	0.798	0.593
	OSTrack [53]	0.871	0.636	0.887	0.670	0.631	0.498	0.796	0.590	0.807	0.606
	SMAT [10]	0.831	0.600	0.866	0.669	0.448	0.335	0.744	0.531	0.737	0.550
	AQATrack [47]	0.865	0.647	0.809	0.618	0.613	0.424	0.820	0.603	0.785	0.585
	EVPTrack [37]	0.870	0.649	0.843	0.644	0.574	0.424	0.801	0.590	0.785	0.585
	ARTrackV2 [1]	0.868	0.635	0.910	0.667	0.524	0.375	0.798	0.547	0.800	0.568
	ODTrack [57]	0.930	0.693	0.924	0.694	0.606	0.475	0.859	0.653	0.839	0.641
Hyperspectral trackers	MHT [21]	0.883	0.586	0.756	0.418	0.442	0.316	0.765	0.507	0.717	0.453
	ViPT [59]	0.909	0.669	0.950	0.742	0.709	0.563	0.858	0.641	0.871	0.668
	SEE-Net [23]	0.933	0.666	0.785	0.465	0.491	0.359	0.813	0.593	0.763	0.519
	SPIRIT [6]	0.924	0.678	0.838	0.623	0.504	0.381	0.822	0.608	0.786	0.585
	SiamCAT [15]	0.907	0.646	0.920	0.665	0.465	0.479	0.826	0.596	0.838	0.605
	MMF-Net [24]	0.931	0.689	0.910	0.666	0.410	0.324	0.815	0.613	0.797	0.595
	Trans-DAT [45]	0.769	0.532	0.900	0.683	0.726	0.535	0.725	0.511	0.782	0.573
	DaSSPNet [26]	0.916	0.680	0.952	**0.746**	0.731	**0.587**	0.865	0.654	0.878	0.678
Ours	**SProTrack**	**0.952**	**0.704**	**0.970**	0.741	**0.742**	0.583	**0.886**	**0.665**	**0.906**	**0.697**

Here, the addition is element-wise over corresponding spatial tokens, while the temporal token is kept unchanged. This layer-wise interaction allows spectral information to influence representations at multiple semantic levels.

Temporal consistency. We preserve temporal modeling by maintaining a temporal token $\boldsymbol{\Theta}_t \in \mathbb{R}^D$, which encodes the target trajectory across frames. The input sequence is constructed as:

$$\boldsymbol{\Sigma}_0^t = \boldsymbol{S}_x^t \oplus \boldsymbol{S}_z^t \oplus \boldsymbol{\Theta}_t, \tag{9}$$

and the temporal state is propagated as:

$$\boldsymbol{\Theta}_{t+1} = \boldsymbol{\Theta}_L^t + \boldsymbol{\Theta}_{\text{init}}, \tag{10}$$

where $\boldsymbol{\Theta}_{\text{init}} \in \mathbb{R}^D$ is a learnable initialization token. By keeping the temporal update unchanged, the model retains robust trajectory modeling while incorporating spectral guidance. Latent spectral prompting enables implicit multimodal fusion through layer-wise feature modulation, allowing efficient adaptation to hyperspectral inputs while preserving pretrained temporal knowledge.

3.4. Training and Adaptation

We adopt a two-stage training strategy to optimize the proposed framework in a data-efficient manner.

Stage 1: spectral representation learning. The SSAE is trained in a fully self-supervised manner using hyperspectral data without annotations. We use the loss defined in Eq. (5). The SSAE is trained only on the training split of each benchmark to avoid any test-time information leakage.

Stage 2: multimodal tracking. The tracking model is initialized with the pretrained SSAE encoder and a pretrained RGB backbone (see Fig. 2 (b)). The backbone parameters are frozen to preserve temporal modeling capability, while only the prompt generation modules, hyperspectral projection layers, and LoRA adapters are optimized. To improve parameter efficiency, we use LoRA to selected layers (e.g., projection-in and projection-out layers implemented as fully connected layers, as shown in Fig. 2 (c)) in the spectral encoder, enabling effective task adaptation with minimal additional parameters. The overall tracking objective is defined as:

$$\mathcal{L} = L_{\text{cls}} + \lambda_1 L_1 + \lambda_2 L_{\text{GIoU}}, \tag{11}$$

where L_{cls} is the classification loss, L_1 and L_{GIoU}, with corresponding scales $\lambda_1 = 5$ and $\lambda_2 = 2$, denote bounding box regression losses following ODTrack[57] settings. This training strategy uses self-supervised spectral learning and lightweight multimodal adaptation to achieve strong generalization under limited hyperspectral supervision.

4. Experiment

4.1. Datasets and Setups

Datasets. We evaluate on: *(i) HOT2020* [48] contains 40 training and 35 testing sequences, all captured in visible (VIS) spectral domain. *(ii) HOT2023* [49] extends this setting to multiple spectral domains, including VIS, near-infrared (NIR), and red-to-near-infrared (RedNIR), providing a more challenging and diverse evaluation scenario.

Setups. SProTrack is trained on a single NVIDIA RTX 5090 GPU, using ODTrack Base-Fulldata-300ep [8, 14, 32] as the frozen RGB foundation. For each spectral domain, we pretrain a separate SSAE for 150 epochs (batch size 16, patch size $p = 8$) using AdamW (initial lr 8×10^{-5}, weight decay $1{\times}10^{-4}$, cosine annealing), on full hyperspectral images without bounding-box annotations, with random

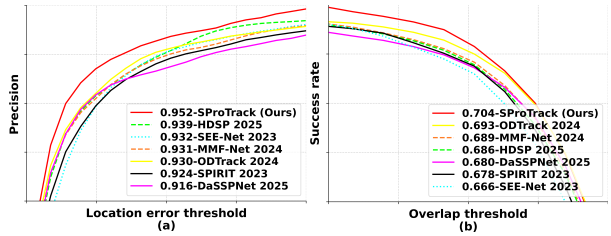


Figure 3. Precision and success curves on HOT2020, where (a) shows the precision plot and (b) shows the success plot.

cropping, rotation, and horizontal flipping as augmentation. For tracking, we optimize only the prompt modules, hyperspectral projection layers, and LoRA on SSAE, while keeping the backbone frozen; each epoch uses 7,200 sampled pairs (batch size 16) with three 384×384 search and three 192×192 template images sampled within max gap 400 frames. Here the three search frames form a temporally adjacent clip sharing one template set, processed autoregressively so the temporal token propagates across them, following ODTrack[57]. On VIS, we train 60 epochs with a step schedule (lr decayed $10\times$ after 50 epochs). On NIR and RedNIR, where data is scarcer, we train 40 epochs with a constant learning rate to prevent overfitting.

