CROSSFREQ-UHR: CONTEXT-ALIGNED FREQUENCY LEARNING FOR ULTRA-HIGH-RESOLUTION OBJECT DETECTION

Anonymous authors
Paper under double-blind review

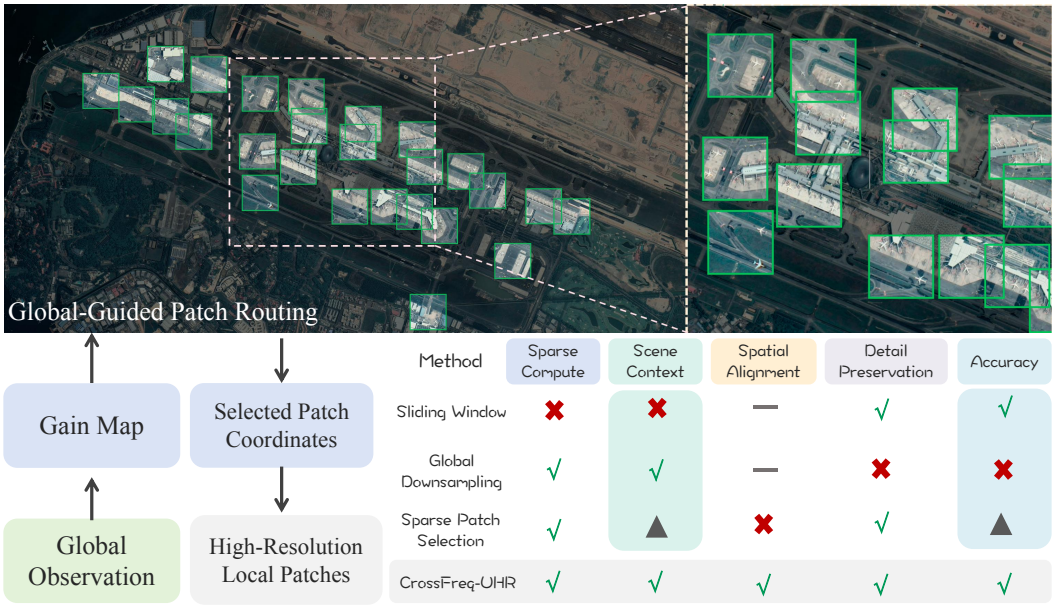

Figure 1: Motivation and overview of CrossFreq-UHR. The upper panel shows inherited global-guided routing for selecting high-resolution local patches. The lower panel compares representative UHR detection paradigms, highlighting that CrossFreq-UHR combines sparse computation, spatially aligned scene context, and local-detail preservation. Symbols denote supported, absent, partial, or inapplicable properties.

ABSTRACT

Sparse global–local detectors improve ultra-high-resolution object detection by allocating high-resolution computation to a limited number of informative regions. However, their global observations are mainly used for region selection and downstream decoding, leaving selected patches to form their multi-scale representations without location-specific scene context before aggregation. We refer to this limitation as the pre-aggregation representation bottleneck. To address it, we propose CrossFreq-UHR, a context-aware frequency learning framework for sparse UHR detection. CrossFreq-UHR maps global features to the coordinate system of each selected patch to provide spatially aligned scene context, and decomposes local features into a smoothed structural representation and a detail-oriented residual. An asymmetric interaction mechanism conditions the structural path on aligned global context while independently refining patch-specific detail, introducing scene awareness without directly perturbing local fine-scale information. The enhanced representations are subsequently aggregated for query-based global–local detection. Experiments on SODA-A and STAR demonstrate that the proposed pre-aggregation conditioning consistently improves detection accuracy over the sparse global–local baseline with modest additional computation.

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

Ultra-high-resolution (UHR) remote-sensing imagery combines broad spatial coverage with the fine-grained visual evidence required to recognize small and densely distributed objects (Xia et al., 2018). This capability is valuable for aerial observation, yet it creates a fundamental accuracy–efficiency conflict (Yang et al., 2022). Processing an entire image at its native resolution is computationally expensive, uniform downsampling weakens small-object evidence, and exhaustive sliding-window detection introduces substantial redundant computation while fragmenting scene context (Cheng et al., 2023; Gao et al., 2018; Li et al., 2026). Considerable progress has been made by allocating high-resolution computation selectively (Li et al., 2024). Coarse-to-fine and cluster-based detectors first identify promising regions and then apply expensive local detection only to those regions (Gao et al., 2018; Yang et al., 2019). More recently, sparse global–local detectors use a low-resolution observation of the complete scene to select a limited set of high-resolution patches and combine global and local information during detection (Li et al., 2026).

Fig. 1 illustrates the sparse observation process adopted by CrossFreq-UHR and summarizes its motivation. As shown in the upper panel, a low-resolution global observation is used to predict a gain map, from which informative patch coordinates are selected to extract a sparse set of high-resolution local patches. This inherited routing mechanism determines where high-resolution computation should be allocated and reduces the need for exhaustive processing of the complete image. The lower panel highlights the remaining representational trade-off. Sparse patch selection preserves local detail and reduces redundant computation, but it does not explicitly align scene-level context with each selected patch when its local representation is formed. Existing sparse global–local designs instead use global information primarily for region selection and downstream global–local decoding (Li et al., 2026). Consequently, how the global observation should condition each selected patch before multi-scale aggregation remains insufficiently explored.

We refer to this missing connection as the pre-aggregation representation bottleneck. Scene information is available to the detector, but is not explicitly aligned with the selected local region when its multi-scale features are formed. Controlled context interventions support this diagnosis (see Fig. 5). A generic global-context fusion operator provides only a limited improvement, while replacing spatially aligned context with an averaged or spatially shifted alternative consistently degrades detection. Further ablations show that applying the same global conditioning to both semantic structure and spatial detail is also suboptimal. These observations motivate two requirements for local feature construction. Global context should preserve its spatial correspondence with each patch, and semantic structure and fine detail should follow distinct interaction paths.

To address this bottleneck, we propose **CrossFreq-UHR**, a context-aware frequency learning framework for sparse UHR object detection. As illustrated in Fig. 2, the overall detector follows a sparse global–local pipeline. A downsampled observation of the complete image is encoded into a global feature pyramid, from which a gain map is predicted to select a limited set of informative high-resolution patches. A local backbone then extracts multi-scale features from the selected patches. CrossFreq-UHR blocks are inserted at the intermediate feature levels to condition these local representations with spatially corresponding global context before multi-scale aggregation. The enhanced local features, together with the global representation, are subsequently processed by the inherited hybrid encoder and global–local query decoder to produce detections in the full-image coordinate system.

Fig. 3 presents the internal design of a CrossFreq-UHR block. It comprises three coordinated components. Spatially Aligned Context Mapping (SACM) uses the selected patch coordinates to extract location-specific context from the global representation. Explicit Frequency Decomposition (EFD) separates the corresponding local feature into low-frequency semantic structure and high-frequency spatial detail. Asymmetric Semantic–Detail Interaction (ASDI) then conditions the semantic path on the aligned global context while independently refining the local detail path. The outputs of the two paths are fused to produce an enhanced local representation. Through this design, CrossFreq-UHR introduces scene awareness during local feature construction while preserving the boundaries and textures required for precise small-object detection. Extensive experiments on SODA-A (Cheng et al., 2023) and STAR (Li et al., 2025) demonstrate that CrossFreq-UHR consistently improves sparse global–local detection across different UHR aerial benchmarks. Our contributions are summarized as follows:

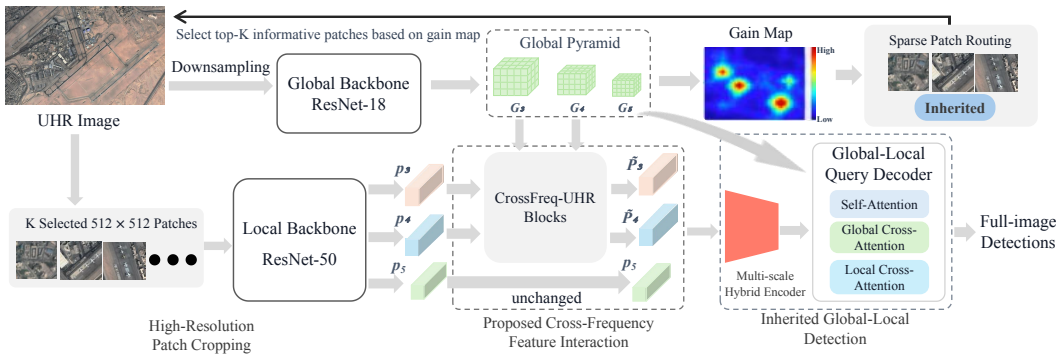

Figure 2: Overview of the CrossFreq-UHR detection pipeline. The global pyramid guides sparse high-resolution patch routing, while CrossFreq-UHR enhances P_3 and P_4 with spatially aligned context from G_3 and G_4. The enhanced feature pyramid is processed by the inherited encoder and query decoder for full-image detection.

- We identify a pre-aggregation representation bottleneck in sparse UHR detection: global information determines where to allocate high-resolution computation, but does not sufficiently condition how each selected local region is represented before feature aggregation. We experimentally characterize the need for spatially corresponding and branch-specific global conditioning.
- We propose CrossFreq-UHR, a unified context-aware frequency learning framework comprising Spatially Aligned Context Mapping, Explicit Frequency Decomposition, and Asymmetric Semantic–Detail Interaction. Together, the three components introduce scene-level guidance into local semantic representation while preserving high-resolution detail.
- Experiments on SODA-A and STAR demonstrate improved detection accuracy over the sparse global–local baseline, while controlled ablations on SODA-A support the effectiveness of the proposed components.

2 Related Work

Efficient UHR and Tiny Object Detection. Detecting small, densely distributed objects in UHR aerial imagery requires balancing spatial detail against computational cost (Xia et al., 2018; Cheng et al., 2023). Coarse-to-fine and region-selective methods reduce redundant processing by allocating computation to informative regions (Gao et al., 2018; Yang et al., 2019; Du et al., 2023; Zhang et al., 2023; Chen et al., 2022). QueryDet sparsely computes high-resolution features around predicted object locations (Yang et al., 2022), while SparseFormer uses attentive tokens to select informative windows (Li et al., 2024). UHR-DETR further integrates sparse patch allocation with global–local decoding for UHR detection (Li et al., 2026). Complementing these allocation mechanisms, CrossFreq-UHR investigates how spatially corresponding scene context should condition selected patch representations before multi-scale aggregation.

Context and Frequency-Aware Representation Learning. Context modeling enriches local features through long-range dependencies and scene relationships (Wang et al., 2018; Liu et al., 2018). Global–local approaches such as GLNet and locality-aware contextual correlation further connect local predictions with broader scene information (Chen et al., 2019a; Li et al., 2021). Frequency-aware methods distinguish spatial variations through multi-frequency representations, spectral channel descriptors, and adaptive feature processing (Chen et al., 2019b; Qin et al., 2021; Chen et al., 2024b;a). For remote sensing, SET suppresses distracting high-frequency responses in tiny-object detection (Sun et al., 2025), while F2Net employs high- and low-frequency pathways for UHR segmentation (Chen et al., 2025). CrossFreq-UHR combines spatially aligned scene context with explicit local frequency decomposition, conditioning the structural path on global context while independently refining local detail.

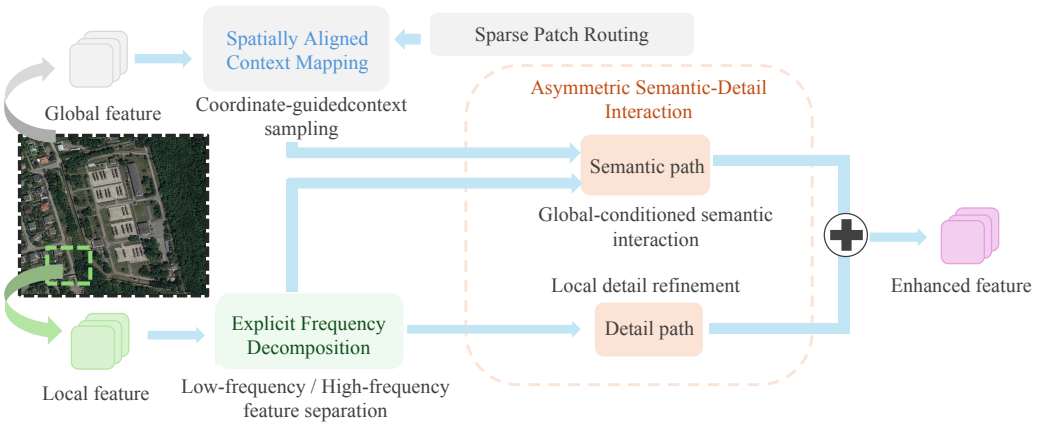

Figure 3: Simplified CrossFreq-UHR block. SACM extracts spatially aligned global context, EFD constructs structural and residual features, and ASDI processes them through context-conditioned structural and local-detail paths before output fusion.

