UNIFLOW-RS: UNIFIED GENERATIVE PREDICTION FOR HETEROGENEOUS REMOTE-SENSING TASKS

Anonymous authors
Paper under double-blind review

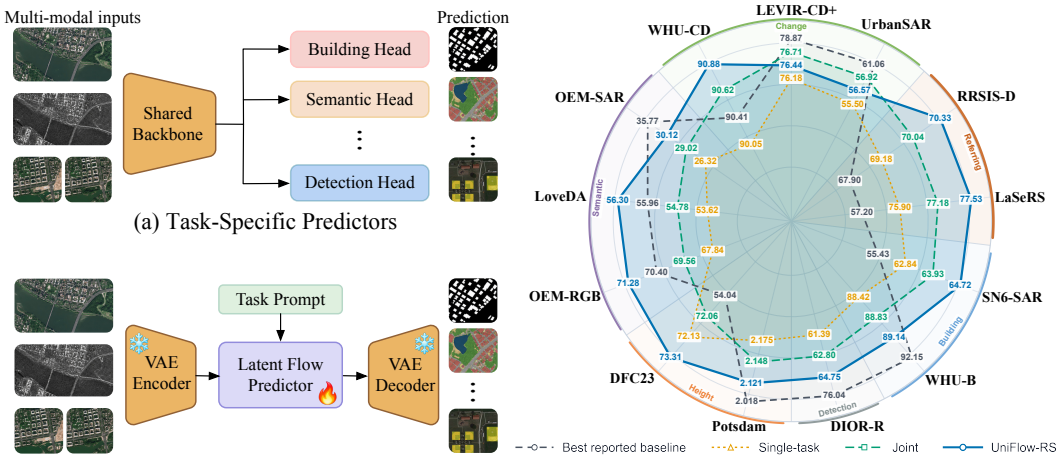


Figure 1: UniFlow-RS unifies heterogeneous remote-sensing perception through a shared generative prediction process. Unlike prior multi-task models that share representations but retain task-specific predictors, UniFlow-RS uses a single generative predictor across six tasks and achieves competitive performance on thirteen datasets.

ABSTRACT

Remote-sensing foundation models increasingly unify representations across sensing modalities and downstream tasks, yet their prediction mechanisms remain largely task-specific. Separate decoders, prediction heads, and objectives leave prediction fragmented across tasks. This naturally raises a question: *can heterogeneous remote-sensing tasks share the prediction process?* **UniFlow-RS** answers this question with a single latent flow-matching predictor shared across tasks. This requires addressing two key challenges: *unifying heterogeneous prediction spaces* and *harmonizing joint optimization across tasks*. First, we cast heterogeneous task outputs into structured visual targets in a common latent space, allowing them to be modeled by the same pretrained flow predictor. Task-dependent conditioning further accommodates differences in input configurations. Second, we optimize all tasks with a common Flow Matching objective, eliminating task-specific prediction losses. To mitigate optimization mismatch across tasks and modalities, we introduce Task–Modality Adaptive Timestep Sampling (TMAS), which dynamically allocates training timesteps according to the online learning difficulties of each task–modality group. Using a single shared model, UniFlow-RS handles six tasks and thirteen datasets, surpassing recent task-specific state-of-the-art methods on several benchmarks while remaining competitive on the others. The same checkpoint also transfers strongly to two unseen datasets, demonstrating generalization ability of the unified model.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Remote-sensing interpretation increasingly requires models to operate across diverse tasks, sensing modalities, and input configurations. Discriminative foundation models such as SkySense and MTP (Guo et al., 2024; Wang et al., 2024a) have substantially advanced representation reuse across optical, SAR, multi-temporal observations, and downstream tasks. However, as illustrated in Figure 1(a), most unified models still retain task-specific prediction heads, decoders, or output mechanisms after the shared representation, leaving the final prediction stage fragmented across tasks. This exposes a remaining gap in unified remote-sensing perception: *can heterogeneous remote-sensing tasks share not only representations, but the prediction process itself?*

Generative foundation models provide a natural route toward this goal by reformulating prediction as conditional visual generation. Recent work has shown that pretrained image generators can serve as generalist visual predictors (Gabeur et al., 2026), while flow matching provides a simple formulation for conditional generation through velocity-based transport (Lipman et al., 2023; Liu et al., 2023). In remote sensing, generative models have further demonstrated strong flexibility across sensing modalities and generation settings (Jakubik et al., 2025; Chen et al., 2026). This suggests a unified paradigm for generalist remote-sensing perception: encoding diverse structured outputs as visual representations and predicting them with a shared latent flow model.

Making this paradigm work in remote sensing requires resolving two key mismatches. The first is a *prediction-space and conditioning mismatch*. Remote-sensing tasks involve heterogeneous outputs, from binary masks and semantic maps to height fields, referred regions, and rotated instances, with inputs spanning optical/SAR, single-/bi-temporal, and optional text conditioning. A unified predictor must therefore accommodate these heterogeneous output structures and input conditioning configurations without resorting to task-specific branches. The second is an *optimization mismatch*. Even under a unified prediction formulation, different task–modality groups can exhibit different learning dynamics along the flow trajectory. A fixed timestep distribution imposes the same sampling schedule on groups whose optimization needs may differ substantially.

We introduce **UniFlow-RS**, a latent-flow framework that addresses these two mismatches with a single generative predictor shared across heterogeneous remote-sensing tasks. As shown in Figure 1(b), UniFlow-RS replaces a collection of task-specific predictors with one pretrained latent flow predictor and one shared LoRA (Hu et al., 2022) adapter, together with a frozen VAE for target encoding and decoding. All tasks are trained with the same Flow Matching objective, without learned task-specific output heads or task-specific losses. To make this shared predictor applicable across tasks, UniFlow-RS maps task-specific structured targets into a common VAE-compatible visual latent space, while different tasks are specified through task-dependent input conditioning. We further design deterministic recovery rules to convert the generated latent outputs back into their native evaluation formats. Building on this unified prediction space, we address the heterogeneous optimization dynamics across different tasks with Task–Modality Adaptive Timestep Sampling (TMAS). TMAS dynamically reallocates timestep-level training budget according to the online learning state of each task–modality group, allowing the sampling schedule to adapt to group-specific optimization needs.

As shown in Figure 1(c), UniFlow-RS covers six remote-sensing tasks across thirteen datasets, spanning optical, SAR, bi-temporal, continuous-output, language-conditioned, and instance-level prediction settings. Despite using a shared LoRA adapter, UniFlow-RS surpasses the best reported results on multiple benchmarks while remaining competitive on the others.

Our contributions are threefold:

- We introduce a unified remote-sensing framework in which heterogeneous tasks share the *prediction process itself*, rather than only representations. A single pretrained latent flow predictor and shared LoRA support six tasks under the same Flow Matching objective, without learned task-specific output heads or losses.
- We formulate heterogeneous remote-sensing outputs in a common structured visual space and combine it with task-dependent conditioning, allowing diverse output structures, sensing modalities, and input configurations to be handled by one latent predictor.
- We formulate heterogeneous task–modality learning dynamics along the flow trajectory as a timestep-allocation problem and propose TMAS, which adaptively adjusts timestep sampling based on the online learning state of each group, improving joint optimization and convergence.

2 Related Work

Unified remote-sensing models. Remote-sensing foundation models have progressively expanded from single-modality representation learning toward multi-modal and multi-task unification. SkySense and MTP (Guo et al., 2024; Wang et al., 2024a) establish shared representations across heterogeneous observations and downstream tasks, while SkySense V2 (Zhang et al., 2025) further improves multi-modal representation sharing through modality-aware architectural design. SelectiveMAE (Wang et al., 2025) scales masked pretraining, while FedSense (Tan et al., 2025) studies distributed foundation-model learning. S5 and RingMoE (Lv et al., 2026; Bi et al., 2026) further explore multi-dataset and modality-specialized adaptation. To handle heterogeneous output spaces and evaluation protocols, these models typically introduce task- or dataset-specific prediction modules after the shared representation. In contrast, we study a stronger form of unification: whether heterogeneous remote-sensing tasks can share the prediction process itself, including the latent output space, generative predictor, training objective, and image decoder.

Generative models for structured perception. Generative models have increasingly been explored for structured prediction beyond image synthesis. Painter (Wang et al., 2023) represents heterogeneous vision outputs as images and learns multiple tasks through masked image modeling. InstructDiffusion (Geng et al., 2024) and InstructCV (Gan et al., 2024) further investigate instruction-conditioned generation for vision tasks, while FlowDIS (Sargsyan & Navasardyan, 2026) applies Flow Matching to language-guided segmentation. These studies show that generative formulations can support diverse vision prediction tasks. However, whether generative prediction can serve as a unified interface for heterogeneous remote-sensing perception remains unclear. Remote-sensing tasks involve diverse output structures, sensing modalities, and input configurations, requiring a prediction framework beyond existing task-specific formulations. We address this question by casting heterogeneous outputs into structured visual targets and enabling a latent flow predictor to handle binary masks, semantic maps, continuous fields, and rotated instances within one prediction process.

Adaptive optimization for heterogeneous joint training. Heterogeneous multi-task learning often exhibits imbalanced optimization dynamics across tasks. GradNorm (Chen et al., 2018b) adapts task weights by balancing gradient magnitudes, while PIVRG (Qin et al., 2025) further addresses optimization imbalance through performance-informed gradient aggregation. In parallel, generative-model training has explored non-uniform timestep allocation. Improved DDPM (Nichol & Dhariwal, 2021) introduces loss-aware timestep sampling, while Adaptive Non-Uniform Timestep Sampling (Kim et al., 2025) further prioritizes diffusion timesteps according to their online optimization difficulty. Our problem lies at the intersection of these two directions. Unlike task reweighting or a single global timestep sampler, TMAS adapts the timestep distribution separately for each task–modality group while keeping the task and dataset mixture fixed.

3 UniFlow-RS

In this section, we introduce the latent flow formulation (Sec. 3.1), then specify the target representations and task-dependent conditioning (Sec. 3.2). Finally, we present TMAS, which adapts timestep sampling for each task–modality group (Sec. 3.3). Figure 2 summarizes the framework.

3.1 Unified latent flow formulation

We build on a pretrained latent flow model and jointly train a single shared LoRA adapter across all tasks. The pretrained parameters of the image VAE, text encoder, and MMDiT backbone remain frozen. Let E, D, and T denote the VAE encoder, VAE decoder, and text encoder, respectively, and let θ denote the trainable LoRA parameters.

For a target y from task τ and dataset d, a deterministic rendering operator $C_{\tau,d}$ converts y into a structured visual target. Its latent representation is $z_y = E(C_{\tau,d}(y))$, which allows all tasks to share the same latent prediction space without task-specific neural output heads. We construct the source latent z_s from the input observations and connect it to the target latent z_y through a linear path:

$$z_t = (1-t)z_y + tz_s, \qquad v^\star = z_s - z_y, \qquad t \in [0,1]. \tag{1}$$

The visual condition c_I and text condition c_T are detailed in Sec. 3.2.

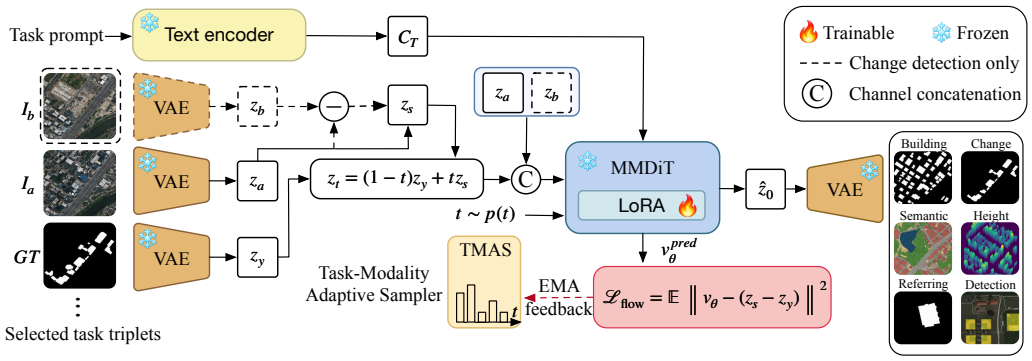


Figure 2: Overview of UniFlow-RS. Native task outputs are rendered as structured visual targets and encoded by a frozen VAE. A pretrained MMDiT with one shared trainable LoRA predicts the conditional velocity field, while TMAS adapts timestep sampling from online Flow Matching errors. At inference, transport starts from the task-dependent source latent; the shared VAE decoder and deterministic recovery operators then produce native task predictions.