Metrics. We follow the standard one-pass evaluation protocol, where the tracker processes each sequence sequentially without re-initialization. Performance is evaluated using: (i) the area under the success curve (AUC), computed from the intersection-over-union between predicted and ground-truth bounding boxes, and (ii) distance precision (DP), defined as the percentage of frames where the center location error is within a 20-pixel threshold.

4.2. Comparison with State-of-the-Art

Tab. 1 presents a comparison with state-of-the-art RGB and hyperspectral trackers on HOT2020 and HOT2023. SProTrack achieves the best DP across all domains and the best overall (ALL) AUC. On HOT2020, our method attains 0.952 DP and 0.704 AUC, surpassing the strongest RGB baseline ODTrack[57] and recent hyperspectral methods such as DaSSPNet[26] (see Fig. 3). We trained with 5 random seeds and observed a standard deviation of 0.0018. On HOT2023, we achieve 0.906 DP and 0.697 AUC on the aggregated setting, improving over the second-best by a clear margin. Notably, SProTrack stays strong even on the challenging NIR and RedNIR domains where many methods degrade, owing to globally learned spectral representations that capture material properties beyond individual bands and are integrated via layer-wise prompting.

RGB trackers. When applied to hyperspectral data via false-color images, RGB trackers exhibit strong performance in the VIS domain but degrade in NIR and RedNIR settings due to the domain gap. For example, ODTrack[57] achieves competitive results on VIS but drops noticeably on RedNIR. In contrast, SProTrack consistently improves performance across all domains, indicating that latent spectral prompting effectively bridges the spectral gap without sacrificing temporal modeling capability.

Table 2. Comparison of efficiency and tracking performance (MACs, FPS, parameters, AUC).

Method	MACs (G)	FPS	Params (M)	AUC
ODTrack [57]	103.3	74.8	92.83	0.693
DaSSPNet [26]	25.9	43.8	95.56	0.680
SProTrack (ours)	145.14	43.1	111.5	0.704

Hyperspectral trackers. Compared to dedicated hyperspectral trackers, our method achieves the best overall performance, with superior or competitive results across individual domains. Notably, while DaSSPNet[26] achieves strong results in certain domains, our method provides more consistent gains across VIS, NIR, and RedNIR, leading to the best overall performance. This highlights the advantage of decoupling spectral representation learning from temporal modeling and integrating them through latent prompting.

4.3. Analysis and Discussion

Computational efficiency. We evaluate efficiency in terms of MACs, inference speed (FPS), and parameter count, as summarized in Tab. 2. SProTrack runs at 43.1 FPS on a single RTX 5090 GPU, satisfying real-time requirements and on par with DaSSPNet[26] (43.8 FPS) despite a substantially higher MACs count. This apparent gap arises because MACs measure theoretical operations, whereas wall-clock speed also reflects hardware efficiency: our Mamba-based spectral branch attains higher throughput than the Conv3D operations common in prior hyperspectral trackers, and the SSAE latent for the template is computed once and cached across frames rather than recomputed. As a result, the spectral modeling adds little to the per-frame critical path. While the total parameter count (111.5M) is higher than prior methods, the backbone is frozen and only 0.895M parameters (0.80%) are actually trainable, so cross-modal adaptation is achieved by tuning under 1% of the network. The higher MACs stem mainly from the transformer-based ODTrack[57] backbone, which operates on larger input resolutions and multiple reference frames. Beyond that the additional cost yields consistent accuracy gains: +1.1% AUC over ODTrack[57] and +2.4% over DaSSPNet[26], demonstrating a favorable accuracy-efficiency trade-off.

Attribute-based analysis. We report attribute-based AUC on HOT2020 across 11 attributes in Tab. 3. SProTrack ranks first on 6 attributes (DEF, FM, IV, OCC, OPR, SV) and second on 2 (BC, MB). Gains concentrate on two groups: appearance-variation attributes (DEF, OCC, IV), where latent spectral representations capture material properties that stay stable when RGB cues become unreliable;

Table 3. Attribute-based AUC comparison on the HOT2020 benchmark over 11 challenging attributes: background clutter (BC), deformation (DEF), fast motion (FM), illumination variation (IV), in-plane rotation (IPR), low resolution (LR), motion blur (MB), occlusion (OCC), out-of-plane rotation (OPR), out-of-view (OV), and scale variation (SV). The best and second results are in **bold** and underline.

Method	BC	DEF	FM	IV	IPR	LR	MB	OCC	OPR	OV	SV
BAE-Net [22]	0.651	0.680	0.608	0.525	0.699	0.489	0.593	0.555	0.701	0.517	0.604
SEE-Net [23]	0.705	0.710	0.711	0.566	0.742	0.626	0.702	0.622	0.729	0.633	0.649
SPIRIT [6]	**0.715**	0.723	0.715	0.567	0.760	0.629	0.723	0.643	0.751	0.689	0.665
ViPT [59]	0.633	0.764	0.685	0.642	0.743	0.637	**0.733**	0.667	0.749	**0.822**	0.673
DaSSPNet [26]	0.655	0.765	0.708	0.659	0.762	0.655	0.724	0.683	0.763	0.815	0.689
HDSP [52]	0.660	0.769	0.721	0.655	**0.767**	**0.672**	0.718	0.684	0.772	0.767	0.695
SProTrack (ours)	0.714	**0.782**	**0.727**	**0.665**	0.740	0.670	0.730	**0.694**	**0.773**	0.724	**0.704**

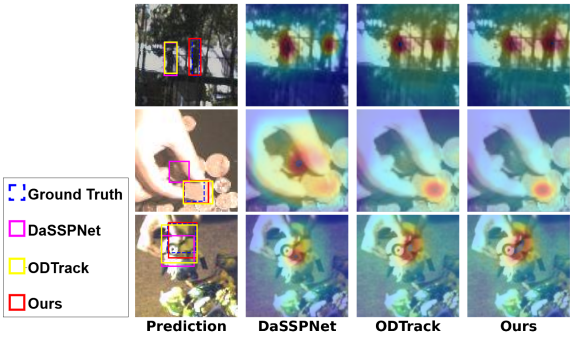


Figure 4. Qualitative comparison on three challenging HOT2020 sequences. Our method yields more concentrated responses under occlusion, distractors, and appearance ambiguity.