3 Motivation

3.1 Pre-Aggregation Representation Bottleneck

Sparse global–local detectors combine a low-resolution observation of the complete scene with a sparse set of high-resolution local patches. Let $\mathbf{I} \in \mathbb{R}^{3\times H_I\times W_I}$ denote the detector input after image-level preprocessing, where H_I and W_I are its spatial dimensions. The global branch receives

$$\mathbf{I}_g = \mathrm{Down}_4(\mathbf{I}), \tag{1}$$

where Down_4 denotes four-times spatial downsampling. Let $\Lambda = \{3, 4, 5\}$ denote the pyramid levels. The global encoder produces

$$\{\mathbf{G}^\ell\}_{\ell\in\Lambda} = E_g(\mathbf{I}_g), \qquad \mathbf{G}^\ell \in \mathbb{R}^{d\times H_g^\ell\times W_g^\ell}, \tag{2}$$

where d is the projected channel dimension and $H_g^\ell \times W_g^\ell$ is the spatial resolution of global level ℓ.

The inherited gain-map head predicts a routing map $\mathbf{R}_{\text{gain}}$, from which the sparse router selects K patch windows:

$$\mathbf{R}_{\text{gain}} = \mathcal{H}_{\text{gain}}(\mathbf{G}^5), \qquad \mathcal{W}_K = \mathcal{Q}_{\text{route}}\left(\mathbf{R}_{\text{gain}}, K\right) = \{\mathbf{b}_k\}_{k=1}^{K}. \tag{3}$$

Here K is the patch budget and $\mathbf{b}_k = (x_k, y_k, W_p, H_p)$ specifies the top-left position and fixed spatial extent of the k-th patch in the coordinate frame of $\mathbf{I}$. Because a selected window may intersect padded image regions, $\mathrm{CropPad}$ denotes cropping followed by boundary padding:

$$\mathbf{I}_k = \mathrm{CropPad}(\mathbf{I}, \mathbf{b}_k), \qquad \{\mathbf{F}_k^\ell\}_{\ell\in\Lambda} = E_{\text{loc}}(\mathbf{I}_k), \tag{4}$$

where $\mathbf{I}_k \in \mathbb{R}^{3\times H_p\times W_p}$ and $\mathbf{F}_k^\ell \in \mathbb{R}^{d\times h_\ell\times w_\ell}$ is the projected local feature at level ℓ.

In the inherited baseline, global information determines which patches are selected and participates in downstream decoding. However, conditional on a selected window $\mathbf{b}_k$, the local representation in Eq. 4 is constructed only from $\mathbf{I}_k$. The global feature associated with the same spatial region does not explicitly condition the local feature before multi-scale aggregation. We refer to this missing connection as the pre-aggregation representation bottleneck. CrossFreq-UHR addresses this bottleneck at levels 3 and 4 through

Table 1: Comparison with state-of-the-art methods on the STAR. † denotes global downsampling to 1024×1024.

ID	Method	Backbone	Car	Boat	Tank	Truck	BB	Crane	Airplane	LT	mAP	AP_S	AP_M	AP_L	Lat. (s)	Mem. (GB)
Global Downsampling																
1	RT-DETR†	ResNet-50	0.69	1.09	16.86	1.04	4.42	12.04	21.82	12.96	8.86	0.12	5.80	16.68	0.444	0.69
2	Fovea†	ResNet-50	0.00	0.00	5.20	0.00	0.00	7.80	9.70	11.20	4.20	0.00	0.50	11.30	3.910	2.92
Sliding-Window UHR Detection																
3	FCOS	ResNet-50	24.75	6.68	28.81	3.73	27.12	13.09	64.13	35.94	25.50	13.20	25.80	25.10	2.330	1.96
4	RT-DETR	ResNet-50	28.29	12.31	29.44	6.08	40.46	11.83	63.93	31.70	28.00	11.46	28.99	24.66	2.391	1.62
5	CEASC	ResNet-18	22.50	6.50	29.30	3.10	29.50	14.30	68.10	37.90	26.40	11.60	26.70	25.00	6.110	0.29
Patch-based UHR Detection																
6	SPDet	CSPDarknet	24.67	4.86	39.26	6.11	34.14	17.18	70.49	34.82	28.94	10.66	27.42	33.88	6.397	2.84
7	GigaDet	ResNet-50	9.78	0.27	29.03	0.97	11.78	9.38	52.17	6.66	15.00	3.10	13.86	19.38	0.680	1.06
8	UHR-DETR	ResNet-50	30.40	6.30	30.80	10.00	39.80	18.80	59.40	33.30	28.62	12.11	28.67	28.42	0.371	3.13
9	**CrossFreq-UHR (Ours)**	ResNet-50	35.80	7.40	35.10	11.80	43.10	18.70	59.00	33.00	30.47	14.37	29.99	31.38	0.374	3.50

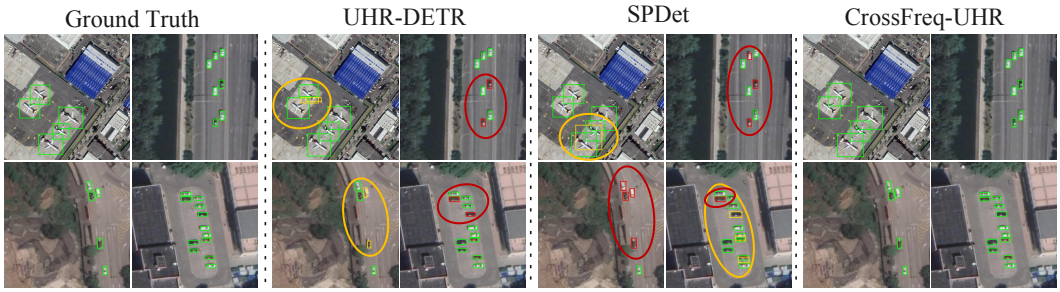

Figure 4: Qualitative comparison on STAR. Red and yellow ellipses highlight missed detections and localization or duplicate-detection errors, respectively.

$$\mathbf{C}_k^\ell = \mathcal{M}_\ell\left(\mathbf{G}^\ell, \mathbf{b}_k\right), \qquad \widetilde{\mathbf{F}}_k^\ell = \mathcal{X}_\ell\left(\mathbf{F}_k^\ell, \mathbf{C}_k^\ell\right), \qquad \ell \in \{3,4\}, \tag{5}$$

where $\mathbf{C}_k^\ell \in \mathbb{R}^{d\times h_\ell \times w_\ell}$ is the spatially aligned global context and $\widetilde{\mathbf{F}}_k^\ell$ is the enhanced local feature of the same shape. The coarsest local level is passed through unchanged:

$$\widetilde{\mathbf{F}}_k^5 = \mathbf{F}_k^5. \tag{6}$$

The mapping operator $\mathcal{M}_\ell$ determines how the global feature is aligned with the selected patch, while the interaction operator $\mathcal{X}_\ell$ determines how the aligned context modifies the local representation. This formulation exposes two design questions: how to preserve patch-specific spatial correspondence and which local responses should receive global conditioning.

4 CrossFreq-UHR

4.1 Framework Overview

Figs. 2 and 3 illustrate the detection pipeline and CrossFreq-UHR block, respectively. Following Sec. 3.1, k indexes a selected patch and $\ell \in \Lambda = \{3,4,5\}$ indexes a pyramid level. The global and local features are projected to $d = 256$ channels. The local feature $\mathbf{F}_k^\ell \in \mathbb{R}^{d\times h_\ell \times w_\ell}$ has spatial dimensions (h_ℓ, w_ℓ) and stride $s_3 = 8$, $s_4 = 16$, or $s_5 = 32$ relative to the input patch. These levels correspond to P_3, P_4, P_5 in the pipeline.

At levels 3 and 4, CrossFreq-UHR aligns the corresponding global feature with the patch window $\mathbf{b}_k$, decomposes the local feature, and applies asymmetric interaction before multi-scale aggregation. The aligned context $\mathbf{C}_k^\ell$ and enhanced output $\widetilde{\mathbf{F}}_k^\ell$ share the shape of $\mathbf{F}_k^\ell$; level 5 remains unchanged, $\widetilde{\mathbf{F}}_k^5 = \mathbf{F}_k^5$. The enhanced pyramid enters the inherited hybrid encoder and global–local query decoder, which also receives $\mathbf{G}^5$. Predictions in the preprocessed-image coordinate frame are

Table 2: Comparison with state-of-the-art methods on SODA-A. † denotes global downsampling to 1024×1024.

ID	Method	Backbone	AI	HE	SV	LV	SH	CO	ST	SP	WI	mAP	AP_S	AP_M	AP_L	Lat. (s)	Mem. (GB)
Global Downsampling																	
1	RT-DETR†	ResNet-50	29.0	23.4	2.3	4.8	8.1	19.0	11.1	48.0	19.0	18.3	1.2	21.0	34.3	0.036	0.46
2	Fovea†	ResNet-50	5.0	0.0	0.0	2.5	1.0	1.0	3.5	2.8	1.5	1.9	0.3	2.0	4.6	2.100	2.53
Patch-based UHR Detection																	
3	SPDet	CSPDarknet	64.5	8.2	31.9	20.9	42.8	38.8	55.1	70.8	18.0	39.0	13.1	48.5	39.3	1.957	0.75
4	GigaDet	ResNet-50	68.8	16.1	45.9	10.5	66.5	31.2	45.0	21.4	13.6	35.5	23.5	40.2	25.6	0.588	6.72
5	UHR-DETR	ResNet-50	83.8	53.4	45.9	20.8	56.8	36.3	69.4	51.4	61.9	53.3	30.9	59.4	40.8	0.647	5.57
6	**CrossFreq-UHR (Ours)**	ResNet-50	84.5	56.8	47.1	26.6	59.9	43.9	70.6	52.2	63.1	56.1	31.2	62.9	53.5	0.676	6.03

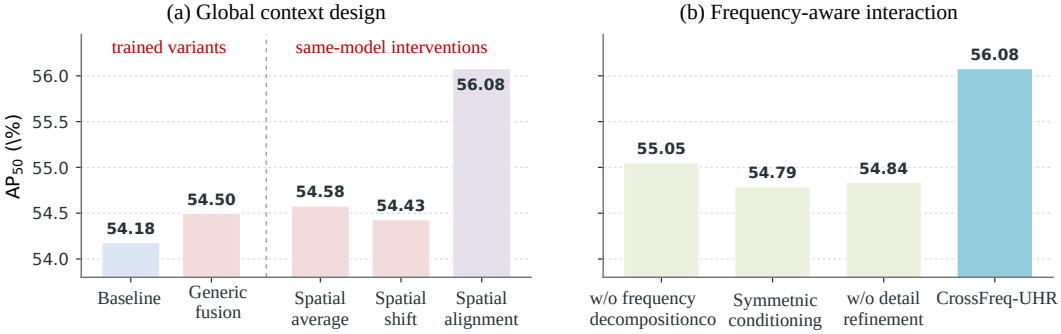

Figure 5: Evidence for the pre-aggregation representation bottleneck on SODA-A. **(a)** Independently trained context-fusion baselines and same-checkpoint spatial-mapping interventions. **(b)** Independently trained frequency-interaction ablations.

mapped back to source-image coordinates when required. The complete formulation, pseudocode, and training objective appear in Appendices D.1 and D.2.

4.2 Spatially Aligned Context Mapping

For $\ell \in \{3, 4\}$, SACM samples location-specific context from the projected global feature $\mathbf{G}^\ell \in \mathbb{R}^{d \times H_g^\ell \times W_g^\ell}$, where H_g^ℓ and W_g^ℓ are its spatial dimensions. The mapping Γ_ℓ converts the patch window $\mathbf{b}_k$ into a normalized horizontal–vertical sampling grid of shape $h_\ell \times w_\ell \times 2$. With fixed channel-wise 5×5 Gaussian smoothing $\mathcal{S}$, the aligned context is

$$\begin{aligned} \mathbf{C}_k^\ell &= \mathcal{M}_\ell(\mathbf{G}^\ell, \mathbf{b}_k) \\ &= \mathbf{V}_k^\ell \odot \mathrm{GridSample}\big(\mathbf{V}_g^\ell \odot \mathcal{S}(\mathbf{G}^\ell), \Gamma_\ell(\mathbf{b}_k)\big)\,. \end{aligned} \tag{7}$$

Here $\mathbf{V}_g^\ell \in \{0,1\}^{H_g^\ell \times W_g^\ell}$ and $\mathbf{V}_k^\ell \in \{0,1\}^{h_\ell \times w_\ell}$ mark valid, unpadded locations on the global and local feature grids. The operator $\odot$ denotes element-wise multiplication with channel broadcasting, and $\mathrm{GridSample}$ performs bilinear sampling. The masks zero invalid source locations after smoothing and invalid output locations after sampling. Unlike a broadcast scene descriptor, the sampled context retains spatial variation corresponding to the patch's position and extent. Coordinate, mask, and boundary conventions are specified in Appendix D.3.

