# SCA-DETR: STRUCTURED CROSS-SCALE ATTENTION FOR DETECTION TRANSFORMERS WITH HIERARCHICAL BACKBONES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Detection Transformers are a leading paradigm for end-to-end object detection, using learned object queries to attend to encoded image features and directly produce a set of detections. With hierarchical backbones, these detectors must integrate multi-scale features whose resolution, locality, and semantics differ across stages, making cross-scale interaction difficult to structure efficiently. Existing encoder designs face a trade-off between efficient local convolutional mixing, which restricts the scope of cross-scale exchange, and more expressive dense multi-scale attention, which incurs greater computational and memory cost. We introduce **SCA-DETR**, a Detection Transformer that uses Structured Cross-Scale Attention to organize information exchange across hierarchical backbone stages. To determine how this exchange should be structured, we use alignment diagnostics to distinguish geometric support from residual representational mismatch across adjacent backbone stages. Guided by this analysis, SCA uses the deterministic scale relationship between adjacent stages to center a local source neighborhood for each target location, constraining where cross-scale exchange occurs. Within this support, structured cross-scale attention determines what information to transfer, followed by residual refinement as the updates propagate in both directions through the hierarchy. On COCO val2017, SCA-DETR reaches 56.08 AP with TinyViT under the DEIM pipeline. With Swin-T, SCA improves the corresponding RT-DETR and DEIM-D-FINE baselines by 0.71 and 0.75 AP, respectively. Under compiled FP32 inference on an A100 GPU, the evaluated SCA configurations achieve 11–15% lower model-forward latency than their corresponding reference implementations. Additional results across hierarchical backbone families and on VisDrone2019-DET and PASCAL VOC support its applicability across architectures and detection settings.

## 1 INTRODUCTION

Detection transformers (DETRs) recast object detection as direct set prediction, reducing reliance on hand-designed anchors and duplicate-removal post-processing Carion et al. (2020). A visual backbone first extracts image features, which the transformer encoder contextualizes before the decoder uses learned object queries to retrieve object-level information and produce a set of predictions. During training, one-to-one bipartite matching assigns these predictions to ground-truth objects. Through attention, this encoder-decoder pathway enables long-range interaction between image features and object queries beyond strictly local convolutional processing. Early DETR models nevertheless exhibited slow convergence and limited support for efficient multi-scale feature processing. Subsequent DETR-family models addressed these limitations through sparse multi-scale attention, while refined query design, denoising-based training, and stronger assignment strategies further improved training and prediction quality Zhu et al. (2021); Zhang et al. (2023); Zong et al. (2023). Together, these advances have established DETRs as an active and competitive family of end-to-end detectors.

Backbone representations are central to DETR performance because they define the visual information available to the transformer encoder and decoder. Vision Transformer backbones apply the same attention-based modeling principle to this feature-extraction stage, producing representations



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.

that capture broader visual context Dosovitskiy et al. (2021). This feature-extraction stage must also preserve information across scale: small, medium, and large objects place different demands on spatial detail, semantic abstraction, and cross-level exchange. Hierarchical vision transformer backbones address this requirement by producing multi-stage features that retain spatial detail at shallower levels while increasing semantic abstraction at deeper levels Liu et al. (2021). However, these feature stages differ in resolution, locality, and semantic abstraction, requiring the detector to exchange information across representations that are structurally related but not directly aligned. Recent DETR variants have made multi-scale processing more efficient through sparse attention, interleaved encoder updates, and hybrid encoder designs Zhu et al. (2021); Li et al. (2023); Zhao et al. (2024b). Although these designs reduce multi-scale processing cost, they do not fully specify how correspondence should be structured across hierarchical stages. Restricting cross-scale interaction may limit information exchange, whereas expanding it increases encoder computation and memory demands. The practical challenge is to improve cross-scale information exchange while maintaining low inference latency.