The shared MMDiT predicts the velocity $v_\theta(z_t, c_I, c_T, t)$ with the standard flow matching objective

$$\mathcal{L}_{\text{flow}} = \mathbb{E}\left[\|v_\theta(z_t, c_I, c_T, t) - v^\star\|_2^2\right]. \tag{2}$$

At inference, we solve $\frac{\mathrm{d}z_t}{\mathrm{d}t} = v_\theta(z_t, c_I, c_T, t)$ from $t = 1$ to $t = 0$ with $z_1 = z_s$, obtaining $\hat{z}_y = z_0$. The native prediction is recovered as $\hat{y} = R_d(D(\hat{z}_y))$ where R_d is the deterministic operator to convert the visual output to its native evaluation format. All tasks therefore share the latent space, velocity predictor, trainable adapter, and the frozen VAE encoder and decoder. Appendix C.2 visualizes the intermediate transport states across all six tasks.

3.2 Structured Visual Targets and Task-Dependent Conditioning

Structured visual targets. Remote-sensing tasks have substantially different native outputs. Building extraction and referring segmentation produce binary masks; change detection includes both binary building-change masks and categorical flood-change maps; semantic segmentation produces categorical maps; height estimation predicts continuous values; and rotated detection returns oriented instances.

We encode these heterogeneous outputs as visual targets, *i.e.,* images, using $C_{\tau,d}$. Binary outputs are represented as two-tone masks, categorical outputs as class-coded maps, height values through a log-compressed grayscale encoding, and rotated instances as class-coded canvases, from which the native instances are recovered via deterministic geometric post-processing. When datasets within the same task follow different label ontologies, $C_{\tau,d}$ first aligns their labels to a shared semantic definition before visual encoding.

These conversions are deterministic and non-learned, providing a common interface to the pretrained image VAE. Detailed palettes, height transformations, ontology mappings, recovery procedures, and frozen-VAE round-trip analyses are provided in Appendix B.

Task-dependent conditioning. Most tasks take a single image as input, whereas change detection uses a pair of temporal observations and referring segmentation additionally requires a language query. We accommodate these task-dependent input configurations through the source state and conditioning inputs of the same flow model. Let I_a denote the current observation and $z_a = E(I_a)$ its latent representation. We use $c_T = T(q)$ to encode a fixed task instruction or, for referring segmentation, the sample's original expression (Appendix A.1). For bi-temporal tasks, the previous observation I_b is additionally encoded as $z_b = E(I_b)$. The source state and visual condition are

$$[z_s, c_I] = \begin{cases} [z_a,\ [\mathbf{0}, z_a]], & \text{single-image tasks,} \\ [z_a - z_b,\ [z_b, z_a]], & \text{bi-temporal tasks,} \end{cases} \tag{3}$$

where $[\cdot,\cdot]$ denotes channel-wise concatenation, and $\mathbf{0}$ has the same shape as z_a. The source state z_s initializes the flow, while c_I provides visual context. We use z_a as the source for single-image tasks

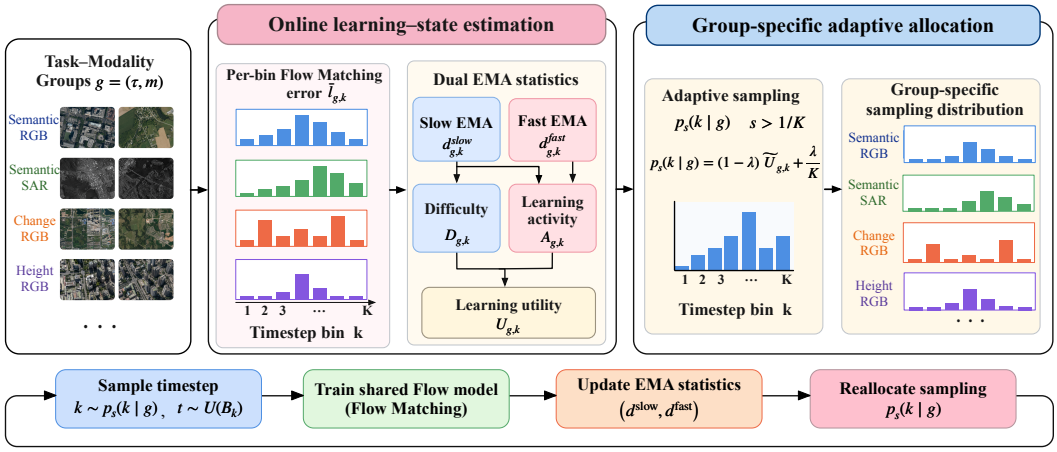


Figure 3: Task–Modality Adaptive Timestep Sampling. Each group is indexed by task and modality. TMAS tracks group-specific Flow Matching error at two timescales, converts the resulting learning state into a sampling utility, and adjusts the timestep distribution after uniform warm-up. A persistent uniform component ensures continued exploration across the flow trajectory.

to preserve input structure, and $z_a - z_b$ for change detection while conditioning on $[z_b, z_a]$ to retain both observations. We validate these choices in Sec. 4.3.

3.3 Task–Modality Adaptive Timestep Sampling

The formulation above unifies heterogeneous tasks under a shared latent predictor and Flow Matching objective, but it does not eliminate differences in their optimization dynamics. In particular, different tasks and modalities may exhibit distinct learning difficulties along the flow trajectory due to differences in output structure and sensing characteristics. A single global timestep distribution may oversample regions that are already well learned for some groups while undersampling more challenging regions for others.

A simple solution is to sample more often from regions with larger prediction errors. However, error magnitude alone does not tell us how a region is being learned. A high-error region may have plateaued, whereas a moderate-error region may still be changing rapidly. We therefore use two signals: *persistent difficulty*, which describes where a group remains difficult, and *learning activity*, which describes where its error is still changing. TMAS estimates these signals for each task–modality group and uses them to adapt timestep sampling online. Figure 3 gives an overview.

Online learning-state estimation. We define each task–modality group as $g = (\tau, m)$, where τ denotes the task and m the physical modality. Datasets with the same task and modality share one sampling profile. We divide $[0, 1]$ into K equal-width bins and let $\bar{\ell}_{g,k}$ be the mean normalized velocity-prediction error observed for group g at bin k.

To characterize both persistent difficulty and recent learning dynamics, we maintain fast and slow exponential moving averages (EMAs) of the bin-wise error:

$$d^{\text{fast}}_{g,k} \leftarrow \left(1 - \frac{1}{K}\right) d^{\text{fast}}_{g,k} + \frac{1}{K}\bar{\ell}_{g,k}, \quad d^{\text{slow}}_{g,k} \leftarrow \left(1 - \frac{1}{K^2}\right) d^{\text{slow}}_{g,k} + \frac{1}{K^2}\bar{\ell}_{g,k}. \tag{4}$$

The slow EMA captures the long-term difficulty of a bin, while the fast EMA is more sensitive to recent updates. Their discrepancy therefore reflects ongoing learning dynamics: $d^{\text{fast}}_{g,k} < d^{\text{slow}}_{g,k}$ suggests recent improvement, while $d^{\text{fast}}_{g,k} > d^{\text{slow}}_{g,k}$ reflects increasing error which can indicate forgetting or cross-task interference. The two timescales are determined directly by K rather than separately tuned. Implementation details including error normalization, initialization, aggregation, and unseen-bin handling are given in Appendix D.

Absolute Flow Matching errors can have different scales across task–modality groups. We therefore measure difficulty relative to other timestep bins of the same group. Before computing the sampling scores, we convert the raw EMA states into stable sampling-time estimates: unobserved bins are

assigned a conservative group-level fallback, and all estimates are lower-bounded by ϵ for numerical stability. We denote the resulting estimates by $\tilde{d}^{\text{slow}}_{g,k}$ and $\tilde{d}^{\text{fast}}_{g,k}$; the exact fallback rule is given in Appendix D. We then define

$$D_{g,k} = \frac{\tilde{d}^{\text{slow}}_{g,k}}{\frac{1}{K}\sum_{j=1}^{K} \tilde{d}^{\text{slow}}_{g,j}}, \quad A_{g,k} = \min\left\{ \left| \frac{\tilde{d}^{\text{slow}}_{g,k} - \tilde{d}^{\text{fast}}_{g,k}}{\tilde{d}^{\text{slow}}_{g,k}} \right|, 1 \right\}. \tag{5}$$

Here, $D_{g,k} > 1$ indicates that bin k is more difficult than the group average, while $A_{g,k}$ measures how strongly its error is changing relative to its long-term level. We use the absolute fast–slow discrepancy because both rapid improvement and rapid degradation indicate active learning dynamics.

We combine them into

$$U_{g,k} = (D_{g,k} + \epsilon)\left(\frac{1}{K} + A_{g,k}\right), \tag{6}$$

where $\epsilon > 0$ ensures numerical stability. Difficulty provides the base allocation, while activity gives additional weight to bins whose learning state is changing. The offset $1/K$ ensures that a difficult but temporarily stable bin still receives training budget when the fast and slow estimates are close.

Group-specific adaptive allocation. The estimates are unreliable early in training because many group–bin pairs have only been observed a few times. We therefore use uniform timestep sampling for the first $1/K$ fraction of training. Let $s \in [0, 1]$ denote normalized training progress. The probability of sampling bin k for group g is

$$p_s(k \mid g) = \begin{cases} \dfrac{1}{K}, & s \le \dfrac{1}{K}, \\ \dfrac{\lambda}{K} + (1-\lambda)\dfrac{U_{g,k}}{\sum_{j=1}^{K} U_{g,j}}, & s > \dfrac{1}{K}, \end{cases} \tag{7}$$

where $\lambda \in (0, 1]$ controls the trade-off between uniform exploration and adaptive sampling. After selecting a bin, t is sampled uniformly within it. The uniform term λ/K prevents a bin that is underestimated early from being permanently neglected.

Importantly, TMAS only redistributes optimization effort across the flow trajectory within each task–modality group. It does not change the task or dataset mixture, the velocity target, or the per-sample Flow Matching objective.

4 Experiments

4.1 Experimental setup

We use the frozen FLUX.2 [klein] 4B Base MMDiT (Black Forest Labs, 2026), together with its frozen VAE and a frozen Qwen3-4B-FP8 text encoder (Qwen Team, 2025). A rank-32 LoRA with approximately 49.1M trainable parameters is shared across all tasks. We train for 60k optimizer steps with batch size 32 on four NVIDIA RTX 6000 Ada GPUs using AdamW and an initial learning rate of 5×10^{-5}, halved at 20k, 35k, and 50k steps. Unless otherwise specified, inference uses two flow steps. TMAS uses $K = 10$ timestep bins and exploration weight $\lambda = 0.2$; all remaining sampler coefficients are fixed analytically from K, with $\epsilon = 10^{-6}$ for numerical stability.

We evaluate six tasks across thirteen datasets, following the official splits and native metrics whenever available. We further assess zero-shot transfer on two unseen datasets, EarthReason and RISBench (Li et al., 2025c; Dong et al., 2024b), neither of which is used to train UniFlow-RS. RISBench mainly introduces a visual domain shift, whereas EarthReason introduces shifts in both image and language queries. Full details of the datasets, preprocessing, and evaluation metrics are provided in Appendix A. For controlled comparisons among unified model variants, we additionally report unified task score (UTS), which first averages the eligible native metrics within each task and then assigns equal weight to all six tasks. Its exact definition is provided in Appendix A.2. The frozen-VAE round-trip protocol used for representation analysis is described in Appendix B.