4.3 Explicit Frequency Decomposition

EFD constructs a smoothed structural feature $\mathbf{L}_k^\ell$, its coarse reconstruction $\overline{\mathbf{L}}_k^\ell$, and a local residual $\mathbf{H}_k^\ell$:

$$\mathbf{L}_k^\ell = \mathcal{S}(\mathbf{F}_k^\ell), \qquad \overline{\mathbf{L}}_k^\ell = \mathcal{U}_r\big(\mathcal{A}_r(\mathbf{L}_k^\ell)\big)\,, \tag{8}$$

$$\mathbf{H}_k^\ell = \mathbf{F}_k^\ell - \overline{\mathbf{L}}_k^\ell. \tag{9}$$

Here $\mathcal{A}_r$ is average pooling with kernel and stride r, and $\mathcal{U}_r$ bilinearly resizes its output to (h_ℓ, w_ℓ); the default scale is $r = 2$. Gaussian smoothing attenuates rapid spatial variation, while pooling

Table 3: EFD reconstruction-scale sensitivity on SODA-A. Only r varies; all other settings remain fixed.

Scale r	AP_{50}	AP^{S}_{50}	AP^{M}_{50}	AP^{L}_{50}	AP	AP_{75}	AR
1	54.60	32.33	61.27	40.80	29.68	27.90	43.56
2	56.08	31.16	62.91	53.51	29.47	26.68	43.22
4	**56.49**	**32.85**	**63.32**	**50.99**	**31.09**	**29.60**	**44.36**

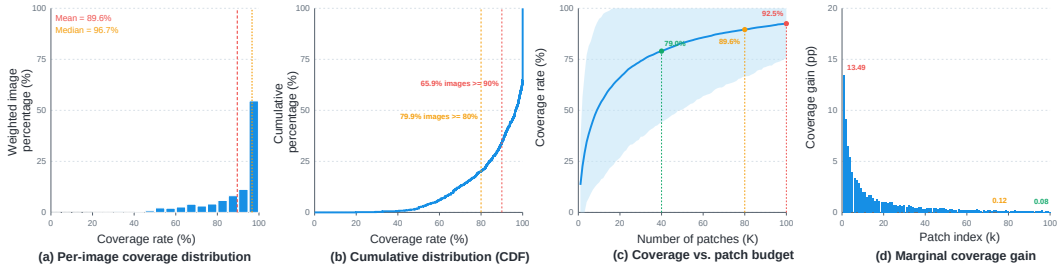

Figure 6: Routing coverage on SODA-A and STAR validation sets with equal dataset weighting, measured by mean per-instance area coverage within each non-empty image. **(a–b)** Coverage distribution and CDF at $K = 80$. **(c)** Mean coverage versus patch budget with weighted 10th–90th percentiles. **(d)** Marginal gain per additional patch.

and interpolation form a coarser reference. Subtracting this reference exposes deviations that may include object boundaries, texture, and background detail. The subsequent detail path therefore refines the residual rather than treating every residual response as useful object evidence.

The structural path receives the full-resolution smoothed feature $\mathbf{L}^{\ell}_{k}$, while $\overline{\mathbf{L}}^{\ell}_{k}$ serves only as the reference for residual construction. This retains a dense structural input for contextual interaction and exposes local variations not retained by the coarse reconstruction. The exact identity is $\mathbf{F}^{\ell}_{k} = \overline{\mathbf{L}}^{\ell}_{k} + \mathbf{H}^{\ell}_{k}$, rather than a sum of the two interaction-path inputs. Structure and detail describe intended roles, not exact semantic separation or disjoint frequency bands. Appendix D.4 provides the corresponding analysis.

4.4 Asymmetric Semantic–Detail Interaction

The downsampled global observation provides scene context but cannot retain all fine-scale evidence in a local patch. ASDI therefore conditions the structural feature on aligned context through the semantic-path operator Φ^{ℓ}_{sem}, while the detail operator $\Phi^{\ell}_{\text{detail}}$ independently refines the local residual:

$$\mathbf{S}^{\ell}_{k} = \Phi^{\ell}_{\text{sem}}(\mathbf{L}^{\ell}_{k}, \mathbf{C}^{\ell}_{k}), \qquad \mathbf{D}^{\ell}_{k} = \Phi^{\ell}_{\text{detail}}(\mathbf{H}^{\ell}_{k}), \tag{10}$$

$$\widetilde{\mathbf{F}}^{\ell}_{k} = \mathcal{X}_{\ell}(\mathbf{F}^{\ell}_{k}, \mathbf{C}^{\ell}_{k}) = \mathbf{S}^{\ell}_{k} + \mathbf{D}^{\ell}_{k}, \quad \ell \in \{3, 4\}. \tag{11}$$

The structural operator concatenates structure and context along channels, projects the resulting $2d$ channels to d, and applies grouped interaction. The detail operator uses local residual refinement with channel and directional spatial gates. Their outputs, $\mathbf{S}^{\ell}_{k}$ and $\mathbf{D}^{\ell}_{k}$, both preserve the local feature shape, making their element-wise addition well-defined. Together with EFD, these operators define the interaction mapping $\mathcal{X}_{\ell}$. For fixed local features and parameters, the detail path has no direct dependence on the supplied global context. Internal operators and the formal dependence analysis are provided in Appendix D.5.

5 Experiments

5.1 Experimental Setups

Datasets and metrics. We evaluate CrossFreq-UHR on SODA-A (Cheng et al., 2023) and STAR (Li et al., 2025), containing nine and eight categories, respectively. The prepared images have resolu-

Table 4: CrossFreq-UHR placement at P_3 and P_4 on SODA-A under identical training and evaluation settings.

P3	P4	AP_{50}	AP^{S}_{50}	AP^{M}_{50}	AP^{L}_{50}	AP	AP_{75}	AR
✓	✗	52.35	30.57	58.86	42.94	27.49	24.91	41.57
✗	✓	56.06	**31.44**	62.72	49.12	28.93	25.54	42.56
✓	✓	**56.08**	31.16	**62.91**	**53.51**	**29.47**	**26.68**	**43.22**

Table 5: Patch-budget sensitivity on SODA-A with matched budgets of K 512×512 patches. Latency is end-to-end per image on two GPUs; memory is peak usage per GPU. Gray marks the default setting.

K	Baseline AP_{50}	CrossFreq-UHR AP_{50}	ΔAP_{50}	AP^{S}_{50}	AP^{M}_{50}	AP^{L}_{50}	Latency (s/image)	Memory (GB/GPU)
40	49.00	50.06	+1.06	29.11	55.89	45.79	0.292	6.60
60	52.63	54.07	+1.44	30.68	60.63	49.35	0.344	7.85
80	54.18	56.08	+1.90	**31.16**	62.91	53.51	0.406	8.70
100	**54.55**	**56.48**	**+1.93**	31.15	**63.35**	**56.05**	0.476	10.42

tions of 9600×9600 for SODA-A and 8192×8192 for STAR; preprocessing details are provided in Appendix B. Predictions are restored to full-image coordinates and evaluated against horizontal bounding-box annotations. The primary metric is category-averaged AP_{50}, with category-wise and small-, medium-, and large-object results also reported. Selected ablations additionally report AP averaged over IoU thresholds from 0.50 to 0.95, AP_{75}, and average recall (AR). Efficiency is measured by full-image inference latency and peak GPU memory.

Implementation details. CrossFreq-UHR adopts a sparse global–local detection architecture with a ResNet-18 global encoder and a ResNet-50 local encoder. The global branch processes a four-times downsampled observation and predicts a gain map for selecting high-resolution regions, while the local branch extracts features from the selected 512×512 patches. All pyramid features are projected to 256 channels. CrossFreq-UHR blocks enhance the P_3 and P_4 levels with spatially aligned global context, while P_5 remains unchanged. The enhanced local features are subsequently processed by the multi-scale encoder and global–local query decoder. The default reconstruction scale is $r = 2$, with patch budgets of 80 for SODA-A and 40 for STAR. All architectural ablations use matched initialization, training schedules, and evaluation settings.

5.2 Main Results

Results on SODA-A. As shown in Tab. 2, CrossFreq-UHR achieves 56.1 AP_{50}, outperforming the listed UHR-DETR baseline by 2.8 percentage points. Improvements extend across all nine categories and all three size groups, with particularly strong gains for medium and large objects. These results support introducing scene context during local representation construction, with modest additional latency and memory under the reported protocol.

Results on STAR. Tab. 1 shows that CrossFreq-UHR achieves 30.47 AP_{50}, improving over UHR-DETR by 1.85 percentage points and obtaining the highest overall accuracy among the compared methods. It improves all three size-specific metrics, with notable category-wise gains for cars, tanks, and boarding bridges. Although gains vary across categories, the overall improvement is obtained with similar reported latency and a moderate memory increase.

Qualitative comparison. Fig. 4 presents representative STAR predictions. CrossFreq-UHR recovers objects missed by the compared methods and reduces localization and duplicate-detection errors in the highlighted regions, consistent with the quantitative results.

Routing coverage. Fig. 6 evaluates access to annotated object regions through sparse routing, with equal weighting of the two validation sets. Coverage increases with the patch budget but exhibits diminishing marginal gains. At the common analysis budget of 80 patches, mean coverage reaches 89.58%. This diagnostic characterizes the regions available for local processing; it does not change the dataset-specific inference budgets or directly measure detection accuracy.

Table 6: Component ablation on SODA-A under identical training and evaluation settings. With SACM and EFD retained, disabling ASDI replaces asymmetric interaction with symmetric conditioning.

Spatially Aligned Context Mapping	Explicit Frequency Decomposition	Asymmetric Semantic–Detail Interaction	Params (M)	AP_{50}	AP^{S}_{50}	AP^{M}_{50}	AP^{L}_{50}
✗	✗	✗	56.45	54.18	30.61	61.01	43.81
✓	✗	✗	57.56	55.05	**31.44**	61.77	45.73
✗	✓	✗	57.56	54.66	30.28	61.42	46.44
✓	✓	✗	57.56	54.79	30.74	61.76	45.61
✓	✓	✓	**57.56**	**56.08**	31.16	**62.91**	**53.51**

5.3 Ablation Studies

Component contributions. Tab. 6 evaluates SACM, EFD, and ASDI under a controlled SODA-A protocol. Aligned context and frequency decomposition each improve over the baseline, while the complete asymmetric design achieves the strongest overall result. Its advantage over symmetric conditioning supports processing structure and detail through distinct paths. The enhanced variants have the same parameter count, indicating that their performance differences arise from the interaction design rather than model size alone.

Controlled context interventions. Fig. 5(a) compares independently trained baseline and generic-fusion models with same-checkpoint mapping interventions. Replacing aligned context with spatially averaged or shifted alternatives degrades performance, demonstrating the trained model's reliance on spatial correspondence. The independently trained interaction variants in panel (b) further support asymmetric conditioning. Together, these comparisons substantiate the two design choices underlying pre-aggregation enhancement.

Reconstruction scale. Tab. 3 shows that the reconstruction scale affects both overall accuracy and performance across object sizes. The larger scale $r = 4$ achieves the best overall AP_{50} and AP, whereas the default $r = 2$ performs best on large-object AP_{50}. The results indicate a scale-dependent trade-off rather than a universally optimal decomposition setting.

Feature-pyramid placement. Tab. 4 shows that conditioning P_4 alone is more effective than conditioning P_3 alone. Joint placement provides little additional AP_{50} over P_4 alone, but improves AP, AP_{75}, and large-object performance. Its benefit is therefore more apparent under stricter localization criteria and for larger objects.

Patch budget. As shown in Tab. 5, CrossFreq-UHR consistently outperforms the baseline at matched patch budgets, indicating that its gains do not simply result from processing more patches. Increasing the budget beyond the default yields limited additional accuracy while increasing latency and memory. This trend supports the default SODA-A budget as an accuracy–efficiency compromise and is consistent with the diminishing coverage gains in Fig. 6.