In this paper, we introduce SCA-DETR, a DETR-family detector built around *Structured Cross-Scale Attention* for encoder-side fusion of hierarchical backbone features, as shown in Figure 1. The central design principle of SCA is to use the spatial organization of hierarchical backbone features to constrain where cross-scale interaction is performed, while learned attention determines the information transferred and refined. SCA structures cross-scale exchange along adjacent levels of the backbone hierarchy, but restricts each directed update to a local source window centered by the deterministic scale relation between feature maps. Each target location attends near its mapped position in the adjacent source level rather than searching the full feature hierarchy; learned attention, gated fusion, and lightweight refinement handle residual displacement and semantic differences between stages, yielding a modular encoder-side fusion block for DETR architectures with multi-stage backbones.

SCA improves detection performance across contemporary DETR pipelines with hierarchical backbones. On COCO val2017 Lin et al. (2014), it raises RT-DETR with Swin-T from 53.31 to 54.02 AP and DEIM-D-FINE with Swin-T from 54.33 to 55.08 AP, with the latter improving across all reported object scales. The strongest configuration, SCA-DEIM-D-FINE with TinyViT, achieves 56.08 AP, exceeding the DEIM-D-FINE-L reference with HGNetv2-L (Huang et al., 2025) by 1.38 AP. These accuracy gains accompany lower measured model-forward latency: the evaluated Swin-T and TinyViT configurations reduce latency by approximately 11–15% relative to their corresponding reference implementations. Further gains across hierarchical backbone families and on VisDrone2019-DET Zhu et al. (2022) and PASCAL VOC Everingham et al. (2010) support structured cross-scale fusion as a transferable encoder-side module. In summary, the contributions of this work are as follows:

1. We formulate cross-scale fusion in hierarchical-backbone DETR detectors as a structured correspondence problem and introduce alignment diagnostics to separate geometric support from residual representational mismatch across feature stages.
2. We propose Structured Cross-Scale Attention, an encoder-side fusion operator that maps each target token to a hierarchy-induced local source support in an adjacent feature level, computes content-adaptive cross-scale attention within that support, and applies spatially adaptive gated fusion with depthwise-separable residual refinement during bidirectional hierarchy updates before decoding.
3. We evaluate SCA across detector architectures, hierarchical backbones, and detection datasets, demonstrating accuracy improvements together with lower measured inference latency in the evaluated deployments, and further gains on VisDrone2019-DET and PASCAL VOC.

## 2 Related Work

Prior literature relevant to SCA-DETR spans three connected areas of object detection research. Research on DETR and real-time end-to-end detectors defines the detector framework, multi-scale encoders address cross-resolution feature exchange, and hierarchical backbones provide the ordered stages over which that exchange occurs. Together, these areas connect end-to-end detector design with multi-scale interaction over hierarchical features.

**DETR and real-time detectors.** Since DETR (Carion et al., 2020), the family has advanced along several complementary axes. Deformable DETR (Zhu et al., 2021) reduced the cost of multi-scale attention; DAB-DETR, DN-DETR, DINO, and Stable-DINO (Liu et al., 2022a; Li et al., 2022; Zhang et al., 2023; Liu et al., 2023) refined query formulation, denoising, and matching; and Co-DETR, Group-DETR, and MS-DETR (Zong et al., 2023; Chen et al., 2023; Zhao et al., 2024a) strengthened assignment-based supervision. MI-DETR (Nan et al., 2025) expanded decoder-side multi-level interaction, while Focus-DETR and Lite DETR (Zheng et al., 2023; Li et al., 2023) revisited encoder workload and multi-scale updates. In the real-time regime, RT-DETR (Zhao et al., 2024b) introduced an efficient hybrid encoder, followed by improvements to training, deployment, localization, and matching in RT-DETRv2, D-FINE, and DEIM (Lv et al., 2024; Peng et al., 2025; Huang et al., 2025); LW-DETR (Chen et al., 2024) explores a parallel lightweight ViT design. These works establish the detector lineage into which SCA-DETR is integrated. SCA instead targets the structure of encoder-side exchange between adjacent hierarchical backbone stages, positioning it as a fusion contribution rather than another query, assignment, regression, or training refinement.