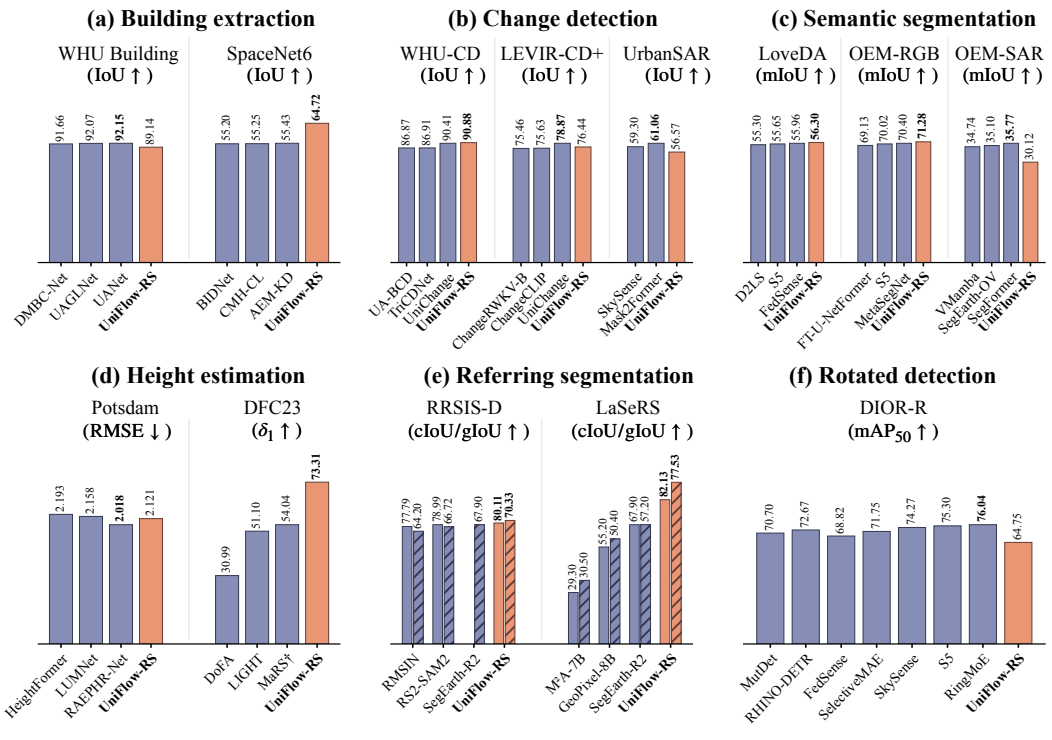


Figure 4: Main benchmark comparison across six tasks and thirteen datasets. Each panel reports the dataset-native metric indicated above the bars, with arrows showing the preferred direction. All UniFlow-RS results are obtained from the same shared checkpoint.

4.2 Main Results

Figure 4 compares the single jointly trained UniFlow-RS checkpoint with representative unified and task-specific baselines across all thirteen datasets. All UniFlow-RS results are obtained from the same pretrained flow backbone and shared LoRA, without task-specific fine-tuning or learned output heads. Full numerical comparisons, venue–year information, and dataset-specific notes are provided in Appendix F (Tables 14 and 15).

Despite using a single shared predictor, UniFlow-RS remains competitive across all six tasks and achieves the best results in our comparison on WHU-CD (Ji et al., 2019), LoveDA (Wang et al., 2021), OpenEarthMap-RGB (Xia et al., 2023), SpaceNet 6 SAR (Shermeyer et al., 2020), DFC23 (Liu et al., 2024a), RRSIS-D (Liu et al., 2024b), and LaSeRS (Xin et al., 2026). These results across diverse modalities and output structures demonstrate the flexibility of a unified generative predictor.

The task-wise results also reflect different compatibility with the unified prediction interface. Referring segmentation benefits from the shared dense prediction prior and explicit language conditioning, making its binary-mask output well aligned with conditional latent transport. In contrast, rotated detection requires recovering discrete instances and precise oriented geometry from a generated class-coded canvas. Without task-specific box or orientation losses, small generation errors can be amplified during deterministic instance recovery, making detection more challenging.

Qualitative results. Figure 5 compares the shared UniFlow-RS predictor with task-specific baselines (Kang et al., 2022; Shabbir et al., 2025; Lv et al., 2026; Chen et al., 2023; Cheng et al., 2022a) across six tasks. Compared with these baselines, UniFlow-RS produces cleaner masks with fewer spurious regions, more coherent semantic boundaries and rotated-instance layouts, and sharper building footprints with height variation. In change detection, UniFlow-RS recovers more change regions while suppressing isolated responses. These examples illustrate that the shared predictor can preserve task-specific output structures without task-specific heads. Additional qualitative comparisons across all six tasks are provided in Appendix G.

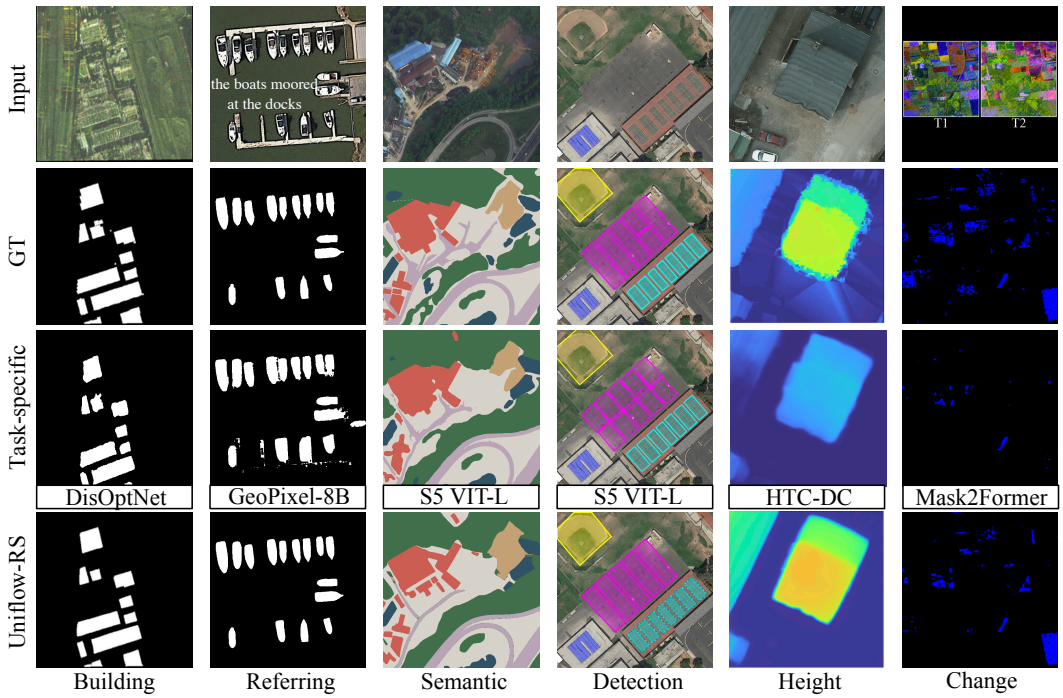


Figure 5: Qualitative comparison of task-specific baselines and UniFlow-RS across six tasks. Baseline names are shown below the corresponding task-specific predictions.

4.3 Analysis

Table 1: Comparison of task-specific and joint adaptation. Single-task uses one LoRA per task, whereas the joint variants share one LoRA.

Setting	Building IoU ↑	Change IoU ↑	Semantic mIoU ↑	Height δ_1 ↑	Ref. gIoU ↑	Det. mAP50 ↑	UTS ↑
Single-task	75.63	73.91	49.26	72.13	72.54	61.39	67.48
Joint	76.38	**74.75**	51.12	72.06	73.61	62.80	68.45
Joint + TMAS	**76.93**	74.63	**52.57**	**73.31**	**73.93**	**64.75**	**69.35**

Joint training vs. task-specific training. Table 1 compares a single jointly trained model with six independently trained task-specific models, all using the same Fixed Beta sampler. Joint training increases UTS from 67.48 to 68.45, with improvements on five of the six tasks. This suggests that heterogeneous tasks can benefit from the joint training process which includes more diverse datasets. With TMAS, the jointly trained model further improves UTS to 69.35.

Table 2: Timestep-sampling comparison under the same joint-training protocol. Steps denotes the updates required to reach the final UTS of Fixed Beta; NR means not reached.

Sampler	Building IoU ↑	Change IoU ↑	Semantic mIoU ↑	Height δ_1 ↑	Ref. gIoU ↑	Det. mAP50 ↑	UTS ↑	Steps ↓
Uniform	75.92	74.42	50.31	71.84	73.20	61.95	67.94	NR
Fixed Beta $(2.5, 1)$	76.38	**74.75**	51.12	72.06	73.61	62.80	68.45	60k
Global adaptive	76.54	74.61	51.68	72.33	73.52	63.03	68.62	53k
Group adaptive	76.74	74.48	52.23	72.96	73.87	63.88	69.03	44k
TMAS	**76.93**	74.63	**52.57**	**73.31**	**73.93**	**64.75**	**69.35**	**37k**

Global vs. task–modality adaptive sampling. Table 2 compares several timestep-sampling strategies under the same joint-training setting. Global difficulty adaptation brings only a small gain over Fixed Beta (68.45→68.62 UTS), while using separate sampling distributions for different task–modality groups improves UTS to 69.03. Adding the dual-EMA activity signal further raises UTS to 69.35. TMAS also matches the final 60k-step performance of Fixed Beta after only 37k steps, showing faster convergence. Activity/exploration ablations are in Appendix D.2.

Figure 6 further explains the benefit of group-specific sampling. Task–modality groups show distinct relative difficulties across timesteps, with warmer colors indicating higher difficulty and cooler colors lower difficulty. The same timestep region can be difficult for one group but easy for another, which a global sampler cannot capture. In contrast, TMAS adapts training effort to each group's learning state, consistent with the larger gain over global adaptation in Table 2.

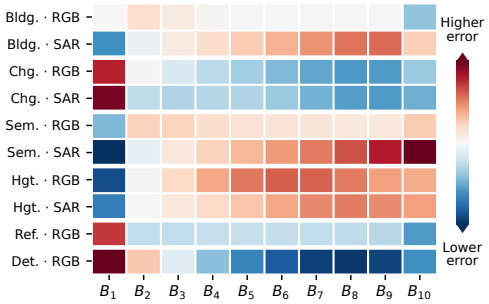


Figure 6: Task–modality timestep difficulty.

Generalization beyond the training datasets. We evaluate the same jointly trained UniFlow-RS checkpoint on two unseen referring-segmentation datasets without target-specific adaptation. RISBench mainly introduces a visual domain shift while retaining explicit referring expressions, whereas EarthReason additionally contains more implicit, reasoning-oriented language queries. As shown in Table 3, UniFlow-RS achieves 70.92/64.62 cIoU/gIoU on RISBench and 44.23/48.24 on EarthReason, outperforming the corresponding baselines by 23.94/13.00 and 8.75/11.69 points, respectively. The results show that the shared predictor generalizes to unseen visual and language distributions without additional adaptation.

Table 3: Zero-shot transfer to unseen datasets under visual and visual–language out-of-distribution (OOD) shifts. EarthReason includes both visual and language shifts, whereas RISBench mainly introduces a visual shift. All UniFlow-RS results use the same jointly trained checkpoint.

EarthReason (Visual+Language OOD)			**RISBench** (Visual OOD)		
Method	cIoU ↑	gIoU ↑	Method	cIoU ↑	gIoU ↑
FLUX.2-Klein Base	17.74	17.04	FLUX.2-Klein Base	6.38	7.74
RS2-SAM2-RRSISD (AAAI'26)	35.48	36.55	VisionReasoner (ICLR'26)	46.98	51.62
UniFlow-RS	**44.23**	**48.24**	**UniFlow-RS**	**70.92**	**64.62**

Source-state initialization. We isolate the source-state design in Eq. 3 under fixed visual conditioning. Figure 7(a) shows that z_a improves all five single-image tasks over Gaussian initialization by 2.64–3.44 points, confirming the benefit of preserving input structure. For bi-temporal change detection, the mean score increases from 70.36 with Gaussian initialization to 72.54 with z_a, 73.82 with $|z_a - z_b|$, and 74.63 with $z_a - z_b$, showing that preserving directional temporal differences is beneficial.

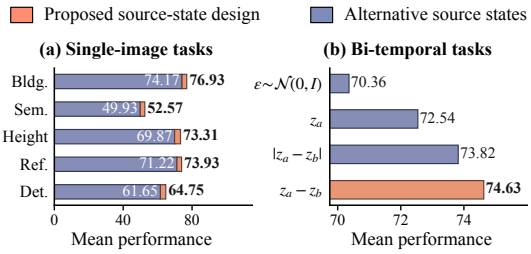


Figure 7: Source-state initialization ablation.

5 Conclusion

We presented UniFlow-RS, a unified latent-flow framework for heterogeneous remote-sensing perception. Instead of sharing only representations while retaining task-specific predictors, UniFlow-RS unifies the prediction process through a pretrained flow model with a jointly trained LoRA. Structured visual targets and task-dependent conditioning enable six tasks across thirteen datasets to share the same latent prediction space and Flow Matching objective, without learned task-specific output heads or losses. To better optimize this shared predictor, TMAS adapts timestep sampling to the online learning state of each task–modality group.

Across a diverse set of benchmarks, the resulting model remains competitive with task-specific methods, benefits from joint training, and generalizes to unseen datasets without target-specific adaptation. Overall, our results demonstrate the potential of a shared generative prediction process as a foundation for generalist remote-sensing perception, and motivate further research toward more scalable and broadly transferable unified models.

Reproducibility Statement

We provide anonymized code, configurations, and evaluation scripts with the submission. Experimental settings and evaluation protocols are documented in Sec. 4.1 and Appendices A–D, providing the details needed to reproduce the reported results.

AI Use Statement

Generative AI tools were used to assist with drafting and editing this manuscript. All AI-assisted content was reviewed by the authors, who take full responsibility for the final content, analysis, and conclusions.