6 Conclusion

In this work, we identify a pre-aggregation representation bottleneck in sparse UHR detection: global information guides patch selection and downstream decoding but does not explicitly condition local feature construction with spatially corresponding scene context. To address this bottleneck, we propose CrossFreq-UHR, a context-aware frequency learning framework. Spatially Aligned Context Mapping supplies location-specific global context for each selected patch. Explicit Frequency Decomposition constructs a smoothed structural feature and a local residual, enabling differentiated processing of structure and detail. Asymmetric Semantic–Detail Interaction then conditions the structural path on aligned context while independently refining local detail, producing enhanced features for multi-scale aggregation. Experiments on SODA-A and STAR demonstrate improved detection accuracy over the sparse global–local baseline, while controlled ablations support the effectiveness of spatial alignment and asymmetric frequency-aware interaction.

AI USE STATEMENT

Generative AI tools were used only for language polishing, grammar correction, and improving the clarity of the manuscript. They were not used to generate experimental results, perform data analysis, or make scientific claims. All technical content, experiments, and conclusions were produced and verified by the authors.

ETHICS STATEMENT

This work uses publicly available remote-sensing benchmarks and does not involve human subjects, private personal information, or newly collected sensitive data. The study is intended to improve the accuracy and efficiency of object detection in ultra-high-resolution aerial imagery. Such technology may support applications including environmental observation, infrastructure inspection, and disaster response, while also having potential dual-use implications in large-scale monitoring. Users should therefore consider applicable privacy requirements, data-governance policies, and deployment contexts when applying the proposed method.

REPRODUCIBILITY STATEMENT

The main paper and supplementary materials provide mathematical definitions, end-to-end formulations, forward-pass pseudocode, training objectives, dataset preparation procedures and evaluation protocols. Key implementation settings including image preprocessing, patch parameters, feature levels, decomposition scales, training environments and ablation experiment setups are documented in detail. All compared methods adopt unified configurations and full-image evaluation criteria, and random seeds and configuration files for experiments are fully preserved. Upon acceptance, we will release the complete source code, configuration files, trained models, data-preparation and evaluation scripts to enable independent reproduction of our experimental results.

REFERENCES

Hengzhi Chen, Liqian Feng, Wenhua Wu, Xiaogang Zhu, Shawn Leo, and Kun Hu. F2net: A frequency-fused network for ultra-high resolution remote sensing segmentation. *CoRR*, abs/2506.07847, 2025.

Kai Chen, Zerun Wang, Xueyang Wang, Dahan Gong, Longlong Yu, Yuchen Guo, and Guiguang Ding. Towards real-time object detection in gigapixel-level video. *Neurocomputing*, 477:14–24, 2022.

Linwei Chen, Ying Fu, Lin Gu, Chenggang Yan, Tatsuya Harada, and Gao Huang. Frequency-aware feature fusion for dense image prediction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12): 10763–10780, 2024a.

Linwei Chen, Lin Gu, Dezhi Zheng, and Ying Fu. Frequency-adaptive dilated convolution for semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 3414–3425. IEEE, 2024b.

Wuyang Chen, Ziyu Jiang, Zhangyang Wang, Kexin Cui, and Xiaoning Qian. Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 8924–8933. Computer Vision Foundation / IEEE, 2019a.

Yunpeng Chen, Haoqi Fan, Bing Xu, Zhicheng Yan, Yannis Kalantidis, Marcus Rohrbach, Shuicheng Yan, and Jiashi Feng. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 3434–3443. IEEE, 2019b.

Gong Cheng, Xiang Yuan, Xiwen Yao, Kebing Yan, Qinghua Zeng, Xingxing Xie, and Junwei Han. Towards large-scale small object detection: Survey and benchmarks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(11):13467–13488, 2023.

Bowei Du, Yecheng Huang, Jiaxin Chen, and Di Huang. Adaptive sparse convolutional networks with global context enhancement for faster object detection on drone images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 13435–13444. IEEE, 2023.

Mingfei Gao, Ruichi Yu, Ang Li, Vlad I. Morariu, and Larry S. Davis. Dynamic zoom-in network for fast object detection in large images. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 6926–6935. Computer Vision Foundation / IEEE Computer Society, 2018.

Jingfang Li, Haoran Zhu, Wen Yang, Jinrui Zhang, Fang Xu, Haijian Zhang, and Gui-Song Xia. UHR-DETR: efficient end-to-end small object detection for ultra-high-resolution remote sensing imagery. *CoRR*, abs/2604.21435, 2026.

Qi Li, Weixiang Yang, Wenxi Liu, Yuanlong Yu, and Shengfeng He. From contexts to locality: Ultra-high resolution image segmentation via locality-aware contextual correlation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 7232–7241. IEEE, 2021.

Wenxi Li, Yuchen Guo, Jilai Zheng, Haozhe Lin, Chao Ma, Lu Fang, and Xiaokang Yang. Sparseformer: Detecting objects in HRW shots via sparse vision transformer. In Jianfei Cai, Mohan S. Kankanhalli, Balakrishnan Prabhakaran, Susanne Boll, Ramanathan Subramanian, Liang Zheng, Vivek K. Singh, Pablo César, Lexing Xie, and Dong Xu (eds.), *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, pp. 4851–4860. ACM, 2024.

Yansheng Li, Linlin Wang, Tingzhu Wang, Xue Yang, Junwei Luo, Qi Wang, Youming Deng, Wenbin Wang, Xian Sun, Haifeng Li, Bo Dang, Yongjun Zhang, Yi Yu, and Junchi Yan. STAR: A first-ever dataset and a large-scale benchmark for scene graph generation in large-size satellite imagery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(3):1832–1849, 2025.

Yong Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Structure inference net: Object detection using scene-level context and instance-level relationships. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 6985–6994. Computer Vision Foundation / IEEE Computer Society, 2018.

Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 763–772. IEEE, 2021.

Huixin Sun, Runqi Wang, Yanjing Li, Linlin Yang, Shaohui Lin, Xianbin Cao, and Baochang Zhang. SET: spectral enhancement for tiny object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 4713–4723. Computer Vision Foundation / IEEE, 2025.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge J. Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. DOTA: A large-scale dataset for object detection in aerial images. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 3974–3983. Computer Vision Foundation / IEEE Computer Society, 2018.

Chenhongyi Yang, Zehao Huang, and Naiyan Wang. Querydet: Cascaded sparse query for accelerating high-resolution small object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 13658–13667. IEEE, 2022.

Fan Yang, Heng Fan, Peng Chu, Erik Blasch, and Haibin Ling. Clustered object detection in aerial images. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 8310–8319. IEEE, 2019.

Zhibin Zhang, Wanli Xue, Kaihua Zhang, and Shengyong Chen. 'skimming-perusal' detection: A simple object detection baseline in gigapixel-level images. In *IEEE International Conference on Multimedia and Expo, ICME 2023, Brisbane, Australia, July 10-14, 2023*, pp. 2471–2476. IEEE, 2023.

Supplementary Material

This supplementary material is organized as follows:

- Appendix A describes the training environment and the implementation protocols used for the comparison methods.
- Appendix B provides dataset preparation, input configurations, and full-image evaluation details for SODA-A and STAR.
- Appendix C presents additional context-mapping experiments, feature-response visualizations, and qualitative detection comparisons.
- Appendix D provides the complete mathematical formulation, training objective, sampling-coordinate derivation, frequency interpretation, and interaction-operator analysis.
- Appendix E discusses the scope, limitations, and possible extensions of CrossFreq-UHR.
- Appendix F discusses potential applications and broader societal considerations.

A Training Environment and Baseline Settings

All experiments are conducted on a single NVIDIA L40 GPU. Our primary implementation uses PyTorch 2.6.0 with CUDA 12.4. Comparison methods are based on their official implementations and method-specific configurations, with the required software dependencies maintained in separate environments.

Baselines are organized into three UHR processing paradigms. Global-downsampling methods resize the complete image to 1024×1024 before detection. Sliding-window methods exhaustively process local crops and then restore and merge their predictions in the full-image coordinate system. Patch-based methods retain their respective region-selection mechanisms to allocate high-resolution local computation. Dataset interfaces and full-image evaluation procedures are adapted to the prepared SODA-A and STAR benchmarks while preserving each method's core architecture and processing strategy.

B Benchmark Details

We evaluate CrossFreq-UHR on SODA-A (Cheng et al., 2023) and STAR (Li et al., 2025) using horizontal bounding boxes. Preparation is performed separately within each data split. Tab. B summarizes the resulting datasets.

B.1 SODA-A

SODA-A contains nine categories: airplane, helicopter, small vehicle, large vehicle, ship, container, storage tank, swimming pool, and windmill. Images and annotations are processed as follows:

- Enlarge each image by $2\times$ along both spatial dimensions using bilinear interpolation.
- Place the enlarged image at the top-left of a 9600×9600 canvas and zero-pad the right and bottom.
- Scale annotation coordinates accordingly, convert oriented polygons to axis-aligned enclosing boxes, and remove ignored or invalid annotations.

The final 9600×9600 size is the prepared canvas size, not the native acquisition resolution.

B.2 STAR

STAR contains eight categories: car, boat, tank, truck, boarding bridge, crane, airplane, and lattice tower. Images are cropped and zero-padded to 8192×8192 without rescaling, according to the following rules:

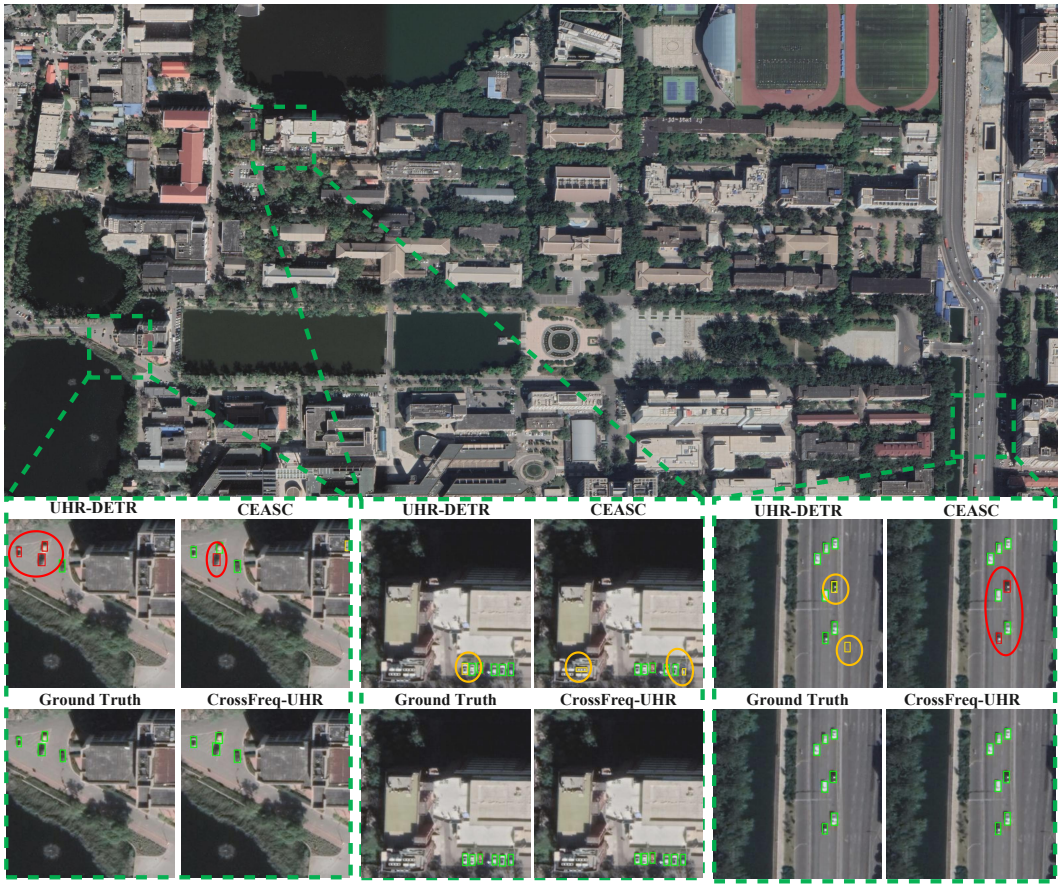

Figure A: Full-scene qualitative comparison on STAR. The upper panel shows three representative regions in a UHR image, while the lower panels compare UHR-DETR, CEASC, and CrossFreq-UHR with the ground truth. Colored ellipses highlight representative missed or inaccurate detections.

Table A: Patch-size sensitivity on SODA-A under an equal local-pixel budget. All variants follow the same six-epoch training and evaluation protocol.

Patch size	K	$\mathbf{AP}_{50}$	AP	$\mathbf{AP}_{75}$	$\mathbf{AP}_S$	$\mathbf{AP}_M$	$\mathbf{AP}_L$	AR
256×256	160	50.69	25.70	**23.30**	13.81	30.34	22.71	41.10
512×512	40	50.77	25.81	23.13	13.57	30.60	**22.93**	40.50
1024×1024	10	**52.72**	**26.39**	23.19	**13.93**	**31.31**	22.04	**41.50**

- **Either dimension** ≤ 6000**:** discard the source image.
- **Both dimensions in** $(6000, 8192]$**:** retain the full image only if its area is at least two-thirds of the target canvas; zero-pad the right and bottom.
- **Only one dimension** > 8192**:** crop along that dimension with a nominal stride of 8192 pixels, shift the final window inward to align with the boundary, and pad the shorter dimension when necessary.
- **Both dimensions** > 8192**:** apply the same cropping rule along both axes, without padding.