**Multi-scale encoding and fusion.** Multi-scale representations have long been central to traditional object detection. Feature Pyramid Networks established a widely adopted top-down pathway with lateral connections, enriching higher-resolution features with coarse semantic context (Lin et al., 2017). In transformer-based detectors, Deformable DETR subsequently enabled efficient multi-scale interaction through sparse, content-dependent sampling (Zhu et al., 2021). Later encoders reduce computation through token sparsification or selection in Sparse DETR and Focus-DETR (Roh et al., 2022; Zheng et al., 2023), interleaved feature-level updates in Lite DETR (Li et al., 2023), or the decoupled intra-scale encoding and lightweight cross-scale fusion of RT-DETR (Zhao et al., 2024b). SCA instead structures this exchange through the ordered geometry of hierarchical backbone stages: each target location maps to a local support region in an adjacent level, within which learned attention selects the information to transfer before refinement.

**Hierarchical vision backbones.** Vision Transformers model images as token sequences using self-attention, typically retaining a single spatial resolution through the encoder (Dosovitskiy et al., 2021). For dense prediction, hierarchical variants instead expose pyramid-like feature stages at progressively coarser resolutions. Swin Transformer constructs this hierarchy through shifted-window attention (Liu et al., 2021), while PVTv2, NAT, and Hiera explore pyramid representations, neighborhood attention, and simplified hierarchical processing (Wang et al., 2022; Hassani et al., 2023; Ryali et al., 2023). FocalNet provides a related hierarchical representation through focal modulation (Yang et al., 2022). These backbones provide ordered multi-stage feature maps for dense detectors, but their stages differ in resolution, locality, tokenization, and semantic abstraction. In DETR detectors that consume such features, backbone substitution alone is therefore insufficient, leaving cross-scale feature interaction as a detector-side design problem.

# 3 METHOD

This section defines Structured Cross-Scale Attention (SCA). Figure 1 provides an overview of the full SCA-DETR architecture. We first describe the hierarchical backbone features and their alignment, then specify the encoder interface and SCA operation. The section then presents the bidirectional hierarchy update and concludes with decoder integration.

## 3.1 HIERARCHICAL BACKBONE FEATURES AND ALIGNMENT

SCA-DETR is built for the setting where an RT-DETR-style detector is paired with a hierarchical vision transformer backbone. The backbone produces a set of multi-scale feature maps $\mathcal{S} = \{\mathbf{S}_\ell\}_{\ell \in \mathcal{L}}$, $\mathbf{S}_\ell \in \mathbb{R}^{B \times C_\ell \times H_\ell \times W_\ell}$, where $\mathcal{L}$ is the set of selected detector feature levels, $\ell \in \mathcal{L}$ indexes a level, and $B$ is the batch size; $C_\ell$ is the backbone channel dimension, and $H_\ell$ and $W_\ell$ are the grid height and width of level $\ell$, respectively. In the detector configurations studied in this paper, the exported hierarchy contains three adjacent levels with progressively coarser strides.

The central design problem is how to exchange information between adjacent levels of a hierarchical ViT feature pyramid without treating cross-scale fusion as an unconstrained global search. CNN feature pyramids inherit a strong local translation structure from convolutional stages He et al. (2016), while hierarchical ViT backbones such as Swin Transformer introduce stage-wise tokenization, patch merging, windowed interaction, and coarser semantic abstraction Liu et al. (2021). These

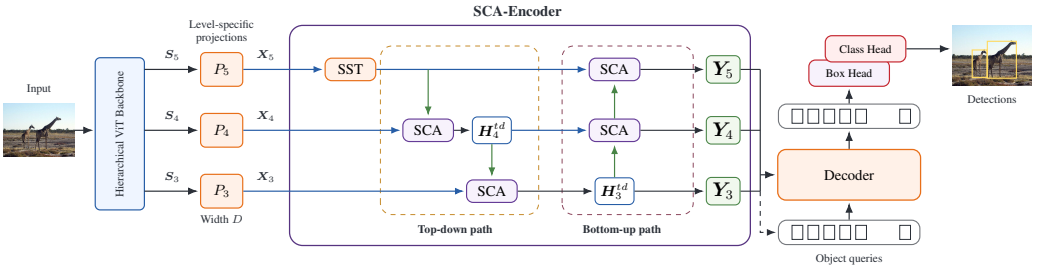


Figure 1: **SCA-DETR architecture overview.** A hierarchical ViT backbone exports multi-scale features, which are projected to a common detector width. SCA updates the hierarchy through structured adjacent-level interaction before the fused features are passed to the detector decoder.

operations do not remove spatial structure, but they change the form of the correspondence that a detector encoder should exploit. Global cross-scale attention would give the encoder flexibility, but would not take advantage of the known stride relation between hierarchical feature maps. Fixed geometric alignment uses this relation, but leaves no mechanism for residual displacement or scale-dependent semantic change. We therefore evaluate whether the hierarchy supports a middle ground: local access centered by geometry, with learned exchange reserved for the remaining cross-scale mismatch.