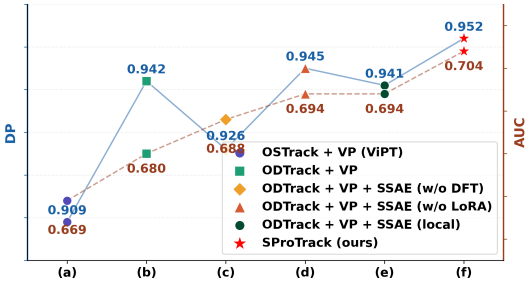


Figure 5. Ablation study on HOT2020 illustrating the impact of progressively adding key components. VP denotes visual prompting. Each point corresponds to a model variant, evaluated by distance precision (DP) and success (AUC).

and motion-dynamics attributes (FM, OPR, SV), where the inherited temporal priors from the frozen RGB backbone maintain trajectory continuity under rapid change. Our method slightly underperforms on OV and IPR by DaSSPNet [26] and SPIRIT [6], whose explicit domain-adaptation modules align spectral distributions across conditions, an advantage under extreme viewpoint or visibility changes.

Qualitative results. Fig. 4 compares three challenging HOT2020 sequences. Across all cases, our method produces more concentrated, well-centered responses, enabling accurate localization where competing methods drift. Under partial occlusion with complex illumination (row 1), baselines yield diffuse responses with background distractions, while ours stays compact. With a nearby similar distractor (row 2), competing methods activate both objects, whereas ours gives a sharp unimodal response. Under occlusion (row 3), most methods shift toward the occluder, while ours preserves activation on the true target. Further qualitative results, failure cases, statistical analysis, and limitations are provided in the supplementary material.

4.4. Ablation Study

We ablate each component on HOT2020 (Fig. 5, Tab. 4).

Temporal and spectral modeling are complementary. The 0.693 AUC reported for ODTrack in Tab. 1 is the plain ODTrack baseline, whereas the 0.680 AUC in Fig. 5(b) is ODTrack with visual prompting (VP); the latter is therefore the matched baseline for evaluating the additional effect of SSAE prompting. Moving from OSTrack[53]+VP (ViPT) to ODTrack[57]+VP raises AUC by 1.1% (Fig. 5(a) vs. (b)). Although this configuration performs worse than the plain version in Tab. 1, likely due to overfitting, the use of pretrained temporal priors is deliberate. Under severe data scarcity, training a strong temporal tracker from scratch is challenging, so our model reuses pretrained RGB temporal knowledge. On top of this backbone, SSAE prompting adds a further 2.4% AUC (Fig. 5(b) vs. (f)), exceeding the gain from the backbone change itself. The two components are thus complementary: reused temporal priors handle trajectory modeling, while spectral prompting supplies discriminative cues unavailable from RGB pretraining alone.

Global vs. local spectral context. To isolate the role of full-frame modeling, we train an SSAE variant (Fig. 5(e)) using only the hyperspectral content within the (jittered) search-crop region, matching the local-region inputs of prior prompt-based methods. Compared to no SSAE (Fig. 5(b)), this local variant yields almost no improvement, indicating that injecting spectral features from local crops alone adds little. In contrast, our full-frame SSAE (Fig. 5(f)) clearly surpasses both (+1.1% DP, +1.0% AUC). This shows that the benefit of SSAE comes not from adding spectral features per se, but from encoding *global*, scene-level context that local crops cannot capture, directly vali-

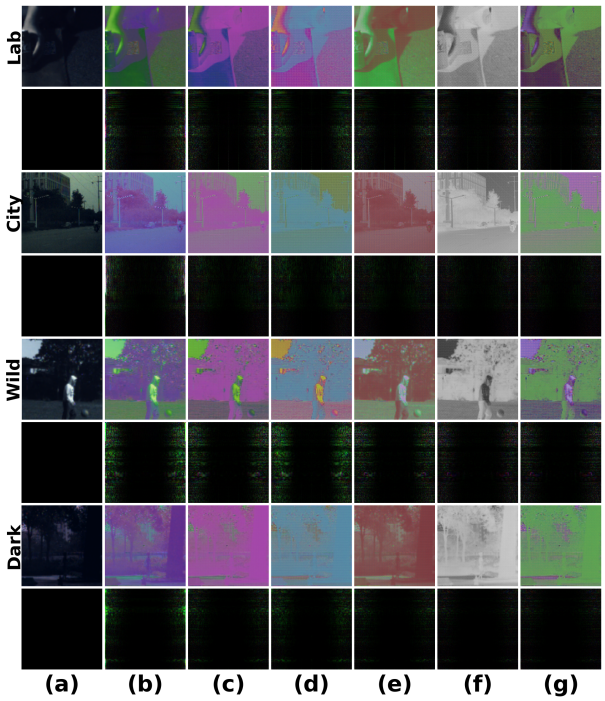


Figure 6. SSAE ablation on HOT2020. Each column shows the reconstructed latent and its frequency-domain error map (darker means lower error). (a) input false-color image and reference (zero-error) map. (b) – (g) SSAE configurations: (b) CNN-only; (c) dual-branch, 2 blocks; (d) 1 latent channel; (e) without frequency loss; (f) standard SSAE (dual-branch, 3 blocks, 3 channels); (g) 5 channels. Settings match Tab. 4.

dating our full-frame design.

Frequency-domain supervision. Removing DFT degrades precision from 0.952 to 0.926 DP (Fig. 5(c)). Because DFT operates on band-wise magnitude spectra under an ℓ_1 norm, it emphasizes the low-energy high-frequency components that a spatial reconstruction loss under-weights. The resulting sharper reconstruction of edges and texture (Fig. 6(e) vs. (f)) benefits center-accurate localization, which DP measures directly, more than it benefits overlap-based AUC.

Parameter-efficient adaptation. Removing LoRA (Fig. 5(d)) lowers both metrics (0.945 DP, 0.694 AUC); the lightweight adaptation recovers the gap with negligible overhead, indicating it suffices to align spectral representations with the tracking objective without overfitting.

Scalability. The decoupled design lets the SSAE leverage extra unlabeled data: pretraining on the larger HOT2023 set and transferring to HOT2020 yields further gains (+0.4% DP, +0.2% AUC) without any additional annotations.

SSAE design. We analyze the SSAE design in Tab. 4 and Fig. 6. The CNN-only encoder produces structured artifacts near image boundaries, while the dual-branch design better captures global spectral-spatial structure. Increasing the latent dimensionality from 1 to 3 substantially improves reconstruction and tracking, while the gain from 3 to 5 channels is marginal (+0.003 AUC) despite the increased latent dimensionality. We therefore adopt the 3-channel configuration as a more compact accuracy-efficiency trade-off. Removing frequency supervision increases residual high-frequency error, further supporting the use of the DFT loss.