References

Hanbo Bi, Yingchao Feng, Boyuan Tong, Mengyu Wang, Haichen Yu, Yongqiang Mao, Hao Chang, Wenhui Diao, Peijin Wang, Yue Yu, Hanyang Peng, Yehong Zhang, Kun Fu, and Xian Sun. RingMoE: Mixture-of-modality-experts multi-modal foundation models for universal remote sensing image interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48 (4):4388–4405, 2026.

Black Forest Labs. FLUX.2 [klein] 4B Base, 2026. URL https://huggingface.co/black-forest-labs/FLUX.2-klein-base-4B. Model card. Accessed September 22, 2026. Code: https://github.com/black-forest-labs/flux2.

Haoyang Chen, Jing Zhang, Hebaixu Wang, Shiqin Wang, Pohsun Huang, Jiayuan Li, Haonan Guo, Di Wang, Zheng Wang, and Bo Du. Any2Any: Unified arbitrary modality translation for remote sensing. *arXiv preprint arXiv:2603.04114*, 2026.

Keyan Chen, Zhengxia Zou, and Zhenwei Shi. Building extraction from remote sensing images with sparse token transformers. *Remote Sensing*, 13(21):4441, 2021.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, pp. 801–818, 2018a. URL https://www.ecva.net/papers/eccv_2018/papers_ECCV/html/Liang-Chieh_Chen_Encoder-Decoder_with_Atrous_ECCV_2018_paper.php.

Sining Chen, Yilei Shi, Zhitong Xiong, and Xiao Xiang Zhu. HTC-DC Net: Monocular height estimation from single remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–18, 2023.

Zhan Chen, Yidan Zhang, Xiyu Qi, Yongqiang Mao, Xin Zhou, Lei Wang, and Yunping Ge. HeightFormer: A multilevel interaction and image-adaptive classification–regression network for monocular height estimation with aerial images. *Remote Sensing*, 16(2):295, 2024.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 794–803, 2018b.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1290–1299, 2022a.

Gong Cheng, Jiabao Wang, Ke Li, Xingxing Xie, Chunbo Lang, Yanqing Yao, and Junwei Han. Anchor-free oriented proposal generator for object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022b.

Sijun Dong, Libo Wang, Bo Du, and Xiaoliang Meng. ChangeCLIP: Remote sensing change detection with multimodal vision-language representation learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 208:53–69, 2024a.

Zhe Dong, Yuzhe Sun, Tianzhu Liu, Wangmeng Zuo, and Yanfeng Gu. Cross-modal bidirectional interaction model for referring remote sensing image segmentation. *arXiv preprint arXiv:2410.08613*, 2024b.

Shouhang Du, Jianghe Xing, Shaoyu Wang, Xiongwu Xiao, Jun Li, and Hao Liu. LUMNet: Land use knowledge guided multiscale network for height estimation from single remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 21:1–5, 2024.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, et al. Image generators are generalist vision learners. *arXiv preprint arXiv:2604.20329*, 2026.

Yulu Gan, Sungwoo Park, Alexander Schubert, Anthony Philippakis, and Ahmed M. Alaa. InstructCV: Instruction-tuned text-to-image diffusion models as vision generalists. In *International Conference on Learning Representations*, 2024.

Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Houqiang Li, Han Hu, Dong Chen, and Baining Guo. InstructDiffusion: A generalist modeling interface for vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12709–12720, 2024.

Xin Guo, Jiangwei Lao, Bo Dang, Yingying Zhang, Lei Yu, Lixiang Ru, Liheng Zhong, Ziyuan Huang, Kang Wu, Dingxiang Hu, et al. SkySense: A multi-modal remote sensing foundation model towards universal interpretation for earth observation imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27672–27683, 2024.

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. In *International Conference on Learning Representations*, 2022.

Ziyue Huang, Yongchao Feng, Qingjie Liu, and Yunhong Wang. MutDet: Mutually optimizing pre-training for remote sensing object detection. In *Computer Vision – ECCV 2024*, pp. 1–17, 2024.

ISPRS. 2D semantic labeling contest – potsdam, n.d. URL `https://www.isprs.org/resources/datasets/benchmarks/UrbanSemLab/2d-sem-label-potsdam.aspx`. Dataset resource. Accessed September 22, 2026.

Johannes Jakubik, Felix Yang, Benedikt Blumenstiel, Erik Scheurer, Rocco Sedona, Stefano Maurogiovanni, Jente Bosmans, Nikolaos Dionelis, Valerio Marsocci, Niklas Kopp, et al. TerraMind: Large-scale generative multimodality for earth observation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7383–7394, 2025.

Donggon Jang, Yucheol Cho, Suin Lee, Taehyeon Kim, and Daeshik Kim. MMR: A large-scale benchmark dataset for multi-target and multi-granularity reasoning segmentation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Shunping Ji, Shiqing Wei, and Meng Lu. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586, 2019.

Jian Kang, Zhirui Wang, Ruoxin Zhu, Junshi Xia, Xian Sun, Ruben Fernandez-Beltran, and Antonio Plaza. Disoptnet: Distilling semantic knowledge from optical images for weather-independent building segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.

Myunsoo Kim, Donghyeon Ki, Seong-Woong Shim, and Byung-Jun Lee. Adaptive non-uniform timestep sampling for accelerating diffusion model training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2513–2522, 2025.

Hakjin Lee, Minki Song, Jamyoung Koo, and Junghoon Seo. Hausdorff distance matching with adaptive query denoising for rotated detection transformer. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1872–1882, 2025.

Yang Lei and Wanshou Jiang. A relation-aware and edge-preserving height refinement network for single-view height estimation from remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 18:12988–13002, 2025.

Jiepan Li, Wei He, Weinan Cao, Liangpei Zhang, and Hongyan Zhang. UANet: An uncertainty-aware network for building extraction from remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–13, 2024.

Jiepan Li, Wei He, Zhuohong Li, Yujun Guo, and Hongyan Zhang. Overcoming the uncertainty challenges in detecting building changes from remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 220:1–17, 2025a.

Kaiyu Li, Ruixun Liu, Xiangyong Cao, Xueru Bai, Feng Zhou, Deyu Meng, and Zhi Wang. SegEarth-OV: Towards training-free open-vocabulary segmentation for remote sensing images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10545–10556, 2025b.

Kaiyu Li, Zepeng Xin, Li Pang, Chao Pang, Yupeng Deng, Jing Yao, Guisong Xia, Deyu Meng, Zhi Wang, and Xiangyong Cao. Segearth-r1: Geospatial pixel reasoning via large language model. *arXiv preprint arXiv:2504.09644*, 2025c.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Guozhang Liu, Baochai Peng, Ting Liu, Pan Zhang, Mengke Yuan, Chaoran Lu, Ningning Cao, Sen Zhang, Simin Huang, Tao Wang, et al. Large-scale fine-grained building classification and height estimation for semantic urban reconstruction: Outcome of the 2023 IEEE GRSS data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17: 11194–11207, 2024a.

Sihan Liu, Yiwei Ma, Xiaoqing Zhang, Haowei Wang, Jiayi Ji, Xiaoshuai Sun, and Rongrong Ji. Rotated multi-scale interaction network for referring remote sensing image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26658–26668, 2024b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Yuqi Liu, Tianyuan Ou, Zhisheng Zhong, Bohao Peng, Shu Liu, Bei Yu, and Jiaya Jia. VisionReasoner: Unified reasoning-integrated visual perception via reinforcement learning. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=QoDOwjsbAq.

Liang Lv, Di Wang, Jing Zhang, and Lefei Zhang. S5: Scalable semi-supervised semantic segmentation in remote sensing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(10): 7726–7734, 2026.

Yongqiang Mao, Xian Sun, Xingliang Huang, and Kaiqiang Chen. LIGHT: Joint individual building extraction and height estimation from satellite images through a unified multitask learning network. In *IEEE International Geoscience and Remote Sensing Symposium*, 2023. URL https://arxiv.org/abs/2304.01090.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8162–8171, 2021.

Hao Pei, Peishuang Ni, Yiguo Qiao, Jianlai Chen, Wenkang Liu, Hanwen Yu, Fan Wu, Gang Xu, and Mengdao Xing. AEM-KD: Adversarial entropy minimization for cross-modality knowledge distillation in SAR building segmentation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 19:16201–16219, 2026.

Xiaohan Qin, Xiaoxing Wang, and Junchi Yan. Revisiting fairness in multitask learning: A performance-driven approach for variance reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20492–20501, 2025.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Fu Rong, Meng Lan, Qian Zhang, and Lefei Zhang. RS2-SAM2: Customized SAM2 for referring remote sensing image segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(11):8751–8759, 2026.

Andranik Sargsyan and Shant Navasardyan. FlowDIS: Language-guided dichotomous image segmentation with flow matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42039–42048, 2026.

Akashah Shabbir, Mohammed Zumri, Mohammed Bennamoun, Fahad Shahbaz Khan, and Salman Khan. GeoPixel: Pixel grounding large multimodal model in remote sensing. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 54095–54111, 2025.

Li Shen, Yao Lu, Hao Chen, Hao Wei, Donghai Xie, Jiabao Yue, Rui Chen, Shouye Lv, and Bitao Jiang. S2Looking: A satellite side-looking dataset for building change detection. *Remote Sensing*, 13(24):5094, 2021.

Jacob Shermeyer, Daniel Hogan, Jason Brown, Adam Van Etten, Nicholas Weir, Fabio Pacifici, Ronny Hänsch, Alexei Bastidas, Scott Soenen, Todd Bacastow, and Ryan Lewis. SpaceNet 6: Multi-sensor all weather mapping dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 768–777, 2020.

Furong Shi and Tong Zhang. A multi-task network with distance–mask–boundary consistency constraints for building extraction from aerial images. *Remote Sensing*, 13(14):2656, 2021.

Jian Song, Hongruixuan Chen, Weihao Xuan, Junshi Xia, and Naoto Yokoya. SynRS3D: A synthetic dataset for global 3D semantic understanding from monocular remote sensing imagery. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d4f3da38b491b44d40a0d5a5b37134ba-Abstract-Datasets_and_Benchmarks_Track.html.

Jieyi Tan, Chengwei Zhang, Bo Dang, and Yansheng Li. Towards privacy-preserved pre-training of remote sensing foundation models with federated mutual-guidance learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1804–1814, 2025.

Di Wang, Jing Zhang, Minqiang Xu, Lin Liu, Dongsheng Wang, Erzhong Gao, Chengxi Han, Haonan Guo, Bo Du, Dacheng Tao, and Liangpei Zhang. MTP: Advancing remote sensing foundation model via multi-task pretraining. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:11632–11654, 2024a.

Fengxiang Wang, Hongzhen Wang, Di Wang, Zonghao Guo, Zhenyu Zhong, Long Lan, Wenjing Yang, and Jing Zhang. Harnessing massive satellite imagery with efficient masked image modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6935–6947, 2025.

Junjue Wang, Zhuo Zheng, Ailong Ma, Xiaoyan Lu, and Yanfei Zhong. LoveDA: A remote sensing land-cover dataset for domain adaptive semantic segmentation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

Libo Wang, Sijun Dong, Ying Chen, Xiaoliang Meng, Shenghui Fang, and Songlin Fei. MetaSegNet: Metadata-collaborative vision-language representation learning for semantic segmentation of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–11, 2024b.

Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6830–6839, 2023.

Yuxuan Wang, Lanxin Zeng, Wen Yang, Jian Kang, Huai Yu, Mihai Datcu, and Gui-Song Xia. Extracting building footprints in SAR images via distilling boundary information from optical images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024c.

Tianyu Wei, He Chen, Wenchao Liu, Liang Chen, Panzhe Gu, and Jue Wang. Optical and SAR cross-modal hallucination collaborative learning for remote sensing missing-modality building footprint extraction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 19:1183–1196, 2026.

Junshi Xia, Naoto Yokoya, Bruno Adriano, and Clifford Broni-Bediako. OpenEarthMap: A benchmark dataset for global high-resolution land cover mapping. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6254–6264, 2023.

Junshi Xia, Hongruixuan Chen, Clifford Broni-Bediako, Yimin Wei, Jian Song, and Naoto Yokoya. OpenEarthMap-SAR: A benchmark synthetic aperture radar dataset for global high-resolution land cover mapping [software and data sets]. *IEEE Geoscience and Remote Sensing Magazine*, 13(4): 476–487, 2025.

Zepeng Xin, Kaiyu Li, Luodi Chen, Wanchen Li, Yuchen Xiao, Hui Qiao, Weizhan Zhang, Deyu Meng, and Xiangyong Cao. SegEarth-R2: Towards comprehensive language-guided segmentation for remote sensing images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13199–13210, 2026.