Adjacent detector levels are related by the backbone's stride and tokenization structure, so a target token at level $t$ has a natural source-level neighborhood at an adjacent level $s$. For a target-grid position $p = (i, j)$, with zero-based row index $i$ and column index $j$, the deterministic cross-scale map $\phi_{t \leftarrow s}(p)$ denotes the source-grid location induced by the known scale relation between the two feature maps. In hierarchical ViTs, patch merging and windowed attention make this relation a plausible geometric prior, but not automatically a complete correspondence model: response energy may spread or shift around the deterministic parent location, and adjacent levels may encode different content even after alignment. We evaluate this prior using two diagnostics. Impulse-response analysis propagates a localized input impulse through the frozen backbone, measures response-centroid displacement from the deterministic mapped location, and compares it with a conservative architecture-conditioned envelope. Natural-image feature consistency evaluates whether deterministic alignment restores agreement across scales. Appendices B.1–B.2 define the mapping and diagnostic envelope, while B.3–B.4 report measurement settings and supplementary evidence. Section 4.3 summarizes the results. Together, these diagnostics motivate local source neighborhoods anchored by the deterministic map: the measured residual displacement motivates access to surrounding source tokens, while the remaining feature mismatch motivates learned exchange within that neighborhood.

### 3.2 Encoder Interface

Each backbone feature is first mapped by a level-specific channel projection to the common detector width $D$, $\mathbf{X}_\ell = \mathrm{Proj}_\ell(\mathbf{S}_\ell) \in \mathbb{R}^{B \times D \times H_\ell \times W_\ell}$. This projection defines the interface between the backbone and the detector encoder while preserving the spatial resolution of each level.

Let $\ell_c = \max(\mathcal{L})$ denote the coarsest selected level. Following the efficient hybrid-encoder design of RT-DETR Zhao et al. (2024b), the single-scale transformer (SST) updates only this level, $\mathbf{X}_{\ell_c} \leftarrow \mathcal{E}_{\mathrm{SST}}(\mathbf{X}_{\ell_c})$, where the lower spatial resolution makes global intra-scale interaction most economical. The remaining projected levels enter cross-scale fusion directly. SCA then updates the resulting multi-scale hierarchy and produces the fused feature set $\{\mathbf{Y}_\ell\}_{\ell \in \mathcal{L}} = \mathcal{F}_{\mathrm{SCA}}(\{\mathbf{X}_\ell\}_{\ell \in \mathcal{L}})$ passed to the detector decoder.

### 3.3 Structured Cross-Scale Attention

Structured Cross-Scale Attention updates each target level by attending to a local neighborhood in an adjacent source level, as illustrated in Figure 2. Each directed edge uses independently parameterized attention, gated fusion, and refinement blocks. For target level $t$, source level $s$, and target position $p$, the mapped local support is

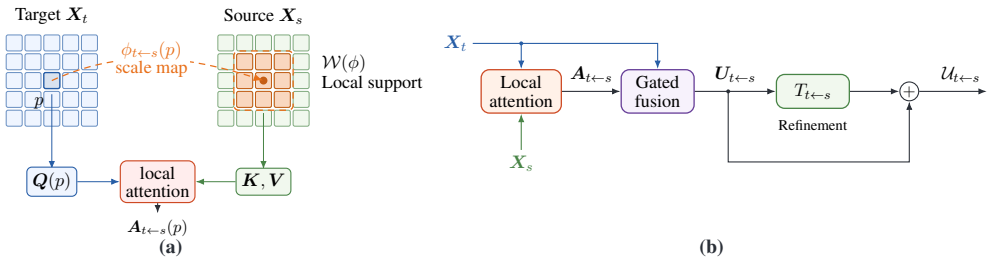


Figure 2: **Structured Cross-Scale Attention module.** (a) Local cross-scale attention: the target position $p$ maps to a local source window (schematic); Appendix D.1 details the direction-specific supports. (b) SCA directed update: the aligned response is fused with the target state through a learned gate and refined by a lightweight residual tail.