Table 4. Comparison of configurations of SSAE.

Block	#Block	#Channel	Validation Loss	AUC
CNN	3	3	0.0613	0.694
Dual	2	3	0.0289	0.691
Dual	3	1	0.0397	0.693
Dual	3	3	0.0277	0.704
Dual	3	5	**0.0255**	**0.707**

Table 5. Cross-backbone generalization. Each backbone uses SProTrack under the same protocol.

Backbone	Paradigm	HOT2020		HOT2023 ALL	
		DP	AUC	DP	AUC
OSTrack [53]	one-stream	0.871	0.636	0.807	0.606
+ SProTrack		**0.922**	**0.665**	**0.875**	**0.669**
SeqTrack [4]	autoregressive	0.908	0.668	0.798	0.593
+ SProTrack		**0.923**	**0.677**	**0.878**	**0.669**
ODTrack [57]	temporal	0.930	0.693	0.839	0.641
+ SProTrack		**0.952**	**0.704**	**0.906**	**0.697**

Generalization across backbones. To verify that SProTrack is not tied to a particular architecture, we also evaluate it on OSTrack [53] and SeqTrack [4]. For each backbone, we compare the original tracker with its SProTrack-enhanced counterpart under the same protocol. As shown in Tab. 5, SProTrack's mechanism consistently improves AUC and DP on both HOT2020 and HOT2023 across all backbones, demonstrating that the performance gains are not specific to ODTrack [57]. SProTrack is not restricted to conventional RGB trackers and could, in principle, be adapted to zero-shot trackers such as CoTracker[16] and CoTracker 3[17] by injecting spectral prompts into their intermediate visual representations. However, adapting its point-tracking formulation to the bounding-box tracking setting of HOT would require additional design.

5. Conclusion

We present SProTrack, a decoupled framework that learns spectral representations independently of temporal modeling. It achieves state-of-the-art results across HOT2020 and HOT2023, with gains driven primarily by latent spectral prompting and effective spectral fusion rather than model capacity. Future work will explore band-agnostic spectral encoders, applying the learned spectral prompts to zero-shot trackers such as CoTracker[16], CoTracker 3[17], and SAM2[35], and explicit re-detection to address prolonged-occlusion and out-of-view failures.

Acknowledgments

Wangshu Cai is supported by the Griffith University International Postgraduate Research Scholarship and the Griffith University Postgraduate Research Scholarship. This work was supported in part by the ARC Grant DP250102625 'Spectral-spatial-temporal object tracking in hyperspectral videos'.

References

[1] Yifan Bai, Zeyang Zhao, Yihong Gong, and Xing Wei. Artrackv2: Prompting autoregressive tracker where to look and how to describe. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19048–19057. IEEE, 2024. 5

[2] Yidong Cai, Jie Liu, Jie Tang, and Gangshan Wu. Robust object modeling for visual tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9589–9600, 2023. 3

[3] Sujata Chakravarty, Bijay Kumar Paikaray, Rutuparnna Mishra, and Satyabrata Dash. Hyperspectral image classification using spectral angle mapper. In *2021 IEEE International Women in Engineering Conference on Electrical and Computer Engineering*, pages 87–90. IEEE, 2021. 2

[4] Xin Chen, Houwen Peng, Dong Wang, Huchuan Lu, and Han Hu. Seqtrack: Sequence to sequence learning for visual object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14572–14581, 2023. 1, 5, 8

[5] Yuzeng Chen, Yuqi Tang, Xin Su, Jie Li, Yi Xiao, Jiang He, and Qiangqiang Yuan. PHTrack: Prompting for hyperspectral video tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–18, 2024. 2

[6] Yuzeng Chen, Qiangqiang Yuan, Yuqi Tang, Yi Xiao, Jiang He, and Liangpei Zhang. SPIRIT: Spectral awareness interaction network with dynamic template for hyperspectral object tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16, 2024. 2, 5, 7, 15

[7] Yaman Dua, Ravi Shankar Singh, Kshitij Parwani, Smit Lunagariya, and Vinod Kumar. Convolution neural network based lossy compression of hyperspectral images. *Signal Processing: Image Communication*, 95:116255, 2021. 2

[8] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5374–5383, 2019. 5

[9] Siwei Feng, Yuki Itoh, Mario Parente, and Marco F Duarte. Hyperspectral band selection from statistical wavelet models. *IEEE Transactions on Geoscience and Remote Sensing*, 55(4):2111–2123, 2017. 2

[10] Goutam Yelluru Gopal and Maria A Amer. Separable self and mixed attention transformers for efficient object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6708–6717, 2024. 5

[11] Gaowei Guo, Zhaoxu Li, Wei An, Yingqian Wang, Xu He, Yihang Luo, Qiang Ling, Miao Li, and Zaiping Lin. Sptrack: Spectral similarity prompt learning for hyperspectral object tracking. *Remote Sensing*, 16(16):2975, 2024. 3

[12] Gang He, Long Gao, Langkun Chen, Yan Jiang, Weiying Xie, and Yunsong Li. Hyperspectral object tracking with spectral information prompt. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025. 2, 3

[13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2

[14] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:1562–1577, 2019. 5

[15] Xiao Jiang, Xinyu Wang, Chen Sun, Zengliang Zhu, and Yanfei Zhong. A channel adaptive dual siamese network for hyperspectral object tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–12, 2024. 5

[16] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European conference on computer vision*. Springer, 2024. 2, 8

[17] Nikita Karaev, Yuri Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labeling real videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2025. 8

[18] Yi Kong, Yuhu Cheng, Yang Chen, and Xuesong Wang. Joint classification of hyperspectral image and LiDAR data based on spectral prompt tuning. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–12, 2024. 2

[19] David Landgrebe. Hyperspectral image data analysis. *IEEE Signal processing magazine*, 19(1):17–28, 2002. 1

[20] Ke Li, Dengxin Dai, and Luc Van Gool. Jointly learning band selection and filter array design for hyperspectral imaging. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6384–6394, 2023. 2

[21] Weiping Li, Mingjing Li, and Lixia Cao. Improved hyperspectral target tracking algorithm. In *2022 2nd International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology*, pages 650–653, 2022. 5

[22] Zhuanfeng Li, Fengchao Xiong, Jun Zhou, Jing Wang, Jianfeng Lu, and Yuntao Qian. BAE-Net: A band attention aware ensemble network for hyperspectral object tracking. In *2020 IEEE International Conference on Image Processing*, pages 2106–2110. IEEE, 2020. 2, 7