Annotations are mapped to crop-local coordinates, and crops without retained annotations are removed. Boundary-aligned windows may overlap, so annotation counts include repeated object occurrences across crops. All crops remain in their source image's data split.

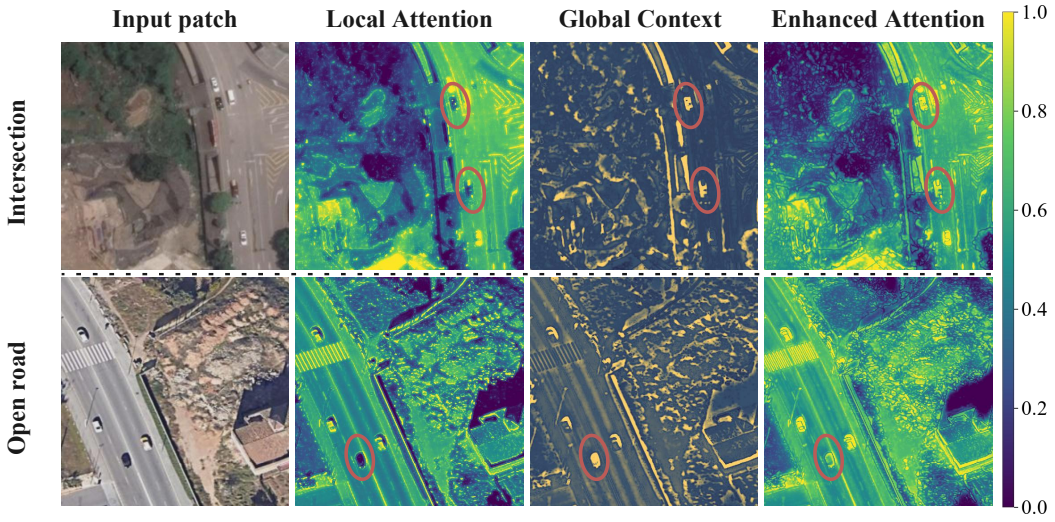

Figure B: Visualization of local-attention enhancement using spatially aligned global context. Columns show the input patch, local attention, global context, and enhanced attention. Red circles indicate vehicles whose initially weak responses become more pronounced after context interaction.

Table B: Dataset statistics after preparation.

Dataset	Split	Images	Boxes	Image size
SODA-A	Train	1,067	344,228	9600×9600
	Val	576	231,439	9600×9600
	Test	870	296,402	9600×9600
STAR	Train	839	114,616	8192×8192
	Val	362	36,805	8192×8192
	Test	263	26,550	8192×8192

B.3 Detector Inputs and Evaluation

CrossFreq-UHR uses the following input settings:

- **SODA-A:** 9600×9600 full image, 2400×2400 global observation, and $K = 80$ local patches.
- **STAR:** 8192×8192 full image, 2048×2048 global observation, and $K = 40$ local patches.

Each local patch is 512×512 and undergoes no additional patch-wise resizing. Predictions are evaluated against full-image annotations in the prepared image coordinates, rather than separately on local patches. STAR predictions are not merged back into the larger source images.

We report mean class-wise AP at IoU 0.5 and its small-, medium-, and large-object breakdowns. Object areas are measured in the prepared coordinates; SODA-A enlargement therefore multiplies box areas by four.

Table notation. For SODA-A, AI, HE, SV, LV, SH, CO, ST, SP, and WI denote airplane, helicopter, small vehicle, large vehicle, ship, container, storage tank, swimming pool, and windmill, respectively. For STAR, BB and LT denote boarding bridge and lattice tower; the remaining category names are written in full. In the main comparison tables, mAP denotes category-averaged AP at IoU 0.5, while AP_S, AP_M, and AP_L denote its small-, medium-, and large-object breakdowns. Lat. reports end-to-end inference latency per full image, and Mem. reports peak GPU memory during inference. Red, orange, and yellow cells indicate the best, second-best, and third-best results, respectively.

Table C: Peak training memory at different input resolutions on SODA-A. OOM indicates failure to complete a stable training iteration.

Method	512	1K	2K	3K	4K	5K	6K	7K	8K
FCOS	0.81	2.29	8.22	21.34	37.67	OOM	OOM	OOM	OOM
RT-DETR	3.14	4.10	8.45	17.54	41.57	OOM	OOM	OOM	OOM
UHR-DETR	17.69	16.73	16.35	17.42	17.94	17.77	17.87	17.90	18.02
CrossFreq-UHR	17.28	17.29	17.34	17.41	17.52	17.66	17.84	18.04	18.28

Table D: Controlled analysis of context mapping in SACM on SODA-A. All rows use the same trained CrossFreq-UHR checkpoint and differ only in the context supplied at inference. This setting isolates spatial correspondence without introducing variation from independent optimization.

Context Mapping	$\mathbf{AP}_{50}$	$\mathbf{AP}^{\mathrm{S}}_{50}$	$\mathbf{AP}^{\mathrm{M}}_{50}$	$\mathbf{AP}^{\mathrm{L}}_{50}$
Spatially averaged	54.58	30.63	61.37	52.44
Spatially shifted	54.43	30.18	61.29	50.42
Spatially aligned	**56.08**	**31.16**	**62.91**	**53.51**

C Additional Experiments

C.1 Analysis of Context Mapping

We further examine whether CrossFreq-UHR relies on the spatial correspondence between each selected patch and its global context. Tab. D compares different context mappings using the same trained checkpoint. Replacing spatially aligned context with spatial averaging or spatial shifting decreases AP_{50} from 56.08 to 54.58 and 54.43, respectively. The degradation is observed across all object-size groups and is particularly pronounced for large objects. These interventions indicate that the trained model uses the spatial correspondence established by SACM rather than only the presence of additional global features.

Tab. E provides a complementary comparison in which each mapping variant is trained independently from the same initialization. The three variants achieve similar overall performance, and the shifted variant obtains slightly higher AP_{50} and AP. The aligned variant nevertheless performs best on medium and large objects, with a clearer advantage on the latter. Compared with the same-checkpoint interventions, these results suggest that end-to-end optimization can partially adapt to a fixed context transformation. Therefore, the controlled intervention demonstrates the model's reliance on its learned spatial correspondence, whereas the independently trained comparison does not imply that alignment is universally superior under every metric.

C.2 Full-Scene Qualitative Analysis

Fig. A presents a full-scene comparison on STAR, with enlarged views of representative regions containing small and sparsely distributed objects. UHR-DETR and CEASC exhibit missed or inaccurate detections in visually complex areas, particularly near cluttered buildings, dark backgrounds, and narrow roads. CrossFreq-UHR recovers more of the annotated objects and produces predictions that more closely match the ground truth. These examples indicate that spatially aligned scene context improves the interpretation of ambiguous local regions, while the detail path retains the fine visual evidence required for small-object localization.

C.3 Visualization of Context-Guided Response Enhancement

Fig. B visualizes how spatially aligned global context affects the response of selected local regions in two representative scenes. The local-response maps contain strong background activations, while several vehicles highlighted by the red circles receive relatively weak responses. The corresponding global-context maps preserve the spatial locations of these vehicles and provide complementary scene-level evidence. After context–local interaction, the enhanced maps produce more pronounced and spatially concentrated responses around the highlighted targets. This behavior is consistent

Table E: Additional results for independently trained SACM mapping variants on SODA-A. Unlike Tab. D, each row is optimized from the same initialization under its own mapping rule. The small differences indicate that end-to-end training can partially compensate for a fixed context transform.

Context Mapping	$\mathbf{AP}_{50}$	$\mathbf{AP}^{\mathrm{S}}_{50}$	$\mathbf{AP}^{\mathrm{M}}_{50}$	$\mathbf{AP}^{\mathrm{L}}_{50}$	AP	$\mathbf{AP}_{75}$	AR
Spatially averaged	56.00	31.85	62.49	49.37	30.29	28.26	43.59
Spatially shifted	**56.25**	**33.07**	62.87	48.53	**30.58**	**28.41**	**43.92**
Spatially aligned	56.08	31.16	**62.91**	**53.51**	29.47	26.68	43.22

across the intersection and open-road examples, despite their different backgrounds and object layouts. The visualization therefore supports the role of spatial alignment in transferring relevant scene context to each patch and shows that CrossFreq-UHR strengthens weak target evidence before multi-scale feature aggregation.

C.4 Feature-Response Visualization

Fig. C visualizes target-centered feature responses for CrossFreq-UHR and its controlled variants. In the illustrated examples, the complete model generally produces more concentrated responses around the annotated targets, while removing frequency decomposition, global conditioning, or asymmetric interaction results in weaker or more spatially diffuse activations. The comparison provides qualitative evidence that aligned context and local-detail processing affect where the model allocates its feature responses. Because these maps visualize intermediate activations rather than final predictions, they are used as supporting evidence rather than a direct measure of detection accuracy.

C.5 Qualitative Detection Analysis

Fig. D compares the final detections produced by CrossFreq-UHR and its ablated variants on representative SODA-A scenes. The examples include isolated small objects, densely distributed targets, low-contrast regions, and structures that can be confused with the surrounding background. Removing individual components leads to additional missed detections, false positives, or less reliable localization in the highlighted regions. The complete model recovers more annotated targets in these examples while suppressing several background responses. These results complement the aggregate ablations by showing how the proposed components affect individual predictions in challenging scenes.

C.6 Training-Memory Scaling with Input Resolution

Tab. C compares training-memory growth as the input resolution increases. FCOS and RT-DETR exhibit rapid memory growth because they construct dense features over the complete image. Their peak memory reaches 37.67 and 41.57 GiB at 4K, respectively, and both fail to complete a stable training iteration at 5K. By contrast, UHR-DETR and CrossFreq-UHR remain close to 18 GiB throughout the tested resolution range because their dominant local computation is determined by a fixed sparse-patch budget. CrossFreq-UHR increases from 17.28 GiB at 512 to 18.28 GiB at 8K, with only 0.26 GiB more memory than UHR-DETR at 8K. The proposed cross-frequency interaction therefore preserves the favorable memory scaling of sparse global–local detection.

C.7 Patch-Size Sensitivity under a Matched Pixel Budget

To isolate patch granularity from the amount of high-resolution computation, we maintain an equal local-pixel budget: $160 \times 256^2 = 40 \times 512^2 = 10 \times 1024^2$. As shown in Tab. F, using 1024×1024 patches achieves the highest AP and AP_{50}, exceeding the 512×512 setting by 0.58 and 1.95 points, respectively. It also performs best on AP_S, AP_M, and AR, indicating that the larger spatial extent provides useful local context under the same processed-pixel budget. The three settings obtain comparable AP_{75}, while 512×512 gives the highest AP_L. These results show that patch granularity controls the trade-off between the contextual extent of each patch and the number of spatial regions selected by the router.

Table F: Patch-size sensitivity on SODA-A under an equal local-pixel budget. All variants follow the same six-epoch training and evaluation protocol.

Patch size	K	$\mathbf{AP}_{50}$	**AP**	$\mathbf{AP}_{75}$	$\mathbf{AP}_S$	$\mathbf{AP}_M$	$\mathbf{AP}_L$	**AR**
256×256	160	50.69	25.70	**23.30**	13.81	30.34	22.71	41.10
512×512	40	50.77	25.81	23.13	13.57	30.60	**22.93**	40.50
1024×1024	10	**52.72**	**26.39**	23.19	**13.93**	**31.31**	22.04	**41.50**

D Mathematical Formulation and Analysis of CrossFreq-UHR

D.1 End-to-End Formulation and Algorithm

This subsection formalizes the data flow in Fig. 2, using the notation introduced in Secs. 3 and 4. Let $\mathbf{I} \in \mathbb{R}^{3\times H_I \times W_I}$ be the preprocessed detector input, where H_I and W_I are its height and width. Let $\Lambda = \{3, 4, 5\}$ be the set of pyramid levels and K the number of selected patches. The global branch computes

$$\mathbf{I}_g = \mathrm{Down}_4(\mathbf{I}), \qquad \{\mathbf{G}^\ell\}_{\ell\in\Lambda} = E_g(\mathbf{I}_g), \tag{12}$$

where Down_4 denotes four-times spatial downsampling and E_g is the global encoder, including its channel projections. At level ℓ, $\mathbf{G}^\ell \in \mathbb{R}^{d\times H_g^\ell \times W_g^\ell}$, where $d = 256$ and H_g^ℓ, W_g^ℓ are the global feature dimensions.