Zhitong Xiong, Yi Wang, Fahong Zhang, Adam J. Stewart, Joëlle Hanna, Damian Borth, Ioannis Papoutsis, Bertrand Le Saux, Gustau Camps-Valls, and Xiao Xiang Zhu. Neural plasticity-inspired multimodal foundation model for earth observation. *arXiv preprint arXiv:2403.15356*, 2024.

Ruoyu Yang, Yinhe Liu, Heng Yan, Yiheng Zhou, Yihan Fu, Han Luo, and Yanfei Zhong. MaRS: A multi-modality very-high-resolution remote sensing foundation model with cross-granularity meta-modality learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14): 11685–11693, 2026a.

Zhenyu Yang, Gensheng Pei, Tao Chen, Xia Yuan, Haofeng Zhang, Xiangbo Shu, and Yazhou Yao. Beyond quadratic: Linear-time change detection with RWKV. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14):11811–11819, 2026b.

Siyuan Yao, Dongxiu Liu, Taotao Li, Shengjie Li, Wenqi Ren, and Xiaochun Cao. UAGLNet: Uncertainty-aggregated global-local fusion network with cooperative cnn-transformer for building extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 64:1–14, 2026.

Xu Zhang, Danyang Li, Xiaohang Dong, Tianhao Wu, Hualong Yu, Jianye Wang, Qicheng Li, and Xiang Li. UniChange: Unifying change detection with multimodal large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42169–42179, 2026.

Yingying Zhang, Lixiang Ru, Kang Wu, Lei Yu, Lei Liang, Yansheng Li, and Jingdong Chen. SkySense V2: A unified foundation model for multi-modal remote sensing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9136–9146, 2025.

Jiaqi Zhao, Zeyu Ding, Yong Zhou, Hancheng Zhu, Wen-Liang Du, Rui Yao, and Abdulmotaleb El Saddik. OrientedFormer: An end-to-end transformer-based oriented object detector in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16, 2024a.

Jie Zhao, Zhitong Xiong, and Xiao Xiang Zhu. UrbanSARFloods: Sentinel-1 SLC-based benchmark dataset for urban and open-area flood mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 419–429, 2024b.

Xinyu Zhao, Jun Xie, Shengzhe Chen, and Jun Liu. Convex combination star shape prior for data-driven image semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14068–14077, 2025.

Renjie Zhou. TriCDNet: A multi-scale tri-stream interaction network for building change detection. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XLVIII-3/W4-2025, pp. 63–69, 2026.

Xuechao Zou, Yue Li, Shun Zhang, Kai Li, Shiying Wang, Pin Tao, Junliang Xing, and Congyan Lang. Dynamic dictionary learning for remote sensing image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22457–22466, 2025.

APPENDIX

A DATASET DETAILS AND EVALUATION PROTOCOLS

We evaluate six tasks across thirteen public remote-sensing datasets, spanning optical and SAR imagery, single- and bi-temporal inputs, continuous prediction, language-conditioned segmentation, and instance-level detection. We follow the official or released dataset splits and evaluation protocols whenever available. Target-codec choices are determined using the validation split, or the training split when no validation split is provided. Dataset-specific preprocessing and evaluation details are described below.

Dataset-specific protocols. For building extraction, WHU Building uses RGB aerial imagery and follows the provider-defined folders (Ji et al., 2019). For SpaceNet 6, we use SAR imagery and follow the data partitioning strategy commonly adopted by recent baselines. The released building polygons are rasterized onto the corresponding image grids and converted into binary targets using the target codec (Shermeyer et al., 2020).

For change detection, WHU-CD and LEVIR-CD+ use bi-temporal RGB imagery. WHU-CD follows the released 512×512 training and test partitions, containing 1,260 and 690 bi-temporal image pairs, respectively, while LEVIR-CD+ follows its released train/test partition (Shen et al., 2021). UrbanSARFloods uses bi-temporal SAR observations and follows the released training/validation lists and held-out event test cases (Zhao et al., 2024b). Geometric transforms are synchronized across both temporal observations and the target.

For semantic segmentation, LoveDA and OpenEarthMap use RGB imagery, whereas OpenEarthMap-SAR uses SAR imagery; all three datasets follow their released partitions (Wang et al., 2021; Xia et al., 2023; 2025). Dataset-native labels are mapped to the canonical visual codebook using deterministic lookup tables, and ignored or unlabeled pixels are excluded from both training loss and evaluation.

For height estimation, Potsdam uses RGB imagery with the 24/14 labeled-train/held-out-test split (ISPRS, n.d.). DFC23 Track 2 provides both RGB and SAR observations and follows the commonly used data split adopted by prior baseline methods (Liu et al., 2024a). Both modalities are used as height-estimation samples during joint training, while the reported benchmark result uses RGB imagery at inference. Invalid height pixels are masked before the log-compressed target encoding.

For referring segmentation, RRSIS-D and LaSeRS use RGB imagery paired with text expressions. RRSIS-D uses its official split (Liu et al., 2024b), whereas LaSeRS uses the released train/test annotations without repartitioning (Xin et al., 2026). For rotated detection, DIOR-R uses RGB imagery and retains its official train/validation/test split (Cheng et al., 2022b).

Common preprocessing. Intensity-normalization statistics are estimated from the training partition. Geometric augmentations are synchronized across inputs and targets, including both temporal observations for change detection. Evaluation predictions are restored to the native image grid before applying the deterministic recovery operator R_d.

A.1 TEXT PROMPTS AND LANGUAGE CONDITIONING

The language condition is constructed as $c_T = T(q)$. For building extraction, change detection, semantic segmentation, height estimation, and rotated detection, q is a fixed instruction determined by the dataset and input modality. Referring segmentation instead uses the sample-specific expression provided by the dataset as q. Tables 4 and 5 list the configured fixed instructions. The same text-conditioning protocol is used for training and evaluation.

Prompt assignment and referring expressions. WHU Building uses the optical building prompt, and SpaceNet 6 SAR uses the SAR building prompt. WHU-CD and LEVIR-CD+ share the optical change prompt, whereas UrbanSARFloods uses the SAR flood-change prompt. Potsdam and the optical branch of DFC23 Track 2 share the optical height prompt, while the SAR branch uses its SAR counterpart. The semantic prompts distinguish LoveDA, OpenEarthMap optical, and OpenEarthMap-SAR.

Table 4: Fixed text prompts for building extraction, change detection, height estimation, and rotated detection. Prompts are shown verbatim.

Task / input	Text prompt
Building / optical	`Generate a binary building-footprint map for the optical aerial image. Buildings are white; background is black.`
Building / SAR	`Generate a binary building-footprint map for the synthetic aperture radar image. Buildings are white; background is black.`
Change / optical pair	`Generate a binary building-change map for the before and after optical aerial images. Changed buildings are white; unchanged areas are black.`
Change / SAR pair	`Generate a per-pixel flood-change map for the pre-event and co-event synthetic aperture radar images, using this RGB color mapping: non-flood <0,0,0>; open-area flood <0,0,255>; urban flood <255,0,0>.`
Height / optical	`Generate a dense height map for the optical aerial image. Encode log1p height from 0 to 200 meters as grayscale, from black to white.`
Height / SAR	`Generate a dense height map for the synthetic aperture radar image. Encode log1p height from 0 to 200 meters as grayscale, from black to white.`
Detection / DIOR-R	`Generate a class-coded mask of all DIOR-R object classes.`

Table 5: Dataset- and modality-specific semantic-segmentation prompts. RGB triplets and hexadecimal colors are explicitly included in the prompts.

Dataset / input	Text prompt
LoveDA / optical	`Generate a per-pixel LoveDA semantic map for the optical aerial image, using the VAE-aware canonical palette: background <0,0,0>; building <222,31,7>; road <255,255,255>; water <0,69,255>; barren <255,0,255>; forest <34,97,38>; agriculture <0,255,255>.`
OpenEarthMap / optical	`Generate an OpenEarthMap optical semantic map using the VAE-aware canonical palette. RGB colors: background black #000000; bareland magenta #FF00FF; rangeland bright green #00FF24; developed space gray #949494; road white #FFFFFF; tree dark green #226126; water blue #0045FF; agriculture land cyan #00FFFF; building red #DE1F07.`
OpenEarthMap-SAR / SAR	`Generate an OpenEarthMap SAR semantic map using the VAE-aware canonical palette. RGB colors: background black #000000; bareland magenta #FF00FF; rangeland bright green #00FF24; developed space gray #949494; road white #FFFFFF; tree dark green #226126; water blue #0045FF; agriculture land cyan #00FFFF; building red #DE1F07.`

For RRSIS-D, we use the annotation's `raw` field, falling back to `sent` when unavailable; for LaSeRS, we use the `description` field. The dataset expressions are otherwise used unchanged except for whitespace trimming.

Text encoding. We use the frozen Qwen3 text conditioner associated with FLUX.2-Klein Base. Each prompt is passed as the content of a single user message through the tokenizer's chat template, with `add_generation_prompt=True` and `enable_thinking=False`. The resulting sequence is padded or truncated to 128 tokens, including the chat-template tokens. Hidden states at indices 9, 18, and 27 are concatenated along the feature dimension to obtain a 128×7680 condition. Fixed instructions are encoded once and cached, while referring conditions are cached per expression.

A.2 Evaluation Metrics

Table 6 lists the evaluation measure used for every dataset. Unless stated otherwise, overlap and accuracy scores are reported as percentages, and ignored or invalid pixels are excluded according to the official evaluation protocol.

Table 6: Evaluation metrics for the thirteen datasets. Arrows indicate the preferred direction.

Task	Dataset	Reported metric(s)
Building extraction	WHU Building	IoU ↑
	SpaceNet 6 (SAR)	IoU ↑
Change detection	WHU-CD	IoU ↑
	LEVIR-CD+	IoU ↑
	UrbanSARFloods (SAR)	mIoU ↑
Semantic segmentation	LoveDA	mIoU ↑
	OpenEarthMap-RGB	mIoU ↑
	OpenEarthMap-SAR	mIoU ↑
Height estimation	Potsdam	RMSE (m) ↓
	DFC23 Track 2	δ_1 ↑
Referring segmentation	RRSIS-D	cIoU / gIoU ↑
	LaSeRS	cIoU / gIoU ↑
Rotated detection	DIOR-R	mAP50 ↑

Unified Task Score (UTS). For controlled comparisons among unified variants, let $M_d^{(a)} \in [0, 1]$ be the selected higher-is-better score for method a on dataset d. We first average the UTS-eligible datasets within each task and then assign equal weight to all six tasks:

$$S_\tau^{(a)} = \frac{100}{|\mathcal{D}_\tau^{\mathrm{UTS}}|} \sum_{d \in \mathcal{D}_\tau^{\mathrm{UTS}}} M_d^{(a)}, \qquad \mathrm{UTS}(a) = \frac{1}{6} \sum_{\tau \in \mathcal{T}} S_\tau^{(a)}, \quad |\mathcal{T}| = 6. \tag{8}$$

We use IoU for building extraction, IoU or mIoU for change detection, mIoU for semantic segmentation, δ_1 for height estimation, gIoU for referring segmentation, and mAP50 for rotated detection. Potsdam RMSE and the complementary cIoU results are reported separately and excluded from UTS. UTS is reported for controlled comparisons among unified-model variants, alongside the native benchmark metrics.

IoU and mIoU. For class c, intersection over union is

$$\mathrm{IoU}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c}. \tag{9}$$

For binary building extraction and optical change detection, we report the foreground-class IoU. For multi-class flood and land-cover segmentation, mean IoU averages class-wise IoU over the C evaluated classes,

$$\mathrm{mIoU} = \frac{1}{C} \sum_{c=1}^{C} \mathrm{IoU}_c \,.$$

Height metrics. Let Ω be the set of valid evaluation pixels, and let h_p and $\hat{h}_p$ denote the reference and predicted height in meters. Potsdam is evaluated with root-mean-square error,

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\Omega|} \sum_{p \in \Omega} (\hat{h}_p - h_p)^2}, \tag{10}$$

where lower is better. DFC23 Track 2 uses the threshold accuracy δ_1, i.e., the percentage of valid pixels whose multiplicative error is below 1.25:

$$\delta_1 = \frac{100}{|\Omega|} \sum_{p \in \Omega} \mathbf{1}\left[\max\left(\frac{\max(\hat{h}_p, \epsilon)}{\max(h_p, \epsilon)}, \frac{\max(h_p, \epsilon)}{\max(\hat{h}_p, \epsilon)} \right) < 1.25 \right]. \tag{11}$$

We use the official validity mask and set ϵ to avoid division by zero.