[23] Zhuanfeng Li, Fengchao Xiong, Jun Zhou, Jianfeng Lu, and Yuntao Qian. Learning a deep ensemble network with band importance for hyperspectral object tracking. *IEEE Transactions on Image Processing*, 32:2901–2914, 2023. 5, 7, 15

[24] Zhuanfeng Li, Fengchao Xiong, Jun Zhou, Jianfeng Lu, Zhuang Zhao, and Yuntao Qian. Material-guided multiview fusion network for hyperspectral object tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024. 5, 15

[25] Zhuanfeng Li, Jing Wang, Jue Zhang, Dong Zhao, Guanyiman Fu, Jiangtao Wang, and Jianfeng Lu. A unified spatial-spectral-temporal network for hyperspectral object tracking. *Pattern Recognition*, page 113005, 2025. 3

[26] Zhuanfeng Li, Fengchao Xiong, Jianfeng Lu, Jing Wang, Diqi Chen, Jun Zhou, and Yuntao Qian. Multi-domain universal representation learning for hyperspectral object tracking. *Pattern Recognition*, 162:111389, 2025. 2, 5, 6, 7, 12, 15

[27] Liting Lin, Heng Fan, Zhipeng Zhang, Yaowei Wang, Yong Xu, and Haibin Ling. Tracking meets lora: Faster training, larger model, stronger performance. In *European Conference on Computer Vision*, pages 300–318. Springer, 2024. 1

[28] Zhenqi Liu, Xinyu Wang, Meng Shu, Guanzhong Li, Chen Sun, Ziying Liu, and Yanfei Zhong. An anchor-free siamese target tracking network for hyperspectral video. In *2021 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–5. IEEE, 2021. 2

[29] Zhenqi Liu, Xinyu Wang, Yanfei Zhong, Meng Shu, and Chen Sun. SiamHYPER: Learning a hyperspectral object tracker from an rgb-based tracker. *IEEE Transactions on Image Processing*, 31:7116–7129, 2022. 2

[30] Zhenqi Liu, Yanfei Zhong, Guorui Ma, Xinyu Wang, and Liangpei Zhang. A deep temporal-spectral–spatial anchor-free siamese tracking network for hyperspectral video object tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16, 2024. 2

[31] Shaohui Mei, Jingyu Ji, Yunhao Geng, Zhi Zhang, Xu Li, and Qian Du. Unsupervised spatial–spectral feature learning by 3d convolutional autoencoder for hyperspectral classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57:6808–6820, 2019. 2

[32] 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 *Proceedings of the European Conference on Computer Vision*, pages 300–317, 2018. 5

[33] Antonio Plaza, Jon Atli Benediktsson, Joseph W Boardman, Jason Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, Paolo Gamba, Anthony Gualtieri, et al. Recent advances in techniques for hyperspectral image processing. *Remote sensing of environment*, 113:S110–S122, 2009. 1

[34] Arjun Raj, Lei Wang, and Tom Gedeon. Tracknetv4: Enhancing fast sports object tracking with motion attention maps. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. IEEE, 2025. 1

[35] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, 2025. 2, 8

[36] Arpit Sharma, Lei Wang, and Yongsheng Gao. Rethinking object detection and tracking. In *Companion Proceedings of the ACM Web Conference 2026*, pages 1259–1268, 2026. 1

[37] Liangtao Shi, Bineng Zhong, Qihua Liang, Ning Li, Shengping Zhang, and Xianxian Li. Explicit visual prompts for visual object tracking. In *AAAI*, 2024. 5

[38] Chen Sun, Xinyu Wang, Zhenqi Liu, Yuting Wan, Liangpei Zhang, and Yanfei Zhong. Siamohot: A lightweight dual siamese network for onboard hyperspectral object tracking via joint spatial-spectral knowledge distillation. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–12, 2023. 2

[39] Weiwei Sun and Qian Du. Hyperspectral band selection: A review. *IEEE Geoscience and Remote Sensing Magazine*, 7 (2):118–139, 2019. 2

[40] Yumei Sun, Chuanming Tang, Hui Luo, Qingqing Li, Xiaoming Peng, Jianlin Zhang, Meihui Li, and Yuxing Wei. Joint spatio-temporal modeling for visual tracking. *Knowledge-Based Systems*, 283:111206, 2024. 2

[41] Yiming Tang, Yufei Liu, Ling Ji, and Hong Huang. Robust hyperspectral object tracking by exploiting background-aware spectral information with band selection network. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022. 2

[42] Qi Wang, Fahong Zhang, and Xuelong Li. Optimal clustering framework for hyperspectral band selection. *IEEE Transactions on Geoscience and Remote Sensing*, 56(10):5910–5922, 2018. 2

[43] Qingwang Wang, Liyao Zhou, Pengcheng Jin, Xin Qu, Hangwei Zhong, Haochen Song, and Tao Shen. Trackingmamba: Visual state space model for object tracking. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:16744–16754, 2024. 3

[44] Ye Wang, Shaohui Mei, Mingyang Ma, Yuheng Liu, Tao Gao, and Huiyang Han. Hyperspectral object tracking with context-aware learning and category consistency. *IEEE Transactions on Geoscience and Remote Sensing*, 2025. 2

[45] Yinan Wu, Licheng Jiao, Xu Liu, Fang Liu, Shuyuan Yang, and Lingling Li. Domain adaptation-aware transformer for hyperspectral object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 2, 5

[46] Zhehui Wu, Yong Chen, Naoto Yokoya, and Wei He. MP-HSIR: A multi-prompt framework for universal hyperspectral image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13009–13020, 2025. 2

[47] Jinxia Xie, Bineng Zhong, Zhiyi Mo, Shengping Zhang, Liangtao Shi, Shuxiang Song, and Rongrong Ji. Autoregressive queries for adaptive tracking with spatio-temporal transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19300–19309, 2024. 5

[48] Fengchao Xiong, Jun Zhou, and Yuntao Qian. Material based object tracking in hyperspectral videos. *IEEE Transactions on Image Processing*, 29:3719–3733, 2020. 2, 5

[49] F Xiong, J Zhou, W Charle, Y Zhong, P Ghamisi, and J Chanussot. The hyperspectral object tracking challenge (hot2023), 2023. 5

[50] Fengchao Xiong, Zhenxing Wu, Jun Zhou, Sen Jia, and Yuntao Qian. Suit: Spatial-spectral union-intersection interac-

tion network for hyperspectral object tracking. *IEEE Transactions on Image Processing*, 34:7786–7800, 2025. 3