The inherited gain-map head $\mathcal{H}_{\mathrm{gain}}$ produces the routing map $\mathbf{R}_{\mathrm{gain}}$. The inherited router $\mathcal{Q}_{\mathrm{route}}$ then selects an indexed collection of patch windows:

$$\mathbf{R}_{\mathrm{gain}} = \mathcal{H}_{\mathrm{gain}}(\mathbf{G}^5), \qquad \mathcal{W}_K = \mathcal{Q}_{\mathrm{route}}(\mathbf{R}_{\mathrm{gain}}, K) = \{\mathbf{b}_k\}_{k=1}^K. \tag{13}$$

Here $\mathbf{b}_k = (x_k, y_k, W_p, H_p)$ records the top-left coordinates and fixed width and height of the k-th patch in the coordinate frame of $\mathbf{I}$. The patch dimensions are $H_p = W_p = 512$ in the reported configuration.

The local branch extracts

$$\mathbf{I}_k = \mathrm{CropPad}(\mathbf{I}, \mathbf{b}_k), \qquad \{\mathbf{F}_k^\ell\}_{\ell\in\Lambda} = E_{\mathrm{loc}}(\mathbf{I}_k), \tag{14}$$

where $\mathrm{CropPad}$ crops the selected window and zero-pads portions outside the input image. Thus $\mathbf{I}_k \in \mathbb{R}^{3\times H_p\times W_p}$. The local encoder E_{loc}, including its channel projections, produces $\mathbf{F}_k^\ell \in \mathbb{R}^{d\times h_\ell \times w_\ell}$. The local strides are $s_3 = 8$, $s_4 = 16$, and $s_5 = 32$, and h_ℓ, w_ℓ denote the resulting feature dimensions.

CrossFreq-UHR modifies levels 3 and 4:

$$\mathbf{C}_k^\ell = \mathcal{M}_\ell(\mathbf{G}^\ell, \mathbf{b}_k), \tag{15}$$

$$\left(\mathbf{L}_k^\ell, \overline{\mathbf{L}}_k^\ell, \mathbf{H}_k^\ell\right) = \mathcal{E}_{\ell,r}(\mathbf{F}_k^\ell), \tag{16}$$

$$\begin{aligned}\widetilde{\mathbf{F}}_k^\ell &= \mathcal{X}_\ell(\mathbf{F}_k^\ell, \mathbf{C}_k^\ell) \\ &= \Phi_{\mathrm{sem}}^\ell(\mathbf{L}_k^\ell, \mathbf{C}_k^\ell) + \Phi_{\mathrm{detail}}^\ell(\mathbf{H}_k^\ell), \qquad \ell \in \{3, 4\}.\end{aligned} \tag{17}$$

The operator $\mathcal{M}_\ell$ is SACM. The operator $\mathcal{E}_{\ell,r}$ denotes the EFD computation defined in Sec. 4.3, with reconstruction scale r. Its outputs are the smoothed structural feature $\mathbf{L}_k^\ell$, coarse reconstruction $\overline{\mathbf{L}}_k^\ell$, and reconstruction residual $\mathbf{H}_k^\ell$. The operators Φ_{sem}^ℓ and $\Phi_{\mathrm{detail}}^\ell$ are the structural and detail paths of ASDI, respectively. Their composition with EFD defines $\mathcal{X}_\ell$. The aligned context and enhanced local feature both have shape $d \times h_\ell \times w_\ell$. The coarsest level is unchanged: $\widetilde{\mathbf{F}}_k^5 = \mathbf{F}_k^5$.

Let E_{ms} denote the inherited hybrid encoder and D_{gl} the inherited global–local decoding stage. The final computation is

$$\begin{aligned}\mathbf{Z}_{\mathrm{enc}} &= E_{\mathrm{ms}}\left(\{\widetilde{\mathbf{F}}_k^\ell\}_{k=1,\ldots,K;\,\ell\in\Lambda}\right), \\ \widehat{\mathcal{Y}} &= D_{\mathrm{gl}}\left(\mathbf{Z}_{\mathrm{enc}}, \mathbf{G}^5, \mathcal{W}_K\right).\end{aligned} \tag{18}$$

Here $\mathbf{Z}_{\mathrm{enc}}$ denotes the encoded local multi-scale representation, and $\widehat{\mathcal{Y}}$ is the collection of predicted boxes, class labels, and confidence scores in the coordinate frame of $\mathbf{I}$. For compactness,

Algorithm 1 Forward pass of CrossFreq-UHR

Require: Preprocessed image $\mathbf{I}$ and its valid-support metadata; patch budget K; reconstruction scale r

Ensure: Detections $\widehat{\mathcal{Y}}$ in the coordinate frame of $\mathbf{I}$

$\mathbf{I}_g \leftarrow \mathrm{Down}_4(\mathbf{I})$
$\{\mathbf{G}^\ell\}_{\ell\in\Lambda} \leftarrow E_g(\mathbf{I}_g)$
$\mathbf{R}_{\mathrm{gain}} \leftarrow \mathcal{H}_{\mathrm{gain}}(\mathbf{G}^5)$
$\mathcal{W}_K = \{\mathbf{b}_k\}_{k=1}^K \leftarrow \mathcal{Q}_{\mathrm{route}}(\mathbf{R}_{\mathrm{gain}}, K)$
for $k = 1$ to K **do**
  $\mathbf{I}_k \leftarrow \mathrm{CropPad}(\mathbf{I}, \mathbf{b}_k)$
  $\{\mathbf{F}_k^\ell\}_{\ell\in\Lambda} \leftarrow E_{\mathrm{loc}}(\mathbf{I}_k)$
  for $\ell \in \{3, 4\}$ **do**
    $\mathbf{C}_k^\ell \leftarrow \mathcal{M}_\ell(\mathbf{G}^\ell, \mathbf{b}_k)$
    $\mathbf{L}_k^\ell \leftarrow \mathcal{S}(\mathbf{F}_k^\ell)$
    $\overline{\mathbf{L}}_k^\ell \leftarrow \mathcal{U}_r(\mathcal{A}_r(\mathbf{L}_k^\ell))$
    $\mathbf{H}_k^\ell \leftarrow \mathbf{F}_k^\ell - \overline{\mathbf{L}}_k^\ell$
    $\mathbf{S}_k^\ell \leftarrow \Phi_{\mathrm{sem}}^\ell(\mathbf{L}_k^\ell, \mathbf{C}_k^\ell)$
    $\mathbf{D}_k^\ell \leftarrow \Phi_{\mathrm{detail}}^\ell(\mathbf{H}_k^\ell)$
    $\widetilde{\mathbf{F}}_k^\ell \leftarrow \mathbf{S}_k^\ell + \mathbf{D}_k^\ell$
  end for
  $\widetilde{\mathbf{F}}_k^5 \leftarrow \mathbf{F}_k^5$
end for
$\mathbf{Z}_{\mathrm{enc}} \leftarrow E_{\mathrm{ms}}(\{\widetilde{\mathbf{F}}_k^\ell\}_{k=1,\dots,K;\,\ell\in\Lambda})$
$\widehat{\mathcal{Y}} \leftarrow D_{\mathrm{gl}}(\mathbf{Z}_{\mathrm{enc}}, \mathbf{G}^5, \mathcal{W}_K)$
return $\widehat{\mathcal{Y}}$

D_{gl} includes the inherited prediction heads and coordinate conversion associated with decoding. Its dependence on $\mathcal{W}_K$ records the relationship between patch-local and full-image coordinates. When image-level preprocessing changes the source-image coordinates, the inherited inverse transformation is applied afterward.

Feature dimensions, strides, and valid image support are treated as known per-image metadata and are omitted from the operator arguments when unambiguous. Algorithm 1 summarizes the forward pass. The patch loops express the computation; implementations may process the selected patches as a batch.

D.2 Training Objective

CrossFreq-UHR retains the inherited detection and routing objectives. Let $\mathcal{T}$ be the index set of supervised prediction groups, including final decoder predictions, intermediate decoder predictions, encoder predictions, and denoising predictions. For each $\nu \in \mathcal{T}$, the superscript (ν) identifies the corresponding loss terms. Regular predictions use the inherited Hungarian matching, whereas denoising predictions use their known target assignments.

The detection objective is

$$\mathcal{L}_{\mathrm{det}} = \sum_{\nu\in\mathcal{T}} \left(\mathcal{L}_{\mathrm{VFL}}^{(\nu)} + 5\mathcal{L}_1^{(\nu)} + 2\mathcal{L}_{\mathrm{GIoU}}^{(\nu)} \right), \tag{19}$$

where $\mathcal{L}_{\mathrm{VFL}}^{(\nu)}$, $\mathcal{L}_1^{(\nu)}$, and $\mathcal{L}_{\mathrm{GIoU}}^{(\nu)}$ are the Varifocal classification, L_1 box-regression, and generalized-IoU losses for prediction group ν, respectively. These terms use the inherited target construction and normalization; the coefficients displayed outside the terms are their loss weights.

For the routing objective, consider one image and let $\mathcal{P}$ be the nonempty set of two-dimensional gain-map indices. Each $p \in \mathcal{P}$ corresponds to a candidate patch window Π_p in the coordinate frame of $\mathbf{I}$. Let $\{\mathbf{B}_i^{\mathrm{gt}}\}_{i=1}^N$ be the ground-truth boxes, where N is the number of objects and each box has positive area. The candidate window Π_p and the ground-truth boxes use the same coordinate system.

Define the raw coverage score and bounded target as

$$\rho_p = \sum_{i=1}^{N} \frac{\mathrm{area}(\Pi_p \cap \mathbf{B}_i^{\mathrm{gt}})}{\mathrm{area}(\mathbf{B}_i^{\mathrm{gt}})}, \qquad y_p = \min\left(\sqrt{\rho_p}, J_{\max} - \varepsilon\right), \qquad J_{\max} = 6. \tag{20}$$

Here $\mathrm{area}(\cdot)$ denotes spatial area, and $0 < \varepsilon < 1$ is the fixed numerical safeguard used to keep the target below the highest bin. The score ρ_p is a sum of per-object coverage fractions, rather than their mean, and may exceed one. When $N = 0$, the empty sum gives $\rho_p = 0$.

Let $z_{p,j}$ be the predicted logit for bin $j \in \{0, \ldots, J_{\max}\}$ at location p. The bin probabilities and continuous gain prediction are

$$\pi_{p,j} = \frac{\exp(z_{p,j})}{\sum_{j'=0}^{J_{\max}} \exp(z_{p,j'})}, \qquad g_p = \sum_{j=0}^{J_{\max}} j\, \pi_{p,j}. \tag{21}$$

The routing map in Eq. 13 is $\mathbf{R}_{\mathrm{gain}} = (g_p)_{p \in \mathcal{P}}$; the gain-map head notation includes the conversion from bin logits to continuous gains.

Let $a_p = \lfloor y_p \rfloor$ and $c_p = a_p + 1$ denote the lower and upper interpolation bins, respectively. The target bound ensures that both indices lie in $\{0, \ldots, J_{\max}\}$. The gain-map objective averages the distribution focal loss over gain-map locations:

$$\mathcal{L}_{\mathrm{map}} = -\frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \left[(c_p - y_p) \log \pi_{p,a_p} + (y_p - a_p) \log \pi_{p,c_p} \right], \tag{22}$$

where $|\mathcal{P}|$ denotes the number of gain-map locations.

The local peak margin objective compares eligible neighboring gain-map locations. Let

$$\Delta_4 = \{(1,0), (-1,0), (0,1), (0,-1)\} \tag{23}$$

be the set of four unit displacements on the gain-map grid. For each $\delta \in \Delta_4$, define

$$\begin{aligned} \Omega_\delta &= \{p \in \mathcal{P} : p + \delta \in \mathcal{P},\ y_p \geq 1,\ y_p > y_{p+\delta}\}, \\ \Delta_4^+ &= \{\delta \in \Delta_4 : |\Omega_\delta| > 0\}. \end{aligned} \tag{24}$$

Thus Ω_δ contains locations whose targets exceed those of their in-grid neighbors in direction δ, and Δ_4^+ contains directions with at least one such pair. With margin $m = 0.01$, the weighted objective is

$$\mathcal{L}_{\mathrm{LPM}} = \begin{cases} \dfrac{10}{|\Delta_4^+|} \displaystyle\sum_{\delta \in \Delta_4^+} \frac{1}{|\Omega_\delta|} \sum_{p \in \Omega_\delta} \max(0, g_{p+\delta} + m - g_p), & \Delta_4^+ \neq \varnothing, \\ 0, & \Delta_4^+ = \varnothing. \end{cases} \tag{25}$$

The notation $|\cdot|$ in Eqs. 22–25 denotes set cardinality. The coefficient 10 is included in $\mathcal{L}_{\mathrm{LPM}}$ and is not applied again.