Referring-segmentation metrics. For query i, let I_i and U_i be the pixel intersection and union between its predicted and reference masks. gIoU averages the per-query IoUs, whereas cIoU pools pixels across the complete evaluation set:

$$\mathrm{gIoU} = \frac{100}{N} \sum_{i=1}^{N} \frac{I_i}{U_i}, \qquad \mathrm{cIoU} = 100 \frac{\sum_{i=1}^{N} I_i}{\sum_{i=1}^{N} U_i}. \tag{12}$$

gIoU weights every query equally, while cIoU gives greater influence to masks with larger union areas. We use the same cIoU/gIoU names for both referring benchmarks; these correspond to the oIoU/mIoU terminology, respectively, used in the original RRSIS-D evaluation. For zero-shot transfer analyses, we additionally report $\mathrm{Pr@}\alpha$, the percentage of queries whose individual IoU is at least threshold α.

Rotated-detection metric. For each DIOR-R class, AP50 is the area under the precision–recall curve when a prediction is counted as a true positive at a rotated-box IoU threshold of 0.5. mAP50 is the unweighted mean of AP50 over the 20 object classes; duplicate detections are treated as false positives by the standard one-to-one matching procedure.

B Structured-Target Codecs and Deterministic Recovery

This section specifies the deterministic visual carriers used for each task and evaluates how faithfully they survive the frozen VAE interface before any flow-model learning is involved.

Binary targets. Building extraction, binary change detection, and referring segmentation use a black background and white foreground. This maximizes separation in the visual carrier while keeping recovery a deterministic thresholding operation.

Semantic targets. LoveDA and OpenEarthMap use a shared canonical semantic codebook. Dataset-native class IDs are deterministically mapped into canonical semantics before rendering, and evaluation maps the recovered predictions back to each dataset's native class IDs. To determine the visual carrier, we compare three palette choices: *dataset official palettes*, which preserve each dataset's released visualization colors; a *shared canonical palette*, which uses one common palette after ontology alignment; and our *VAE-aware canonical palette*, which keeps the same canonical semantic mapping but selects the visual color codes according to their frozen-VAE round-trip stability. UrbanSARFloods uses a separate three-class flood encoding because its ontology is not a land-cover codebook.

Metric height targets. Potsdam and DFC23 share a log-compressed height representation over $[0, 200]$ meters:

$$u = \frac{\log(1 + \mathrm{clip}(h, 0, 200))}{\log(201)}, \qquad C_h(h) = (2u - 1, 2u - 1, 2u - 1), \tag{13}$$

with inverse mapping

$$\hat{h} = \exp(\hat{u} \log(201)) - 1, \tag{14}$$

where $\hat{u} = \mathrm{clip}((\hat{c} + 1)/2, 0, 1)$ for decoded grayscale $\hat{c} \in [-1, 1]$. The three-channel image is only a VAE carrier; evaluation is performed on a continuous single-channel height map in meters.

Multi-class rotated detection targets. DIOR-R is rendered as a 20-class palette canvas. After VAE decoding, each pixel is assigned to its nearest palette code. We then separate connected components by class, extract minimum-area rotated rectangles, and apply deterministic rotated-box post-processing. This recovery path contains no learned parameters and makes representation error measurable at both raster and instance levels.

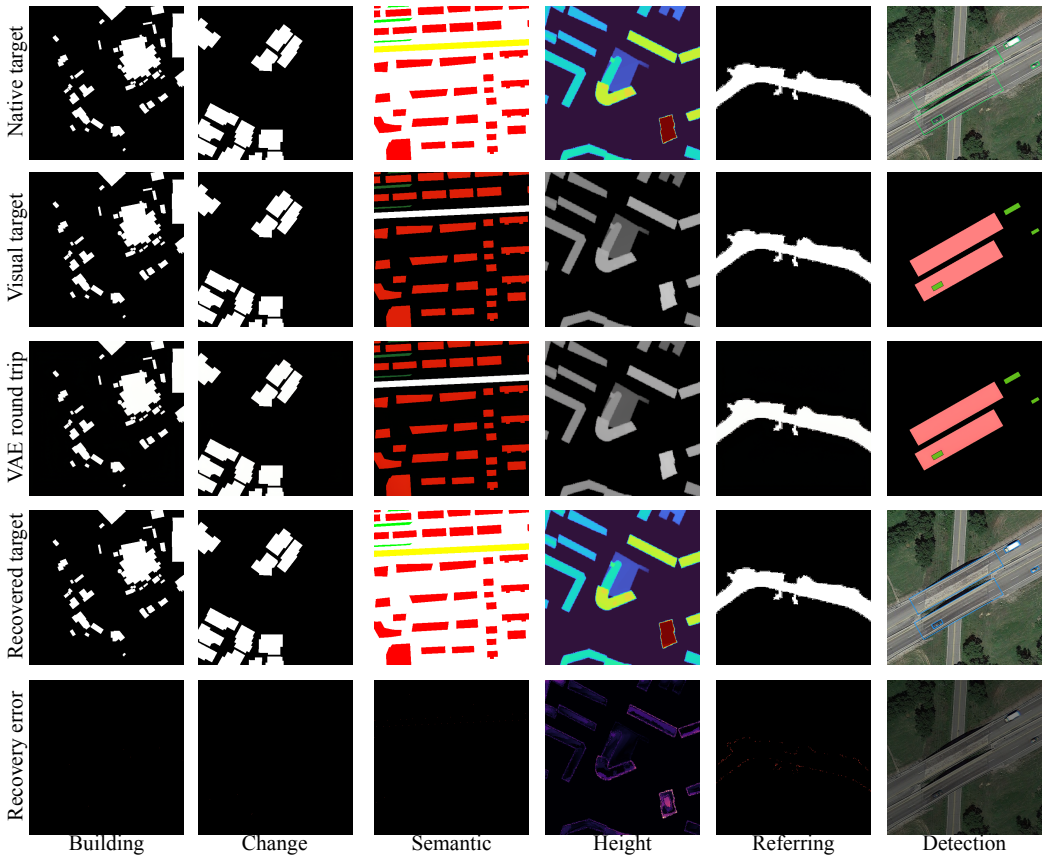


Figure 8: Frozen-VAE round-trip analysis of the structured visual target space. Columns show building, change, semantic, height, referring, and rotated detection targets; rows show the native target, visual target, VAE round trip, recovered target, and recovery error. The experiment measures whether task semantics are preserved by the frozen VAE independently of flow-model learning.

Direct codec sanity check. Before involving the frozen VAE, we verify the direct deterministic path

$$y_d^{\text{codec}} = R_d(C_{\tau(d),d}(y_d)), \qquad B_d^{\text{codec}} = M_d(y_d^{\text{codec}}, y_d). \tag{15}$$

For binary masks, discrete semantic palettes, and valid-range height encodings, this check verifies that the evaluation implementation itself is consistent with the encoding definition. For DIOR-R, we report raster mIoU, mAP50, and split rate after connected components, minimum-area rectangle fitting, and rotated-box post-processing. These numbers quantify the geometry loss introduced by deterministic instance recovery independently of both the VAE and the flow model.

Frozen-VAE round-trip protocol. To isolate representation loss from flow-model prediction error, we evaluate

$$\tilde{y}_d = R_d\big(D\big(E\big(C_{\tau(d),d}(y_d)\big)\big)\big), \qquad B_d^{\text{repr}} = M_d(\tilde{y}_d, y_d). \tag{16}$$

All codec comparisons follow the data-split protocol described in Appendix A. The input resolution, VAE precision, and recovery operator are fixed, and only the visual encoding is varied. No LoRA checkpoint, MMDiT forward pass, text condition, or visual condition is involved.

Round-trip results. Figure 8 visualizes the frozen-VAE round trip for all six structured output families.

Binary masks remain visually stable through the round trip, while the error maps make the more subtle palette and geometric deviations in semantic, height, and rotated-instance targets explicit. We next quantify the semantic, height, and rotated-instance visual carriers under the same frozen-VAE round-trip protocol.

Table 7: Semantic-palette ablation under frozen-VAE round-trip evaluation. Results are reported as native mIoU.

Semantic target palette	LoveDA ↑	OEM-RGB ↑	OEM-SAR ↑	Mean ↑
Dataset official palettes	99.41	96.57	90.79	95.59
Shared canonical palette	**99.73**	93.28	87.91	93.64
VAE-aware canonical (ours)	99.72	**98.83**	**99.11**	**99.22**

Semantic-palette ablation. Table 7 shows that simply sharing a canonical palette does not guarantee good compatibility with the frozen VAE: its mean round-trip mIoU is 93.64. Selecting the canonical visual codes according to frozen-VAE round-trip stability raises the mean to 99.22, while retaining a shared semantic codebook across LoveDA and OpenEarthMap. This supports the use of the VAE-aware canonical palette as the semantic visual carrier in the main model.

Table 8: Height-encoding ablation under frozen-VAE round-trip evaluation. Both variants use the same frozen VAE and fixed 0.2-m ground-recovery threshold.

Height encoding	Potsdam RMSE (m) ↓	DFC23 δ_1 (%) ↑
Linear grayscale	0.378	72.22
Log1p grayscale (ours)	**0.176**	**97.42**

Table 8 shows that log-compressed height encoding substantially improves frozen-VAE round-trip fidelity, reducing Potsdam RMSE from 0.378 to 0.176 m and increasing DFC23 δ_1 from 72.22% to 97.42%.

Table 9: DIOR-R target-palette round-trip and deterministic instance recovery.

Palette	Raster mIoU ↑	mAP50 ↑	Split ↓
Initial palette	97.90	89.44	0.50
VAE-aware hybrid palette (ours)	**98.43**	**90.10**	**0.44**

Table 9 shows that the VAE-aware hybrid palette improves both pixel-level fidelity and deterministic instance recovery, increasing raster mIoU from 97.90 to 98.43 and mAP50 from 89.44 to 90.10, while reducing the split rate from 0.50 to 0.44.

C Additional Analyses

This section complements the main-text analysis with detailed source-state results, transport-state visualizations, cross-task interaction, and zero-shot transfer studies.

C.1 Source-State Initialization Details

Figure 7 summarizes the source-state ablation in the main text. Table 10 reports the corresponding numerical results. Visual conditioning is fixed within each panel so that only the source-state initialization changes.

Table 10: Full numerical results for the source-state initialization ablation. Visual conditioning is fixed within each panel.

(a) Single-image tasks

Source z_s	Bldg. IoU ↑	Sem. mIoU ↑	Height δ_1 ↑	Ref. gIoU ↑	Det. mAP50 ↑
$\epsilon \sim \mathcal{N}(0, I)$	74.17	49.93	69.87	71.22	61.65
Current image latent z_a	**76.93**	**52.57**	**73.31**	**73.93**	**64.75**

(b) Bi-temporal change detection

Source z_s	WHU-CD IoU ↑	LEVIR-CD+ IoU ↑	UrbanSAR mIoU ↑	Mean ↑
$\epsilon \sim \mathcal{N}(0, I)$	86.49	72.26	52.32	70.36
z_a	88.91	74.49	54.23	72.54
$\lvert z_a - z_b \rvert$	90.14	75.70	55.61	73.82
$z_a - z_b$	**90.88**	**76.44**	**56.57**	**74.63**

The numerical results confirm the trends summarized in Fig. 7. For single-image tasks, z_a improves over Gaussian initialization on all five task metrics. For change detection, the mean score increases from 70.36 with Gaussian initialization to 72.54 with z_a, 73.82 with $|z_a - z_b|$, and 74.63 with the signed residual $z_a - z_b$.

C.2 Transport-State Trajectories

Figure 9 visualizes how the shared predictor transports task-dependent source states toward structured visual outputs. We use the same trained UniFlow-RS checkpoint for all six tasks and run ten-step Euler inference to expose intermediate states along the trajectory. The main quantitative evaluation continues to use the default two-step inference setting. For each example, we show solver states at $t = 1, 0.815, 0.618, 0.525, 0.398$, and 0, with times rounded for presentation.