[51] Jinyu Yang, Zhe Li, Feng Zheng, Ales Leonardis, and Jingkuan Song. Prompting for multi-modal tracking. In *Proceedings of the 30th ACM international conference on multimedia*, pages 3492–3500, 2022. 3

[52] Rui Yao, Lu Zhang, Yong Zhou, Hancheng Zhu, Jiaqi Zhao, and Zhiwen Shao. Hyperspectral object tracking with dual-stream prompt. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–12, 2024. 3, 7

[53] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision*, pages 341–357. Springer, 2022. 1, 5, 7, 8

[54] Florence Yellin, Scott McCloskey, Cole Hill, Eric Smith, and Brian Clipp. Concurrent band selection and traversability estimation from long-wave hyperspectral imagery in off-road settings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7483–7492, 2024. 2

[55] Jiacheng Ying, Hui-Liang Shen, and Si-Yuan Cao. Unaligned hyperspectral image fusion via registration and interpolation modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021. 2

[56] Yifan Zhang, Xu Li, Baoguo Wei, Lixin Li, and Shigang Yue. A fast hyperspectral tracking method via channel selection. *Remote Sensing*, 15:1557, 2023. 2

[57] Yaozong Zheng, Bineng Zhong, Qihua Liang, Zhiyi Mo, Shengping Zhang, and Xianxian Li. ODTrack: Online dense temporal token learning for visual tracking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38:7588–7596, 2024. 5, 6, 7, 8, 12, 14, 15

[58] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International journal of computer vision*, 130(9):2337–2348, 2022. 3

[59] Jiawen Zhu, Simiao Lai, Xin Chen, Dong Wang, and Huchuan Lu. Visual prompt multi-modal tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9516–9526, 2023. 2, 5, 7

Latent Spectral Prompting for Multimodal Hyperspectral Object Tracking

Supplementary Material

Wangshu Cai[1] Jun Zhou[2*] Lei Wang[1*]
Mohammad Aminul Islam[3] Xu Han[1] Kuldip Paliwal[1]

[1]Griffith University, Australia, [2]East China Normal University, China,
[3]Bangladesh Agricultural University, Bangladesh
{wangshu.cai, xu.han2}@griffithuni.edu.au, {l.wang4, k.paliwal}@griffith.edu.au,
amin@bau.edu.bd, jzhou@ee.ecnu.edu.cn

[*]Corresponding authors.

A. Dataset Details

Table 6. Dataset statistics.

Dataset	Domain	Split	Band	#Seqs	#Images
HOT2020	VIS	Train	16	40	12,887
HOT2020	VIS	Test	16	35	16,461
HOT2023	VIS	Train	16	55	19,251
HOT2023	VIS	Test	16	46	20,074
HOT2023	NIR	Train	25	40	8,239
HOT2023	NIR	Test	25	30	4,872
HOT2023	RedNIR	Train	15	15	6,412
HOT2023	RedNIR	Test	15	11	3,500

We summarize the statistics of the hyperspectral object tracking datasets used in this paper in Tab. 6. Both HOT2020 and HOT2023 contain diverse scenes including laboratories, urban streets, and outdoor environments, with image resolutions varying up to 1080p.

A notable imbalance exists across spectral domains. The VIS domain provides 55 training sequences with 19,251 frames in HOT2023, whereas NIR offers 40 sequences (8,239 frames) and RedNIR contains only 15 sequences (6,412 frames). This data scarcity in NIR and RedNIR domains directly impacts the SSAE's ability to learn robust spectral representations, as the self-supervised reconstruction objective benefits from diverse training samples. Furthermore, the NIR (25 bands) and RedNIR (15 bands) domains capture wavelength ranges that differ substantially from the visible spectrum on which the RGB foundation model was pretrained. This introduces a mismatch to VIS, making cross-modal adaptation inherently more challenging. These factors jointly explain the performance variation across domains observed in the main paper (Tab. 1): our method achieves the largest improvements in VIS where sufficient training data is available, while gains in RedNIR and NIR are more modest due to data scarcity, different false-color quality, and the greater spectral distance from the RGB pretraining domain.

B. Additional Qualitative Results

We provide additional qualitative comparisons on difficult sequences across different spectral domains in Fig. 7.

Results further confirm that SProTrack maintains robust tracking under diverse spectral conditions. Comparing with DaSSPNet[26], our method provides more robustness to small objects (row 1 and row 2), distractors (row 3), and occlusion (row 1 and row 4). Comparing with our backbone ODTrack[57], our method is more accurate, showing more concentration of scores to the objects of interest.

C. Implementation Details

This section provides our implementation details.

SSAE. The encoder maps the B raw bands to 64 channels with a 1×1 convolution, applies three dual-branch blocks, and projects to the 3-channel latent with a second 1×1 convolution; the decoder mirrors this structure. Each dual-branch block contains (i) a spatial branch, a depthwise 3×3 convolution with groups $=64$, and (ii) a spectral-spatial branch that tokenises the feature map into 8×8 patches, projects each flattened patch ($8\times8\times64=4096$) to 512 with a linear layer, applies Mamba (d_{model}=512, d_{state}=16, d_{conv}=4) along the resulting spatial token sequence, and projects back to 4096. The two branches are concatenated and fused by a 1×1 convolution ($128\rightarrow64$). B is 16 for HOT2020 and HOT2023-VIS, 25 for HOT2023-NIR and 15 for HOT2023-RedNIR, corresponding to their band numbers.

Prompt branch. We follow the ViPT-deep design with one prompt block per transformer layer (12 in total). Each block reduces the RGB and latent token maps from 768 to 8 channels with two 1×1 convolutions and projects the fused result back to 768 with a third 1×1 convolution; a learnable scalar smooths the fovea weighting. The latent is embedded by a separate patch embedding of shape $768\times3\times16\times16$, and each layer carries its own LayerNorm (12×768).