The total objective is

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{det}} + \mathcal{L}_{\mathrm{map}} + \mathcal{L}_{\mathrm{LPM}}. \tag{26}$$

The routing expressions above are written for a single image; batch reduction follows the inherited implementation. The discrete window-selection operation is not assumed to be differentiable.

SACM and EFD use fixed operators and introduce no additional learnable parameters. Gradients propagate through their feature-processing operations to the encoders, while the ASDI parameters are optimized jointly with the inherited detector. No additional loss is attached to the CrossFreq-UHR block.

D.3 Sampling Coordinates and Validity Masks

This subsection specifies the sampling grid and masks in Eq. 7. All image-space coordinates refer to the preprocessed input $\mathbf{I}$. The selected patches retain their native patch resolution without an additional patch-wise resize. The window $\mathbf{b}_k = (x_k, y_k, W_p, H_p)$ is equivalent to the corner representation $(x_k, y_k, x_k + W_p, y_k + H_p)$.

For local level ℓ, the horizontal and vertical feature indices satisfy

$$u \in \{0, \dots, w_\ell - 1\}, \qquad v \in \{0, \dots, h_\ell - 1\}. \tag{27}$$

The implemented local lattice has offset $a_{\mathrm{loc}} = 0.5$ and stride s_ℓ. The corresponding global lattice has offset $a_g = 2$ and stride $4s_\ell$ in the coordinate frame of $\mathbf{I}$. These offsets specify the sampling convention used by the implementation; they are not inferred from stride alone.

The image-space location associated with local index (u, v) is

$$\mathbf{p}_k^\ell(u, v) = (x_k + s_\ell u + a_{\mathrm{loc}}, y_k + s_\ell v + a_{\mathrm{loc}}) . \tag{28}$$

The corresponding continuous horizontal and vertical indices on $\mathbf{G}^\ell$ are

$$\xi_k^\ell(u) = \frac{x_k + s_\ell u + a_{\mathrm{loc}} - a_g}{4s_\ell}, \qquad \eta_k^\ell(v) = \frac{y_k + s_\ell v + a_{\mathrm{loc}} - a_g}{4s_\ell}. \tag{29}$$

For $\mathbf{G}^\ell \in \mathbb{R}^{d \times H_g^\ell \times W_g^\ell}$, the normalized grid entry is

$$\Gamma_\ell(\mathbf{b}_k)[v, u] = \left(2\frac{\xi_k^\ell(u) + 0.5}{W_g^\ell} - 1, 2\frac{\eta_k^\ell(v) + 0.5}{H_g^\ell} - 1 \right). \tag{30}$$

The two components are the normalized horizontal and vertical sampling coordinates. Bilinear sampling uses zero padding and `align_corners=False`. Under this convention, inverse normalization recovers $\xi_k^\ell(u)$ and $\eta_k^\ell(v)$ exactly. Coordinates outside the feature-map support are handled by the sampler's zero-padding rule.

Let Ω_I denote the valid, unpadded rectangular support of $\mathbf{I}$. The indicator $\mathbf{1}_{\Omega_I}(\mathbf{x})$ equals one when the image-space point $\mathbf{x}$ lies in Ω_I and zero otherwise. For global feature indices $i_g \in \{0, \dots, W_g^\ell - 1\}$ and $j_g \in \{0, \dots, H_g^\ell - 1\}$, the validity masks are

$$\begin{aligned} \mathbf{V}_g^\ell[j_g, i_g] &= \mathbf{1}_{\Omega_I}\left(a_g + 4s_\ell i_g,\, a_g + 4s_\ell j_g\right), \\ \mathbf{V}_k^\ell[v, u] &= \mathbf{1}_{\Omega_I}\left(\mathbf{p}_k^\ell(u, v)\right). \end{aligned} \tag{31}$$

Their shapes are $H_g^\ell \times W_g^\ell$ and $h_\ell \times w_\ell$, respectively. Both masks are broadcast across channels when used in Eq. 7.

The global mask is applied after Gaussian smoothing and before interpolation; the local mask is applied after interpolation. These operations zero invalid feature locations. They do not renormalize interpolation weights near boundaries or remove all boundary effects introduced by smoothing.

D.4 Residual Reconstruction and Frequency Interpretation

The smoothing operator $\mathcal{S}$ used by SACM and EFD applies the same fixed Gaussian kernel independently to every channel. Define the one-dimensional kernel vector $\boldsymbol{\kappa} \in \mathbb{R}^5$ by

$$\kappa[\tau] = \frac{\exp(-\tau^2/2)}{\sum_{\tau'=-2}^{2} \exp(-\tau'^2/2)}, \qquad \tau \in \{-2, -1, 0, 1, 2\}. \tag{32}$$

The two-dimensional kernel is $\boldsymbol{\kappa}\boldsymbol{\kappa}^\top$. Smoothing uses two-pixel zero padding.

The reconstruction scale r is a positive integer, with default value $r = 2$. The operator $\mathcal{A}_r$ applies average pooling with kernel size and stride r. The operator $\mathcal{U}_r$ reconstructs the pooled feature at the original local dimensions (h_ℓ, w_ℓ) using bilinear interpolation with `align_corners=False`. For fixed feature dimensions and boundary conventions, define

$$\mathcal{T}_r = \mathcal{U}_r \mathcal{A}_r \mathcal{S}. \tag{33}$$

This linear operator maps a local feature tensor to its coarse reconstruction. Its dependence on the fixed feature level is suppressed for readability.

Proposition 1 (Residual reconstruction). For the EFD computation in Eqs. 8 and 9,

$$\begin{aligned} \mathbf{F}_k^\ell &= \overline{\mathbf{L}}_k^\ell + \mathbf{H}_k^\ell, \\ \mathbf{L}_k^\ell + \mathbf{H}_k^\ell - \mathbf{F}_k^\ell &= \mathbf{L}_k^\ell - \overline{\mathbf{L}}_k^\ell. \end{aligned} \tag{34}$$

Proof. Substituting $\mathbf{H}_k^\ell = \mathbf{F}_k^\ell - \overline{\mathbf{L}}_k^\ell$ gives both identities directly. Thus the structural and residual inputs sum to the original feature if and only if $\mathbf{L}_k^\ell = \overline{\mathbf{L}}_k^\ell$. The model does not require this equality. □

Let Id denote the identity operator on the local feature space. The residual is

$$\mathbf{H}_k^\ell = (\mathrm{Id} - \mathcal{T}_r)\mathbf{F}_k^\ell. \tag{35}$$

For an additive input perturbation $\Delta\mathbf{F}$, define the induced residual change as

$$\Delta\mathbf{H} = (\mathrm{Id} - \mathcal{T}_r)(\mathbf{F}_k^\ell + \Delta\mathbf{F}) - (\mathrm{Id} - \mathcal{T}_r)\mathbf{F}_k^\ell. \tag{36}$$

Linearity then gives

$$\Delta\mathbf{H} = (\mathrm{Id} - \mathcal{T}_r)\Delta\mathbf{F}. \tag{37}$$

These relations characterize reconstruction residuals without assuming an ideal separation into frequency bands.

Gaussian smoothing, averaging, and interpolation tend to suppress rapid spatial variation, motivating the detail-oriented interpretation of the residual. However, strided pooling and reconstruction form a multirate operation that is generally not invariant to one-pixel shifts. Finite-image padding introduces additional boundary effects. Linearity alone therefore does not imply a scalar frequency response.

To state the exact frequency-domain relation, consider one channel at a fixed feature level. Let $\mathbf{f} \in \mathbb{R}^{h_\ell w_\ell}$ and $\mathbf{h}_{\mathrm{res}} \in \mathbb{R}^{h_\ell w_\ell}$ be the vectorized local feature and residual, respectively. Let $\mathbf{T}_r$ be the matrix representation of $\mathcal{T}_r$ on this spatial grid, and let $\mathbf{W}_{\mathrm{DFT}}$ be the invertible two-dimensional discrete Fourier transform matrix under the same vectorization. Define

$$\widehat{\mathbf{f}} = \mathbf{W}_{\mathrm{DFT}}\mathbf{f}, \qquad \widehat{\mathbf{h}}_{\mathrm{res}} = \mathbf{W}_{\mathrm{DFT}}\mathbf{h}_{\mathrm{res}}. \tag{38}$$

Then

$$\widehat{\mathbf{h}}_{\mathrm{res}} = \left(\mathbf{I}_{h_\ell w_\ell} - \mathbf{W}_{\mathrm{DFT}}\mathbf{T}_r\mathbf{W}_{\mathrm{DFT}}^{-1}\right)\widehat{\mathbf{f}}, \tag{39}$$

where $\mathbf{I}_{h_\ell w_\ell}$ is the identity matrix. The same relation holds independently for each channel. The transformed reconstruction matrix need not be diagonal.

The decomposition therefore does not imply orthogonality, disjoint frequency support, or guaranteed separation of object boundaries from background texture. The reconstruction identities are algebraic statements; their usefulness for detection is evaluated experimentally.

D.5 Interaction Operators and Conditional Context Dependence

This subsection expands the operators in Eq. 10. Fix one patch k and one level $\ell \in \{3, 4\}$. To simplify notation, write $\mathbf{F} = \mathbf{F}_k^\ell$, $\mathbf{C} = \mathbf{C}_k^\ell$, $\mathbf{L} = \mathbf{L}_k^\ell$, and $\mathbf{H} = \mathbf{H}_k^\ell$. The same convention applies to the outputs $\mathbf{S}$ and $\mathbf{D}$ and to the operators $\mathcal{X}$, Φ_{sem}, and Φ_{detail}. Write $h = h_\ell$ and $w = w_\ell$. Each feature level has its own learnable parameters.

In the following equations, $*$ denotes convolution, $[\cdot, \cdot]$ denotes channel concatenation, and $\odot$ denotes element-wise multiplication with broadcasting where needed. All convolutions preserve spatial dimensions. The functions $\mathrm{SiLU}(x) = x\sigma(x)$ and $\mathrm{ReLU}(x) = \max(0, x)$ act element-wise, where $\sigma(x) = (1 + \exp(-x))^{-1}$ is the sigmoid function. Group normalization, layer normalization, and batch normalization are abbreviated as GN, LN, and BN, respectively.

Structural operator. The concatenated input $[\mathbf{L}, \mathbf{C}]$ has shape $2d \times h \times w$. Its projected feature is

$$\mathbf{Z}_{\mathrm{str}} = \mathrm{SiLU}\left(\mathrm{GN}_{32}\left(W_f * [\mathbf{L}, \mathbf{C}]\right)\right), \tag{40}$$

where W_f is a learnable 1×1 convolution from $2d$ to d channels, and GN_{32} denotes group normalization with 32 groups.

Let $\mathrm{LN}_{\mathrm{grp}}$ and $\mathrm{LN}_{\mathrm{aux}}$ be separately parameterized layer normalizations across channels at each spatial location. Compute

$$\mathbf{A}_{\mathrm{grp}} = W_a * \mathrm{LN}_{\mathrm{grp}}(\mathbf{Z}_{\mathrm{str}}), \tag{41}$$