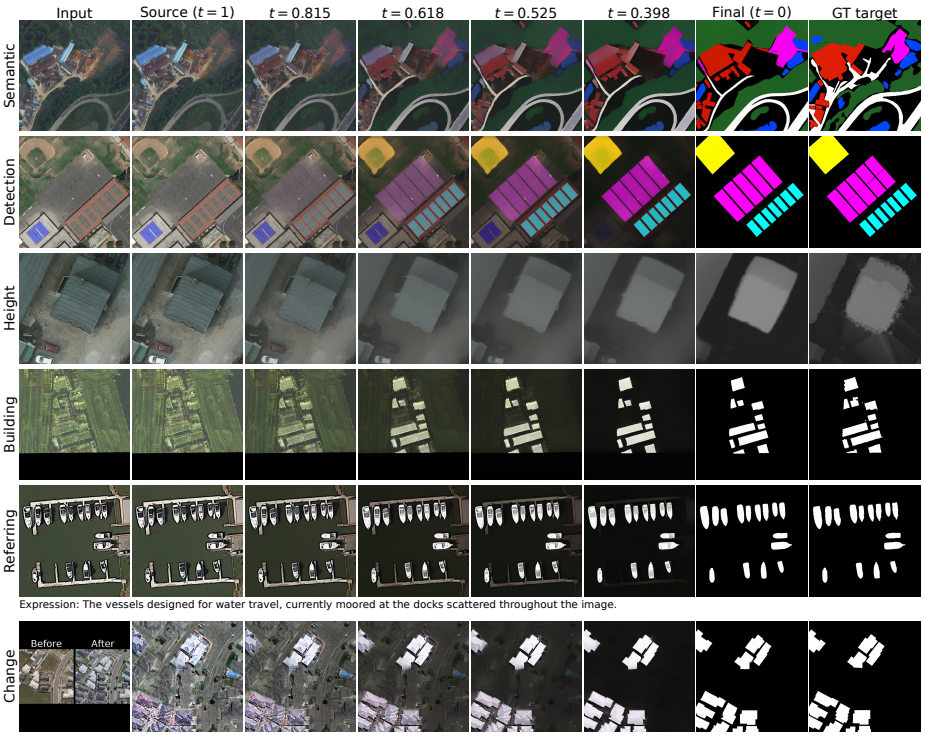


Figure 9: **Transport-state trajectories across six tasks.** All rows use the same shared UniFlow-RS checkpoint. From top to bottom: semantic segmentation (LoveDA), rotated detection (DIOR-R), height estimation (Potsdam), building extraction (SpaceNet 6 SAR), referring segmentation (LaSeRS), and change detection (WHU-CD). Columns show the input, decoded source state, selected intermediate states $D(z_t)$, final decoded state, and ground-truth structured visual target.

Decoded transport states. Each intermediate panel directly decodes the current solver state as $D(z_t)$ using the frozen VAE decoder. No thresholding, palette snapping, or instance recovery is applied to these intermediate outputs, and the ground-truth structured target is shown only in the rightmost column for reference. Accordingly, height estimation is visualized in its log-compressed grayscale representation, while rotated detection is shown as the generated class-coded canvas before geometric instance recovery. For change detection, the source state is the signed latent difference $z_a - z_b$ (after minus before), so its initial decoding differs from the appearance of either input observation. Referring segmentation uses the sample's original expression, shown below the corresponding row.

Across the single-image tasks, the decoded states progressively evolve from image-like source representations toward binary, categorical, or continuous target structures. Change detection follows a different trajectory from its latent-difference source. Together, these examples visualize how the same latent flow process produces structurally different outputs across tasks.

C.3 Cross-Task Interaction

To examine how different tasks interact during joint optimization, we analyze their gradients on the shared LoRA at the same initialization checkpoint. For each task τ, we compute the gradient of its loss with respect to the shared LoRA parameters ϕ. For every pair (τ, j), their interaction is measured by cosine similarity:

$$g_\tau = \nabla_\phi \mathcal{L}_\tau, \qquad \rho_{\tau,j} = \frac{g_\tau^\top g_j}{\|g_\tau\|_2 \, \|g_j\|_2}. \tag{17}$$

Positive values indicate locally aligned updates, whereas negative values indicate conflicting gradient directions.

We compare UniFlow-RS with a task-specific-head variant that uses the same frozen FLUX.2/MMDiT backbone, shared LoRA parameterization, conditioning inputs, initialization checkpoint, and TMAS setting, but replaces the unified generative prediction interface with independent task-specific prediction heads. This comparison keeps the optimization setting fixed while varying the prediction formulation.

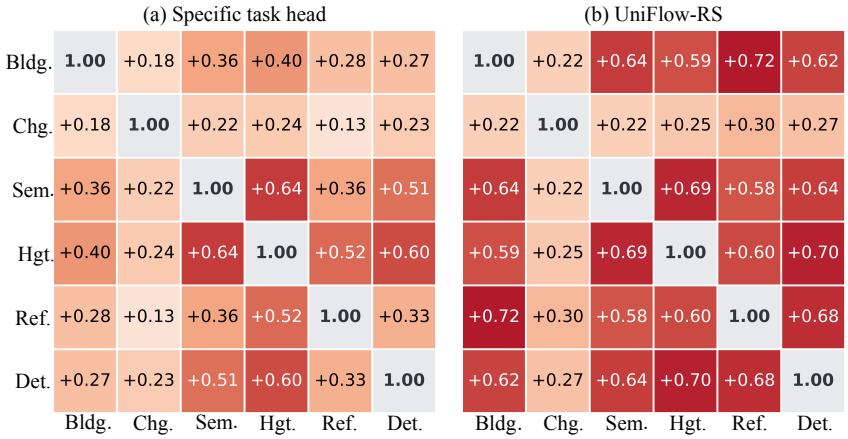


Figure 10: Pairwise gradient cosine similarities between tasks on the shared LoRA at the same initialization checkpoint: (a) task-specific heads and (b) UniFlow-RS. Both configurations use the same frozen FLUX.2/MMDiT backbone, LoRA parameterization, and TMAS setting. The diagonal is 1 by construction.

As shown in Figure 10, all fifteen task pairs have positive cosine similarity at the measured checkpoint. The mean off-diagonal similarity is 0.51 for UniFlow-RS, compared with 0.35 for the task-specific-head variant. Building–referring (0.72) and height–detection (0.70) show the strongest alignment, while change detection has lower similarities with the other tasks (0.22–0.30). These results indicate more aligned local updates under the unified prediction formulation, consistent with the joint-training gains in Table 1.

C.4 Zero-Shot Transfer Details

We extend the main-text zero-shot evaluation with thresholded overlap precision, which measures how often individual queries remain accurate under progressively stricter IoU criteria.

Table 11: Detailed zero-shot transfer results on two unseen datasets. Pr@α denotes the percentage of queries with IoU $\geq \alpha$, and "–" indicates that the corresponding result is not reported.

Target	Method	cIoU	gIoU	Pr@0.5	Pr@0.6	Pr@0.7	Pr@0.8	Pr@0.9
EarthReason	FLUX.2-Klein Base	17.74	17.04	8.92	4.15	1.97	0.78	0.05
	RS2-SAM2-RRSISD	35.48	36.55	37.14	32.47	28.01	22.51	14.63
	UniFlow-RS	**44.23**	**48.24**	**50.88**	**45.64**	**41.65**	**35.79**	**24.74**
RISBench	FLUX.2-Klein Base	6.38	7.74	2.78	1.44	0.75	0.29	0.07
	VisionReasoner	46.98	51.62	–	–	–	–	–
	UniFlow-RS	**70.92**	**64.62**	**72.07**	**67.31**	**60.96**	**51.03**	**32.47**

Table 11 reports thresholded overlap precision in addition to cIoU and gIoU. On EarthReason, where both imagery and text queries are out of distribution with respect to joint training, the advantage of UniFlow-RS persists as the overlap criterion becomes stricter: Pr@0.9 increases from 14.63 for RS2-SAM2-RRSISD to 24.74. RISBench, which primarily introduces a visual distribution shift, similarly retains 32.47 Pr@0.9, complementing the unseen-dataset results reported in the main text.

D TMAS Implementation and Component Ablations

D.1 Estimator and Adaptive Allocation

The main text gives the conceptual TMAS formulation; this subsection specifies the estimator state, distributed aggregation, unseen-bin handling, numerical safeguards, and exact sampling procedure used in the implementation.

Group keys and per-sample error. TMAS maintains one estimator profile for each task–modality pair, $g = (\tau, m)$, where τ is one of the six tasks and m is the physical modality of the current training sample. Dataset identity is not included in the key. Consequently, datasets sharing the same task and modality jointly update the same timestep profile, whereas RGB and SAR samples remain separated by modality. For multimodal sources, the group is determined by the modality of the actual sampled view.

Under the current training mixture, the active group set is

$$\mathcal{G} = \{\texttt{building/rgb}, \texttt{building/sar}, \texttt{change/rgb}, \texttt{change/sar}, \texttt{semantic/rgb}, \texttt{semantic/sar}, \texttt{height/rgb}, \texttt{height/sar}, \texttt{referring/rgb}, \texttt{detection/rgb}\}.$$

For example, WHU-CD and LEVIR-CD+ jointly update the `change/rgb` profile, while RGB and SAR semantic samples update `semantic/rgb` and `semantic/sar`, respectively.

For sample i, the online estimator uses the token-normalized velocity-prediction error

$$\ell_i = \frac{\sum_{l,c} w_{i,l} \left(v_{\theta,i,l,c} - v^{\star}_{i,l,c} \right)^2}{\max(D_z \sum_l w_{i,l}, 1)}, \tag{18}$$

where l and c index packed latent tokens and feature dimensions, $w_{i,l}$ is the valid-token weight, and D_z is the token feature dimensionality. Without a validity mask, $w_{i,l} = 1$. The corresponding timestep bin is $k_i = \min(\lfloor K t_i \rfloor, K - 1)$.

Distributed group–bin aggregation. Let $\mathcal{B}^{\circ}_r$ contain samples with finite estimator losses on worker r. Workers reduce group–bin loss sums and counts, giving

$$\bar{\ell}_{g,k} = \frac{\sum_r \sum_{i \in \mathcal{B}^{\circ}_r} \mathbf{1}[g_i = g,\ k_i = k]\, \ell_i}{\sum_r \sum_{i \in \mathcal{B}^{\circ}_r} \mathbf{1}[g_i = g,\ k_i = k]}, \tag{19}$$

whenever the denominator is positive. This is a sample-count-weighted global mean, not an unweighted mean of worker means. The first valid observation initializes both EMA tables; later observations follow Eq. 4. Cells with no observation in a microbatch retain their previous values, so the EMA clock counts observed microbatches rather than optimizer steps.

Stabilized difficulty and activity. Before computing the sampling utility, unseen bins are filled conservatively. If group g has observed bins $\mathcal{O}_g$, every unseen bin receives the mean observed slow-EMA value, $|\mathcal{O}_g|^{-1} \sum_{j \in \mathcal{O}_g} d^{\text{slow}}_{g,j}$, in both EMA tables; a wholly unobserved group receives the fallback value 1. The filled estimates are clamped below by ϵ, producing $\tilde{d}^{\text{fast}}_{g,k}$ and $\tilde{d}^{\text{slow}}_{g,k}$. Using the same fallback in both tables avoids inventing an artificial fast–slow discrepancy for unseen bins.

These stabilized estimates are the sampling-time quantities used in the main-text definition in Eq. 5. For completeness, the corresponding expressions are

$$D_{g,k} = \frac{\tilde{d}^{\text{slow}}_{g,k}}{K^{-1}\sum_j \tilde{d}^{\text{slow}}_{g,j}}, \qquad A_{g,k} = \min\left\{ \left| \frac{\tilde{d}^{\text{slow}}_{g,k} - \tilde{d}^{\text{fast}}_{g,k}}{\tilde{d}^{\text{slow}}_{g,k}} \right|, 1 \right\}. \tag{20}$$

$D_{g,k}$ is therefore relative to the difficulty scale of its own group. $A_{g,k}$ measures the magnitude of recent change relative to the slow estimate; because the absolute discrepancy is used, we refer to it as a *learning-activity* signal rather than a signed improvement signal.

For the ablation that retains only positive signed change, we first compute

$$r_{g,k} = \frac{\tilde{d}^{\text{slow}}_{g,k} - \tilde{d}^{\text{fast}}_{g,k}}{\tilde{d}^{\text{slow}}_{g,k}},$$

and use $A_{g,k} = \min(\max(r_{g,k}, 0), 1)$. The default absolute mode instead uses $A_{g,k} = \min(|r_{g,k}|, 1)$ as in Eq. 20.

Utility and numerical safeguards. The implementation uses the exact safeguarded utility

$$U_{g,k} = \max\left\{ (D_{g,k} + \epsilon)\left(\frac{1}{K} + A_{g,k}\right), \epsilon \right\}. \tag{21}$$

Equation 6 in the main text omits only this final lower-bound operator for readability. Non-finite relative changes are replaced by zero, non-finite utilities by one, and utilities are lower-bounded by ϵ before normalization. A wholly unobserved group therefore has constant utility and samples uniformly.