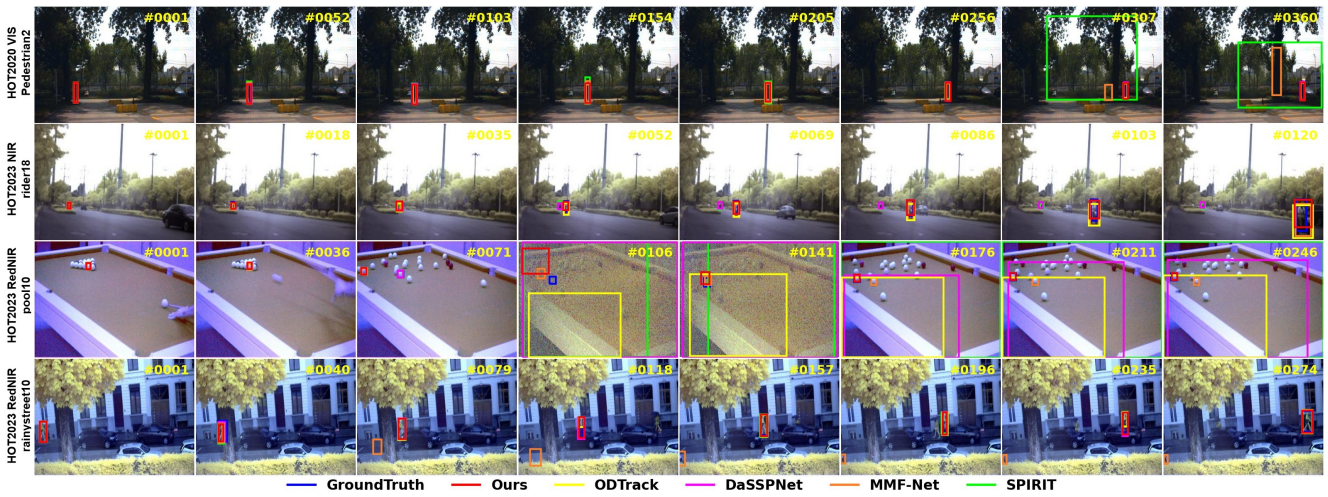


Figure 7. Additional qualitative comparison on four challenging sequences. The dataset name and sequence names are labeled on the left. Our method produces more concentrated responses and more accurate localization under occlusion, distractors, and bad lighting.

Table 7. Per-dataset training settings.

Dataset	Batch	Epochs	LR drop	LR
HOT2020-VIS	16	60	50	8×10^{-5}
HOT2023-VIS	16	60	50	8×10^{-5}
HOT2023-NIR	16	40	–	8×10^{-5}
HOT2023-RedNIR	16	40	–	8×10^{-5}

LoRA. LoRA is applied only to the two linear layers of each dual-branch block in the autoencoder encoder (6 layers, 12 tensors), with rank $r = 2$, $\alpha = 4$, and dropout 0.1. The tracker backbone and head contain no LoRA and remain frozen. With a 3-channel latent, the model has 894,924 trainable parameters in total, comprising 55,296 LoRA parameters, 249,036 prompt-module parameters, and 590,592 projection parameters.

Tracker and optimisation. The base tracker is ODTrack with a ViT-B/16 pre-trained backbone, candidate elimination at layers $\{3, 6, 9\}$ with keep ratio 0.7, search region 384×384 and three 192×192 template frames. We use AdamW (wd $= 10^{-4}$), a step schedule with decay 0.1, a backbone learning-rate multiplier of 0.1, drop-path 0.1, and loss weights $\lambda_1 = 5$, $\lambda_2 = 2$. All runs use 7,200 sampled pairs per epoch.

Frequency loss implementation. DFT is evaluated in `fp32`: inputs are cast from `fp16` before the transform to avoid overflow in the accumulation. We use `torch.fft.fft2` with `norm="ortho"`, which applies the transform over the two trailing (spatial) axes of each band and yields the orthonormal scaling under which Parseval's identity would hold for the residual spectrum. Both loss terms use mean reduction.

D. Efficiency Details

This section provides memory and training-cost measurements complementing the efficiency analysis in the main paper.

Parameter efficiency. Although the full model contains 111.46M parameters, only 0.895M (0.80%) are trainable, while the remaining 99.2% (110.56M) are frozen. The trainable parameters consist solely of (i) LoRA modules inserted into the projection-in/projection-out layers of the SSAE, and (ii) the prompt modules and projection layers. The backbone, SSAE encoder, and prediction head are entirely frozen. We clarify the role and scope of LoRA, as it bears on our decoupling claim. The temporal axis is fully decoupled: the entire temporal backbone and prediction head are frozen, and no temporal parameter is updated by hyperspectral supervision. On the spectral side, the SSAE is pretrained self-supervised and its globally learned spectral representation is not re-trained for tracking; LoRA is applied only as a lightweight, task-conditioned readout on the SSAE projection layers, whose purpose is to extract subject-relevant spectral cues from the otherwise scene-level representation and route them into the prompt. Decoupling therefore holds in the sense that matters: temporal modeling and spectral representation learning are optimized in separate stages and never jointly, with only a sub-0.5M readout bridging them.

Memory usage. At inference (batch size 1, no gradients, fp32), the model requires only 0.76 GiB of peak allocated memory and 0.94 GiB reserved; including the CUDA context ($\sim$0.5 GiB), the process occupies roughly 1.3 GiB. The model is therefore extremely lightweight to deploy, fitting comfortably on any modern GPU. During training (batch size 16, forward + backward + optimizer step), peak allo-

Table 8. Two-stage training cost on a single RTX 5090.

Stage	Per epoch	Epochs	Supervision
SSAE	2.2 min	150	Self-supervised
Tracker	13.9 min	60	Tracking labels

cated memory is 15.24 GiB and peak reserved is 23.25 GiB. The footprint is dominated by activations rather than parameters or optimizer state: since only 0.80% of parameters are trainable, the AdamW optimizer states and gradients are negligible, and the 15.24 GiB of allocated memory is almost entirely forward activations at batch size 16. We adopt a per-frame autoregressive backward pass that releases activations frame by frame, following ODTrack[57], avoiding the roughly 3× activation memory that retaining all three search frames' computation graphs simultaneously would incur.

Training cost. SProTrack is trained in two stages, both lightweight and run on a single GPU (Tab. 8). Stage 1 pretrains the SSAE in a fully self-supervised manner without annotations, at 2.2 minutes per epoch (∼2.88 batches/s) for 150 epochs. Stage 2 trains only the prompt and projection modules for tracking, at 13.9 minutes per epoch (batch size 16, 7,200 sampled pairs per epoch, using three 384×384 search frames and three 192×192 template frames) for 60 epochs on the VIS domain. Because the SSAE is pretrained once and transfers across datasets and spectral domains, and the heavy backbone is reused rather than retrained, the overall training cost remains modest despite the model's total size.

E. Failure Case Analysis

While SProTrack achieves strong overall performance, we identify three systematic failure modes, illustrated in Fig. 8. These limitations are common among hyperspectral trackers that rely on visual backbones pretrained in the RGB domain.