$$\Omega_{t\leftarrow s}(p) = \mathcal{W}_{t\leftarrow s}\big(\phi_{t\leftarrow s}(p)\big), \tag{1}$$

where $\phi_{t\leftarrow s}(p)$ gives the deterministic source-grid correspondence and $\mathcal{W}_{t\leftarrow s}$ constructs the direction-specific local sampling window. We use $3 \times 3$ top-down and $4 \times 4$ bottom-up windows, with the latter centred on the half-token correspondence induced by the twofold stride relation. The sampling operator incorporates grid discretization and boundary clamping. Appendix B.1 defines the continuous correspondence, and Appendix D.1 explains its direction-specific sampling geometry.

The alignment diagnostics motivate local access around this correspondence. Because source tokens cover larger image regions in the top-down direction, compact coarse-grid support provides broad spatial context, while the wider bottom-up window gathers surrounding fine-scale features. Appendix D.2 evaluates the adopted supports through configuration ablations.

For a directed update from $s$ to $t$, $\mathbf{X}_t$ and $\mathbf{X}_s$ denote the current target and source feature maps supplied to the SCA operator. Let $q_{p,k}$ denote the $k$-th sample in $\Omega_{t\leftarrow s}(p)$, which contains $K_{t\leftarrow s}$ sampling slots. Each target feature $\mathbf{X}_t(p)$ forms a query, while the sampled source features $\mathbf{X}_s(q_{p,k})$ form keys and values through learned projections $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$. With query/key dimension $d$ and learned positional bias $b_k$ for window offset $k$, the attention weight $\alpha_{p,k}$ is normalized over the $K_{t\leftarrow s}$ sampling slots, giving the structured cross-scale response

$$\begin{aligned} \mathbf{A}_{t\leftarrow s}(p) &= \sum_{k=1}^{K_{t\leftarrow s}} \alpha_{p,k} \mathbf{W}_V \mathbf{X}_s(q_{p,k}), \\ \alpha_{p,k} &= \mathrm{softmax}_k\left(\frac{(\mathbf{W}_Q \mathbf{X}_t(p))^\top (\mathbf{W}_K \mathbf{X}_s(q_{p,k}))}{\sqrt{d}} + b_k\right). \end{aligned} \tag{2}$$

Each directed edge learns its own positional biases, shared across target locations, to encode preferences for relative positions within the sampling window. The response $\mathbf{A}_{t\leftarrow s}$ has the spatial resolution and channel width of the target level. The hierarchy restricts the candidate source positions, while the attention weights remain content-adaptive.

The aligned response is then fused with the target state through a learned spatial gate. The gate is predicted from the target/aligned pair and modulates the two streams before they are mapped back to the detector width:

$$\begin{aligned} \mathbf{G}_{t\leftarrow s} &= \sigma\left(g_{t\leftarrow s}(\mathbf{X}_t, \mathbf{A}_{t\leftarrow s})\right), \\ \mathbf{U}_{t\leftarrow s} &= f_{t\leftarrow s}\left(\mathbf{G}_{t\leftarrow s} \odot \mathbf{X}_t, (1 - \mathbf{G}_{t\leftarrow s}) \odot \mathbf{A}_{t\leftarrow s}\right). \end{aligned} \tag{3}$$

where $\mathbf{G}_{t\leftarrow s}$ is the gate and $\mathbf{U}_{t\leftarrow s}$ is the fused feature map, $\sigma$ denotes the element-wise sigmoid, $\odot$ denotes element-wise multiplication, $g_{t\leftarrow s}$ predicts the gate, and $f_{t\leftarrow s}$ maps the gated streams back to the detector width. The gate regulates how much aligned cross-scale information enters each target position.