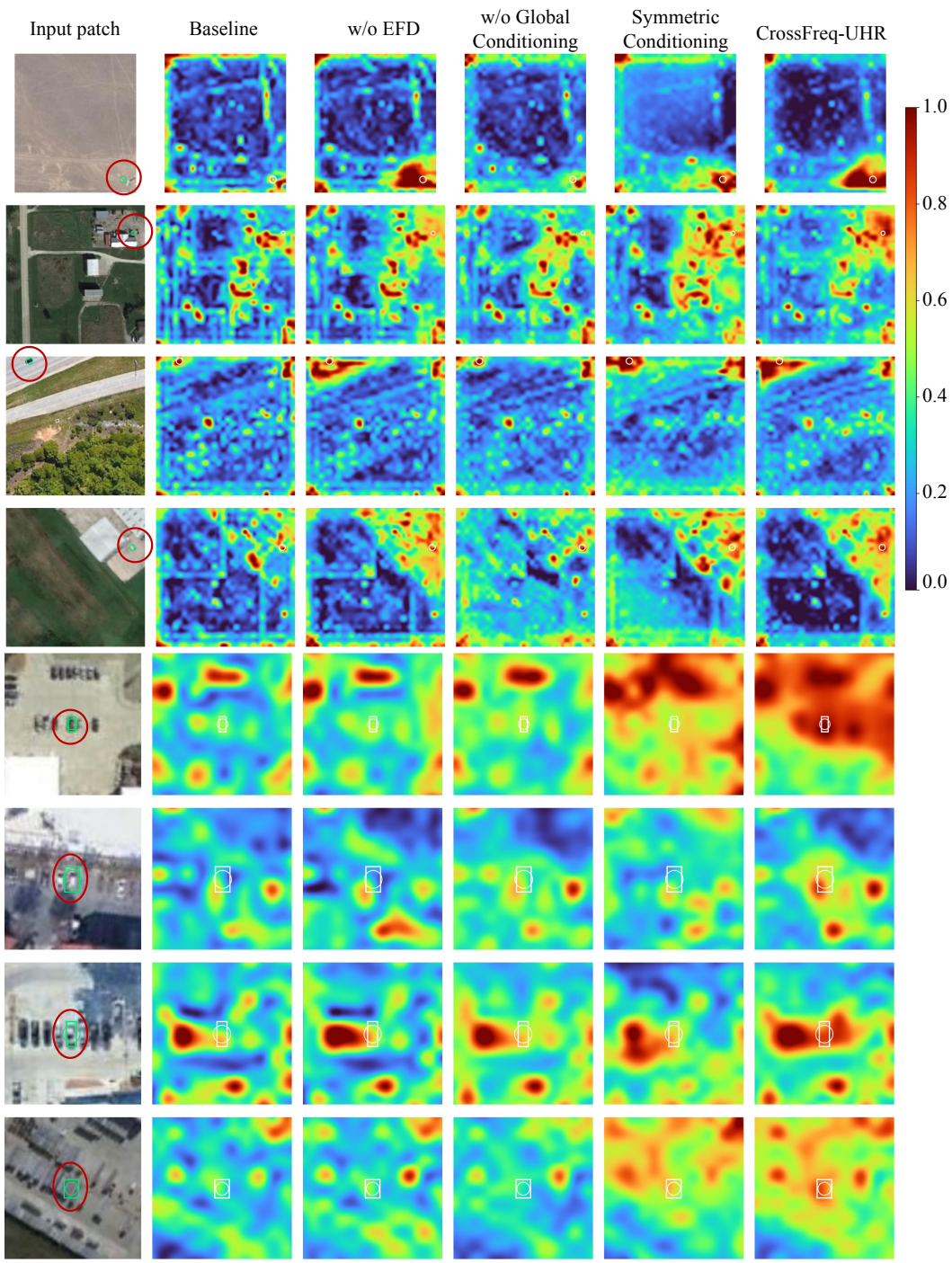

Figure C: Target-centered feature responses on SODA-A. Compared with the controlled ablations, CrossFreq-UHR produces more concentrated responses around annotated targets in the illustrated examples.

where W_a is a learnable 1×1 convolution from d to d channels. Split $\mathbf{A}_{\text{grp}}$ into eight channel groups $\mathbf{A}_{\text{grp},j}$, each with $d/8$ channels. The two internal structural features are

$$\mathbf{T}_{\text{grp}} = \text{Shuffle}_8\left(\text{Concat}_{j=1}^{8}\left[\text{PW}_j\left(\text{DW}_{3,j}(\mathbf{A}_{\text{grp},j})\right)\right]\right), \tag{42}$$

$$\mathbf{T}_{\text{aux}} = \text{PW}_{\text{aux}}\left(\text{DW}_{3,\text{aux}}\left(\text{LN}_{\text{aux}}(\mathbf{Z}_{\text{str}})\right)\right). \tag{43}$$

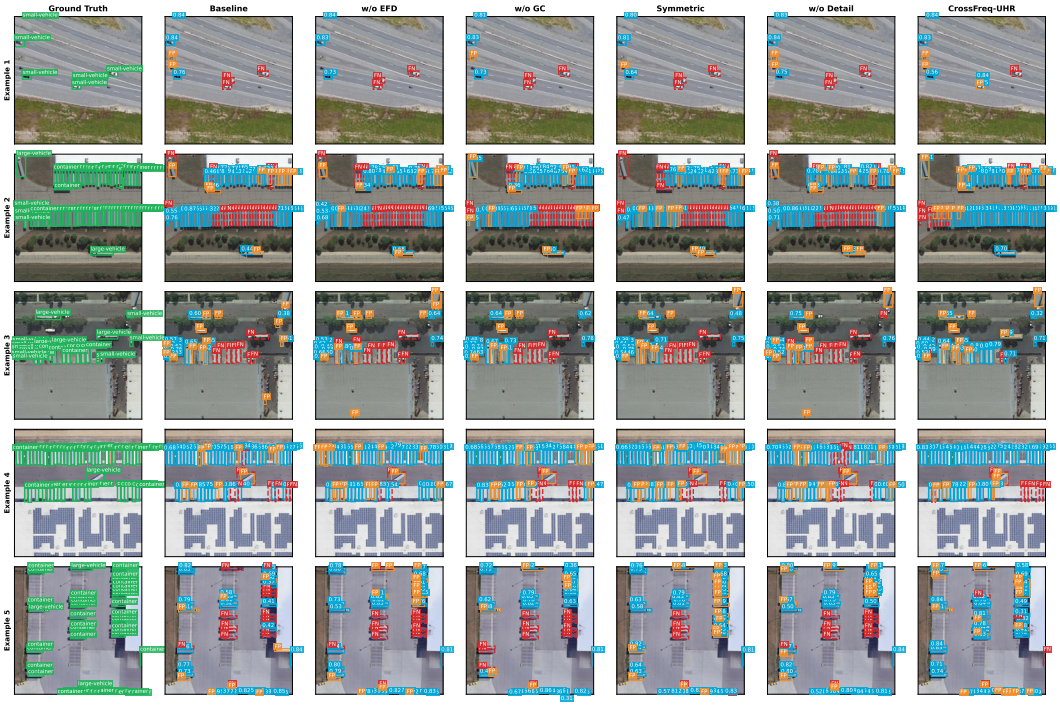

Figure D: Qualitative comparison with ablated variants on SODA-A. Columns show ground truth, the baseline, variants without EFD or global conditioning, symmetric conditioning, the variant without detail refinement, and CrossFreq-UHR. Green, cyan, red dashed, and orange boxes denote ground truth, true positives, false negatives, and false positives, respectively. Values above true-positive boxes indicate confidence scores.

Here $\mathrm{DW}_{3,j}$ is a depth-wise 3×3 convolution on channel group j, and PW_j is a point-wise convolution preserving that group's channel count. The operators with subscript aux act on all d channels and preserve that channel count. Concat concatenates along channels, and $\mathrm{Shuffle}_8$ denotes channel shuffle with eight groups.

The structural output is

$$\mathbf{S} = \Phi_{\mathrm{sem}}(\mathbf{L}, \mathbf{C}) = \mathbf{T}_{\mathrm{grp}} \odot \mathbf{T}_{\mathrm{aux}} \odot \mathrm{SiLU}(\mathbf{T}_{\mathrm{aux}}). \tag{44}$$

Both internal features and $\mathbf{S}$ have shape $d \times h \times w$. The auxiliary feature is internal to the structural operator; it is not the separate ASDI detail path. The multiplicative interaction is not a normalized attention weight and is not constrained to $[0, 1]$.

Detail operator. Local refinement first computes

$$\mathbf{Q}_{\mathrm{loc}} = \mathbf{H} + \phi_{\mathrm{loc}}(\mathbf{H}), \qquad \mathbf{M}_{\mathrm{abs}} = |\mathbf{Q}_{\mathrm{loc}}|, \tag{45}$$

where $|\cdot|$ denotes element-wise absolute value. The operator ϕ_{loc} consists of a depth-wise 3×3 convolution, BN, SiLU, and a point-wise convolution. Its input and output have d channels.

The channel gate is

$$A_c(\mathbf{M}_{\mathrm{abs}}) = \sigma\left(W_2 * \mathrm{ReLU}\left(W_1 * \mathrm{GAP}(\mathbf{M}_{\mathrm{abs}})\right)\right), \tag{46}$$

where GAP averages over both spatial dimensions. The learnable 1×1 convolutions W_1 and W_2 map d channels to $d/16$ channels and back to d channels, respectively. Thus $A_c(\mathbf{M}_{\mathrm{abs}})$ has shape $d \times 1 \times 1$.

For the spatial gate, let $c \in \{0, \ldots, d-1\}$ index channels. Define the axis-wise averages

$$\begin{aligned} \mathbf{M}_h[c, v] &= \frac{1}{w} \sum_{u=0}^{w-1} \mathbf{M}_{\text{abs}}[c, v, u], \\ \mathbf{M}_w[c, u] &= \frac{1}{h} \sum_{v=0}^{h-1} \mathbf{M}_{\text{abs}}[c, v, u]. \end{aligned} \tag{47}$$

Their shapes are $d \times h$ and $d \times w$, respectively.

Let Ψ denote the following one-dimensional filtering operation: split channels into four equal groups, apply a depth-wise convolution to each group using kernel sizes $(3, 5, 7, 9)$ in order, and concatenate the outputs along channels. Padding preserves the length of each axis. The filtering parameters are shared between the two axes, whereas GN_h and GN_w are separately parameterized group normalizations with four groups. The axis gates are

$$\mathbf{a}_h = \sigma\left(\mathrm{GN}_h(\Psi(\mathbf{M}_h))\right), \qquad \mathbf{a}_w = \sigma\left(\mathrm{GN}_w(\Psi(\mathbf{M}_w))\right). \tag{48}$$

Their shapes are $d \times h$ and $d \times w$, respectively. The spatial gate is

$$A_s(\mathbf{M}_{\text{abs}})[c, v, u] = \mathbf{a}_h[c, v]\mathbf{a}_w[c, u], \tag{49}$$

with shape $d \times h \times w$.

The detail output is

$$\begin{aligned} \mathbf{D} &= \Phi_{\text{detail}}(\mathbf{H}) \\ &= \mathbf{Q}_{\text{loc}} + (A_c(\mathbf{M}_{\text{abs}}) \odot A_s(\mathbf{M}_{\text{abs}})) \odot \phi_{\text{ref}}(\mathbf{Q}_{\text{loc}}). \end{aligned} \tag{50}$$

The operator ϕ_{ref} consists of three consecutive point-wise convolution–BN–SiLU blocks, each preserving d channels. Broadcasting expands the channel gate across spatial dimensions, so every term in Eq. 50 has shape $d \times h \times w$. For finite preactivations, sigmoid gate values lie in $(0, 1)$. This does not bound the magnitude of $\mathbf{D}$, because the refinement features and residual addition are unconstrained.

Proposition 2 (Conditional context dependence). Fix the selected patch, local feature $\mathbf{F}$, decomposition settings, model parameters, and normalization state. For any two supplied contexts $\mathbf{C}$ and $\mathbf{C}'$ of the required shape, the detail output is identical, and

$$\begin{aligned} &\mathcal{X}(\mathbf{F}, \mathbf{C}') - \mathcal{X}(\mathbf{F}, \mathbf{C}) \\ &\qquad = \Phi_{\text{sem}}(\mathbf{L}, \mathbf{C}') - \Phi_{\text{sem}}(\mathbf{L}, \mathbf{C}). \end{aligned} \tag{51}$$

Proof. Under the fixed decomposition settings, $\mathbf{L}$ and $\mathbf{H}$ depend only on $\mathbf{F}$. The two evaluations therefore have the same $\mathbf{D} = \Phi_{\text{detail}}(\mathbf{H})$. Subtracting their block outputs cancels this common detail term, giving Eq. 51. □

The proposition characterizes direct computational dependence on the supplied context at a fixed patch and feature level. It does not imply statistical independence between global and local features, independence from global-guided routing, or independence of the learned detail parameters from joint optimization. For training-time batch normalization, the comparison also holds the local batch fixed. Neither this proposition nor the reconstruction identity proves that asymmetric conditioning must outperform symmetric conditioning. That comparison is empirical, as reported in Sec. 5.3.

E Limitations and Discussion

The current study focuses on sparse global–local detection in RGB aerial imagery. CrossFreq-UHR is evaluated on two UHR benchmarks with different scene and object distributions, demonstrating its effectiveness within this setting. Its transferability across sensors, geographic regions, and other remote-sensing modalities remains to be systematically investigated. The current implementation applies CrossFreq-UHR to two predefined pyramid levels with a shared decomposition setting. An input-adaptive extension could select the interaction levels and decomposition scales according to object size and scene composition. Beyond object detection, spatially aligned cross-frequency interaction may also benefit related tasks such as instance segmentation, change detection, and multi-temporal analysis, which we leave for future work.

F BROADER IMPACTS

CrossFreq-UHR aims to improve the analysis of ultra-high-resolution remote-sensing imagery, with potential applications in environmental monitoring, infrastructure inspection, and disaster assessment. Such capabilities may also raise privacy and surveillance concerns when applied to sensitive locations or activities. Responsible use requires appropriate data governance and human oversight, particularly when detection errors could affect consequential decisions.