Target-distractor interaction. When the target physically interacts with or passes closely behind a visually similar object, the predicted bounding box tends to shift toward the distractor, particularly when the distractor is larger or more salient. The score map produces a bimodal response during the interaction period, and the box regression head resolves this ambiguity by averaging the two peaks, resulting in a spatially offset prediction. Although the temporal token facilitates recovery after separation, the transient drift during interaction degrades frame-level metrics.

Prolonged full occlusion. Unlike the partial occlusion in Tab. 3, here we refer to prolonged full occlusion. When the target is fully occluded for an extended duration (e.g., >20 consecutive frames), the temporal token gradually accumulates stale appearance information, causing the tracker to either freeze at the last known position or drift toward a nearby distractor with similar appearance. This limitation is inherent to the frozen ODTrack[57] backbone, which lacks an explicit re-detection mechanism.

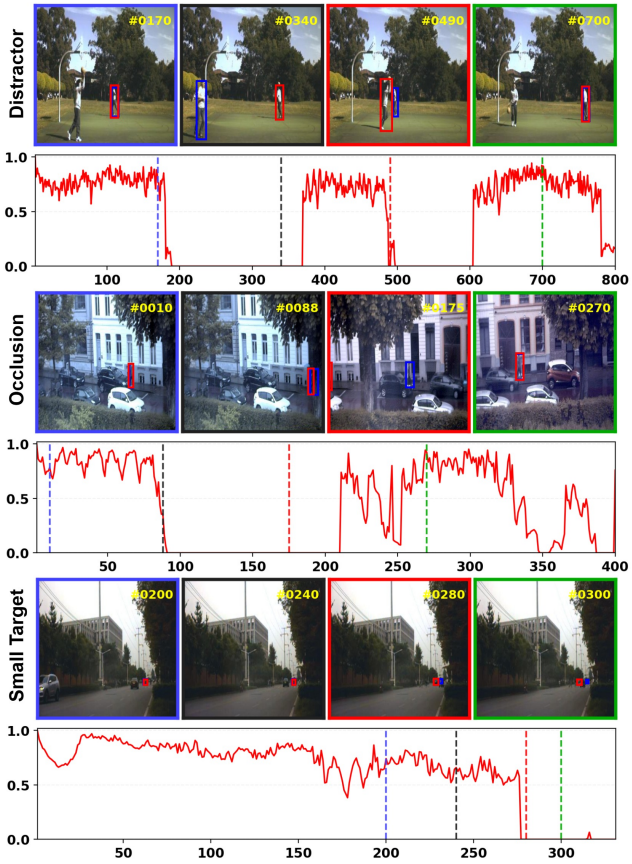


Figure 8. Examples of failure cases and the IoU (overlap ratio) between predictions and ground truth. The colored dashed lines are corresponding to the frames above with borders in the same colors. Dashed blue boxes denote ground truth; red boxes are our predictions. Row 1 (Distractor): target-distractor interaction; Row 2 (Occlusion): prolonged full occlusion; Row 3 (Small Target): extremely small target.

Extremely small targets. For targets occupying fewer than approximately 100 pixels, both the false-color image and the SSAE latent representation lack sufficient spatial resolution to encode discriminative spectral features. In such cases, the spectral prompt contributes minimal additional information, and our method degrades to the performance level of the baseline ODTrack[57].

Analysis. We stress that these failure modes are inherited from the frozen RGB backbone (lack of re-detection) or from input resolution limits (small targets), not from the spectral representation itself. In the appearance-variation and motion-dynamics regimes where spectral cues are most

Table 9. Paired per-sequence comparison on HOT2020. Differences in DP@20 are taken within each sequence, which cancels the difficulty term shared by all trackers. CI is a percentile bootstrap over sequences (10,000 resamples); p is a two-sided Wilcoxon signed-rank test. Ties are frequent because DP@20 saturates at 1.0 on easy sequences, so we report win/tie/loss (W/T/L) counts.

Method	ΔDP	95% CI	p	W/T/L
vs. MMF-Net [24]	$+0.0212$	$[-0.0137, +0.0546]$	0.114	14/12/9
vs. SPIRIT [6]	$+0.0283$	$[-0.0126, +0.0707]$	0.123	13/13/9
vs. SEE-Net [23]	$+0.0199$	$[-0.0163, +0.0572]$	0.108	14/15/6
vs. DaSSPNet [26]	$+0.0357$	$[-0.0049, +0.0907]$	0.012	17/13/5

informative, deformation, occlusion onset, or illumination change, SProTrack ranks first (Tab. 3), confirming that the spectral advantage holds precisely where it is expected to, while the residual failures lie outside the spectral module's scope.

F. Statistical Reliability Analysis

HOT2020 contains only 35 test sequences, and sequence difficulty varies far more than method quality does. Marginal per-sequence standard deviations are therefore dominated by a variance component shared across all trackers and overlap heavily even between clearly different methods. We instead report a paired analysis (Tab. 9), taking DP differences *within* each sequence against hyperspectral baselines, with a percentile bootstrap 95% CI and a two-sided Wilcoxon signed-rank test. SProTrack attains a higher mean paired DP against every baseline and wins on more sequences than it loses in all four comparisons; the comparison against the strongest baseline, DaSSPNet, is significant ($p = 0.012$). Power is limited here for two reasons: DP@20 saturates at 1.0 on easy sequences, leaving 12–15 ties per comparison, and one outlier: a small-target failure of the kind costs 0.32–0.37 DP against every baseline. All intervals include zero while the signed-rank test rejects for DaSSPNet; this is expected, since the interval is computed on the mean, which a few large negative differences shift, whereas the signed-rank test reflects the direction of the differences.

G. Limitations

SProTrack has several limitations. First, its higher MACs and parameter count, inherited mainly from the frozen ODTrack[57], may limit deployment on constrained hardware. Nevertheless, only 0.80% of parameters are trainable, and template latent caching enables real-time inference (43.1 FPS), trading static cost for stronger temporal priors under data scarcity. Second, we train a separate SSAE for each spectral domain rather than jointly across VIS, NIR, and RedNIR; cross-domain adaptation, as explored by DaSSPNet[26], remains future work. Finally, the frozen backbone lacks explicit re-detection, causing failures under prolonged occlusion, while spectral prompting provides limited benefit for extremely small targets.

LLM Usage Declaration

Large language models were used solely as writing assistance tools during the preparation of this manuscript. Specifically, they were used for grammar correction, sentence rephrasing, and improving the clarity of English expressions.

All scientific contributions, including the research idea, method design, experimental setup, analysis, and interpretation of results, were conceived and conducted entirely by the human authors. No large language model was used to generate novel content, and none is considered an author or co-creator of this work.