Finally, a lightweight residual tail smooths local spatial artifacts and remixes channels, completing the directed update operator $\mathcal{U}_{t\leftarrow s}$:

$$\mathcal{U}_{t\leftarrow s}(\mathbf{X}_t, \mathbf{X}_s) = \mathbf{U}_{t\leftarrow s} + T_{t\leftarrow s}(\mathbf{U}_{t\leftarrow s}), \tag{4}$$

where $T_{t\leftarrow s}$ applies spatial depthwise filtering independently over channels, followed by pointwise channel mixing, with an outer residual connection around the block. The tail projects the $D$-channel features to an internal width of $2D$, applies three residual depthwise–pointwise blocks, and projects back to $D$ channels before the outer residual addition. This efficient refinement pattern follows modern lightweight convolutional design Liu et al. (2022b); Wang et al. (2024). Geometry therefore defines the candidate source region, while attention, gated fusion, and refinement select and revise the transferred information.

### 3.4 Bidirectional Hierarchy Update and Decoder Integration

The SCA encoder applies the directed update in both top-down and bottom-up order, as illustrated in Figure 1. The top-down pass propagates coarse semantic context toward finer spatial levels. Let $L$ and $H$ denote the finest and coarsest selected levels, and let $\mathbf{H}_\ell^{\mathrm{td}}$ denote the top-down state at level $\ell$, initialized by $\mathbf{H}_H^{\mathrm{td}} = \mathbf{X}_H$. The top-down update is

$$\mathbf{H}_\ell^{\mathrm{td}} = \mathcal{U}_{\ell\leftarrow\ell+1}\left(\mathbf{X}_\ell, \mathbf{H}_{\ell+1}^{\mathrm{td}}\right), \qquad \ell = H-1, \ldots, L. \tag{5}$$

The bottom-up pass then propagates refined fine-level information toward coarser levels. Starting from $\mathbf{Y}_L = \mathbf{H}_L^{\mathrm{td}}$, the update is

$$\mathbf{Y}_\ell = \mathcal{U}_{\ell\leftarrow\ell-1}\left(\mathbf{H}_\ell^{\mathrm{td}}, \mathbf{Y}_{\ell-1}\right), \qquad \ell = L+1, \ldots, H. \tag{6}$$

Each pass feeds the output of one edge into the next, with every edge using the directed SCA operator defined in the previous subsection.

The resulting feature set $\{\mathbf{Y}_\ell\}_{\ell\in\mathcal{L}}$ retains the common detector width and the spatial resolution of each input level. SCA therefore acts as an interchangeable encoder-side multi-scale fusion module: it replaces cross-scale interaction while preserving the feature interface consumed by the detector-specific decoder and prediction heads. This interface separates the structured fusion mechanism from the downstream detector design.

We instantiate this module with RT-DETR, D-FINE, and DEIM variants using hierarchical ViT backbones. The same formulation is compatible with modern multi-scale DETR architectures that expose adjacent hierarchical feature levels for encoder-side fusion and accept the resulting multi-scale feature set.

## 4 Experiments

In this section, we evaluate SCA-DETR on COCO, examine the selected design through schedule, ablation, and cross-backbone evidence, and report external validation on VisDrone2019-DET and PASCAL VOC. We use these experiments to assess whether Structured Cross-Scale Attention improves hierarchy-aware fusion in RT-DETR-style detectors while preserving a practical accuracy-efficiency tradeoff.

### 4.1 Experimental Setup

**Datasets and metrics.** We use COCO 2017 Lin et al. (2014) as the primary benchmark, training on 118,287 images and evaluating on the 5,000-image validation set over 80 categories. We report AP, $\mathrm{AP}_{50}$, $\mathrm{AP}_{75}$, and scale-specific AP. Additional transfer results on VisDrone2019-DET and PASCAL VOC are reported in Section 4.6.

**Implementation details.** We conduct the main evaluation, component ablations, and external-dataset evaluation within the RT-DETR pipeline. We additionally evaluate SCA with D-FINE and DEIM Huang et al. (2025) to assess compatibility across detector variants. Each baseline–SCA comparison uses a matched protocol within its detector pipeline. SCA replaces the encoder-side cross-scale fusion module while retaining the detector-specific decoder and prediction heads. Detailed implementation settings are provided in the appendix.