Bin and timestep sampling. After drawing a timestep bin from Eq. 7, the continuous timestep is sampled uniformly within that bin:

$$k \sim \text{Categorical}(p_s(\cdot \mid g)), \qquad t \mid k \sim \mathcal{U}\left[\frac{k}{K}, \frac{k+1}{K}\right). \tag{22}$$

Training progress is $s = \text{clip}(n/N, 0, 1)$, where n is the number of completed optimizer updates and N is the planned update budget. Equation 7 is exactly uniform for the first $1/K$ fraction of training and switches to the utility mixture afterward. There is no cosine interpolation at the boundary; subsequent changes in allocation are driven by the online EMA statistics. Probabilities are normalized once more for numerical stability.

Default settings and reproducibility. TMAS exposes only the timestep resolution K and the exploration weight λ. We use $K = 10$ and $\lambda = 0.2$, with $\epsilon = 10^{-6}$ as a numerical safeguard. The remaining quantities are determined by K: the fast and slow EMA coefficients are $1 - 1/K = 0.90$ and $1 - 1/K^2 = 0.99$, the utility offset is $1/K = 0.1$, the activity score is unit-bounded, and the initial uniform phase lasts $1/K = 0.1$ of training (6k steps under the 60k schedule). Thus, this reparameterization is exactly equivalent to the reported default configuration while avoiding independent tuning of these coefficients. With $\lambda = 0.2$, each bin retains probability at least 0.02 in exact arithmetic. Sampler state is detached from the optimization graph and synchronized across workers. Task/data sampling weights are held fixed across sampler ablations.

D.2 Ablation of Learning Activity and Exploration

Beyond task–modality-specific adaptation, TMAS contains two additional design choices. First, the learning-activity signal responds to both improving and degrading timestep regions through the absolute fast–slow EMA discrepancy. Second, uniform exploration is retained both before adaptive sampling begins and throughout the remaining training process. We isolate these choices to examine whether bidirectional learning-state changes and continued timestep exploration are beneficial for stable adaptive allocation.

Table 12: Ablation of learning activity and exploration in TMAS. Improvement-only activity rewards timestep regions whose recent error decreases, but does not assign an activity bonus when the error increases. No uniform warm-up enables adaptive sampling from the beginning of training, whereas no persistent exploration removes the uniform mixture after warm-up.

Variant	Bldg. ↑	Change ↑	Sem. ↑	Height ↑	Ref. ↑	Det. ↑	UTS ↑	Steps ↓
Improvement-only activity	76.79	74.47	52.24	73.06	73.77	64.50	69.14	42k
No uniform warm-up	76.58	74.29	51.96	72.84	73.51	64.22	68.90	49k
No persistent exploration ($\lambda = 0$)	76.71	74.38	52.08	72.96	73.64	64.36	69.02	45k
TMAS	**76.93**	**74.63**	**52.57**	**73.31**	**73.93**	**64.75**	**69.35**	**37k**

The comparison isolates two aspects of the adaptive sampler. Improvement-only activity tests whether the sampler should respond only to timestep regions that are currently improving, or also to regions whose errors are increasing. The latter may indicate forgetting or interference under joint training and therefore also warrant additional optimization. The two exploration ablations examine complementary stages of training: uniform warm-up provides broad initial coverage before the online estimates become reliable, while the persistent uniform mixture prevents timestep regions with underestimated utility from being permanently neglected. Together, these experiments evaluate whether TMAS benefits from both bidirectional learning-state awareness and continued exploration.

E Capacity and Efficiency

The main text focuses on predictive sharing and optimization. This section reports the practical cost of that sharing separately, including adapter capacity and wall-clock behavior.

Table 13: LoRA capacity and efficiency. Rank 32 is the default used in all other experiments and is highlighted in the Rank column. Bold metric values indicate the best result in each column.

Rank	Trainable params ↓	UTS ↑	Peak memory† ↓	Images/s‡ ↑
8	**12.3M**	67.24	**40.92 GiB**	**6.69**
16	24.6M	68.91	41.21 GiB	**6.69**
32	49.1M	69.35	41.53 GiB	**6.69**
64	98.3M	**69.37**	42.23 GiB	**6.69**

†Peak allocated GPU memory. ‡Steady-state training throughput. All efficiency measurements use the same hardware, software, and batch-size settings.

Table 13 reports the capacity–efficiency trade-off for the shared LoRA adapter. Increasing the rank from 8 to 32 raises UTS from 67.24 to 69.35, while doubling the rank from 32 to 64 yields only a 0.02-point gain (69.37). Trainable parameters grow from 12.3M to 98.3M and peak memory increases from 40.92 to 42.23 GiB, whereas the measured throughput remains unchanged at 6.69 images/s. Rank 32 is therefore used as the default operating point in the other experiments.

F Full Benchmark Comparisons

The main text uses Figure 4 as a compact visual summary of performance across the thirteen evaluation sources. For completeness, Tables 14 and 15 report detailed numerical comparisons, including venue–year information and dataset-specific notes.

Table 14: Benchmark comparison for change detection, semantic segmentation, and rotated detection. Gray cells denote results reported by prior work, and bold indicates the best result in each column.

	Change detection			Semantic segmentation			Rotated detection
Method	WHU-CD IoU ↑	LEVIR-CD+ IoU ↑	UrbanSAR mIoU ↑	LoveDA mIoU ↑	OEM-RGB mIoU ↑	OEM-SAR mIoU ↑	DIOR-R mAP50 ↑
Baselines evaluated on multiple datasets							
UniChange (CVPR'26)	90.41	**78.87**					
SkySense (CVPR'24)			59.30				74.27
MTP (JSTARS'24)				54.17			74.54
SelectiveMAE (ICCV'25)				53.92			71.75
FedSense (ICCV'25)				55.96			68.82
S5 (AAAI'26)				55.65	70.02		75.30
MetaSegNet (TGRS'24)				52.00	70.40		
Source-specific baselines							
UA-BCD (ISPRS JPRS'25)	86.87						
TriCDNet (ISPRS Arch.'26)	86.91						
ChangeRWKV-B (AAAI'26)		75.46					
ChangeCLIP (ISPRS JPRS'24)		75.63					
Mask2Former (CVPR'22)			**61.06**				
D2LS (ICCV'25)				55.30			
FT-U-NetFormer (WACV'23)					69.13		
VMamba (GRSM'25)						34.74	
SegEarth-OV (CVPR'25)						35.10	
U-Net-EffNet-B4 (GRSM'25)						35.13	
SegFormer (GRSM'25)						**35.77**	
MutDet (ECCV'24)							70.70
RHINO-DETR (WACV'25)							72.67
RingMoE (TPAMI'26)							**76.04**
UniFlow-RS	**90.88**	76.44	56.57	**56.30**	**71.28**	30.12	64.75

Table 15: Benchmark comparison for building extraction, height estimation, and referring segmentation.

	Building extraction		Height estimation		Referring segmentation			
Method	WHU IoU ↑	SpaceNet 6 SAR IoU ↑	Potsdam RMSE ↓	DFC23 δ_1 ↑	RRSIS-D cIoU ↑	RRSIS-D gIoU ↑	LaSeRS cIoU ↑	LaSeRS gIoU ↑
CCS (SAM2) (CVPR'25)	88.58							
DMBC-Net (Remote Sens.'21)	91.66							
UAGLNet (TGRS'26)	92.07							
UANet (TGRS'24)	**92.15**							
BIDNet (TGRS'24)		55.20						
CMH-CL (JSTARS'26)		55.25						
AEM-KD (JSTARS'26)		55.43						
HeightFormer (Remote Sens.'24)			2.193					
LUMNet (GRSL'24)			2.158					
RAEPHR-Net (JSTARS'25)			**2.018**					
DoFA (arXiv'24)				30.99				
LIGHT (IGARSS'23)				51.10				
MaRS (AAAI'26)				54.04				
RMSIN (CVPR'24)					77.79	64.20		
RS2-SAM2 (AAAI'26)					78.99	66.72		
M^2A-7B (ICLR'25)							29.30	30.50
GeoPixel-8B (ICML'25)							55.20	50.40
SegEarth-R2-3B (CVPR'26)						67.90	67.90	57.20
UniFlow-RS	89.14	**64.72**	2.121	**73.31**	**80.11**	**70.33**	**82.13**	**77.53**

SpaceNet 6 specialists use paired optical imagery during training but are evaluated with SAR at test time; UniFlow-RS does not use SpaceNet 6 optical imagery during training. LaSeRS baseline results for M^2A-7B, GeoPixel-8B, and SegEarth-R2-3B are taken from Table 2 of SegEarth-R2 (Xin et al., 2026); that source reports metrics as gIoU/cIoU, and we reorder them here to match the cIoU/gIoU column order.

G Additional Qualitative Comparisons

Figure 11 provides additional examples across all six task families, complementing the numerical results in Appendix F. Each column shows the same input and target with two representative baselines and UniFlow-RS.

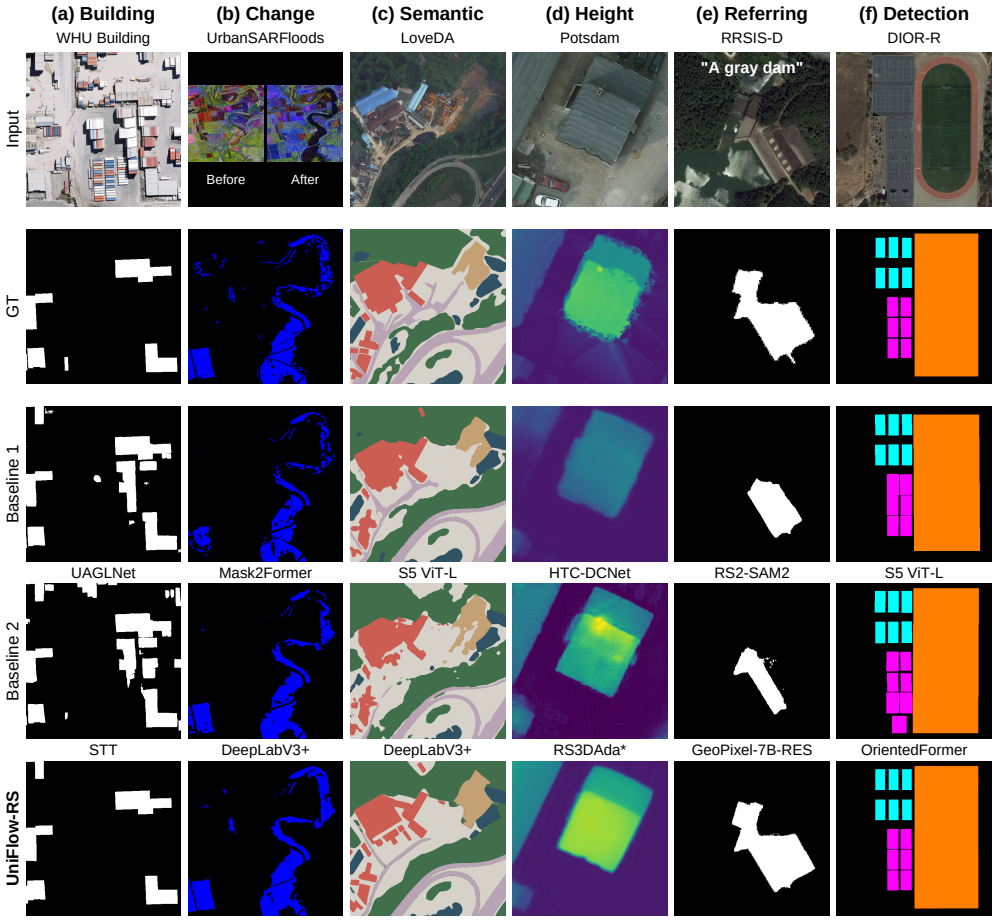


Figure 11: **Additional qualitative comparisons across six tasks.** Rows show the input, ground truth, two baseline predictions, and UniFlow-RS. Baseline methods are labeled below each prediction and cited in the text.

Compared methods. We compare with UAGLNet (Yao et al., 2026) and STT (Chen et al., 2021) for building extraction; Mask2Former (Cheng et al., 2022a) and DeepLabV3+ (Chen et al., 2018a) for flood change mapping; S5 ViT-L (Lv et al., 2026) and DeepLabV3+ for LoveDA; HTC-DCNet (Chen et al., 2023) and RS3DAda (Song et al., 2024) for height estimation; RS2-SAM2 (Rong et al., 2026) and GeoPixel-7B-RES (Shabbir et al., 2025) for referring segmentation; and S5 ViT-L and OrientedFormer-R50 (Zhao et al., 2024a) for rotated detection.

Visual observations. UniFlow-RS shows fewer spurious responses in building extraction, follows the curved flooded region in SAR change mapping, and recovers more of the queried gray dam in referring segmentation. It also preserves clearer roof-height variation and separation between adjacent sports facilities. Semantic differences mainly occur around building and land-cover boundaries. Overall, the shared predictor preserves task-specific spatial structures across heterogeneous tasks.