### 4.2 Main Results

We evaluate SCA on COCO val2017 using hierarchical backbones, whose adjacent feature levels provide the spatial structure for cross-scale interaction. Table 1 places these results in the context of

published DETR-family detectors, while Table 2 compares SCA with the corresponding baseline in each detector pipeline. Adding SCA improves AP by 0.51–0.96 points across all four detector and backbone configurations. This pattern suggests that the encoder-side benefit persists as the surrounding detector changes. In RT-DETR with Swin-T, SCA raises AP from 53.31 to 54.02 (+0.71), with substantial gains for small (+0.49 $AP_S$) and medium (+0.77 $AP_M$) objects. These scale-specific gains are consistent with SCA's goal of improving exchange across hierarchical features, while the subsequent ablations examine the contributions of its individual components.

The AP improvement also extends to D-FINE and DEIM-D-FINE. With Swin-T, SCA raises D-FINE from 54.31 to 54.82 AP and DEIM-D-FINE from 54.33 to 55.08 AP. In DEIM-D-FINE, the improvement spans every reported metric, including gains of 0.81 $AP_S$, 0.76 $AP_M$, and 0.44 $AP_L$. This breadth supports SCA's use as an encoder-side fusion module across detector pipelines while retaining their respective decoders and prediction heads. The gain persists with TinyViT, where SCA adds 0.96 AP to reach 56.08 and leads Table 2 on all six reported metrics. For broader context, the Swin-T and TinyViT SCA-DEIM-D-FINE variants exceed the published DEIM-D-FINE-L/HGNetv2-L result in Table 1 by 0.38 and 1.38 AP, respectively. Together, the consistent gains position SCA as an effective cross-scale fusion method for modern DETR-family detectors using compact hierarchical vision transformer backbones.

Table 1: Published COCO val2017 results for representative DETR-family detectors. Each row follows its source paper's model and training configuration; these results provide literature context for Table 2.

| Method | Backbone | #Query | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|
| *DETR-family references* | | | | | | | | |
| DINO-4Scale (Zhang et al., 2023) | ResNet-50 | 900 | 50.9 | 69.0 | 55.3 | 34.6 | 54.1 | 64.6 |
| Lite DETR (Li et al., 2023) | ResNet-50 | 900 | 50.4 | 68.5 | 54.6 | 33.5 | 53.6 | 65.5 |
| RT-DETR (Zhao et al., 2024b) | ResNet-50 | 300 | 53.1 | 71.3 | 57.7 | 34.8 | 58.0 | 70.0 |
| D-FINE-L (Peng et al., 2025) | HGNetv2-L | 300 | 54.0 | 71.6 | 58.4 | 36.5 | 58.0 | 71.9 |
| DEIM-D-FINE-L (Huang et al., 2025) | HGNetv2-L | 300 | 54.7 | 72.4 | 59.4 | 36.9 | 59.6 | 71.8 |

Table 2: COCO val2017 baseline–SCA comparisons within each detector pipeline and backbone. Each pair uses its corresponding detector configuration.

| Method | Backbone | #Query | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|
| RT-DETR | Swin-T | 300 | 53.31 | 71.76 | 57.64 | 35.36 | 57.65 | 71.45 |
| SCA-DETR | Swin-T | 300 | 54.02 | 72.04 | 58.21 | 35.85 | 58.42 | 71.48 |
| D-FINE | Swin-T | 300 | 54.31 | 72.02 | 59.01 | 36.65 | 58.39 | 71.74 |
| SCA-D-FINE | Swin-T | 300 | 54.82 | 72.44 | 59.27 | 36.30 | 59.30 | 72.30 |
| DEIM-D-FINE | Swin-T | 300 | 54.33 | 72.01 | 59.21 | 35.60 | 58.48 | 72.03 |
| SCA-DEIM-D-FINE | Swin-T | 300 | 55.08 | 72.82 | 59.72 | 36.41 | 59.24 | 72.47 |
| DEIM-D-FINE | TinyViT | 300 | 55.12 | 72.73 | 59.95 | 37.43 | 59.71 | 71.66 |
| SCA-DEIM-D-FINE | TinyViT | 300 | **56.08** | **73.78** | **61.18** | **37.68** | **60.27** | **73.66** |

### 4.3 Alignment Diagnostics

Table 3 reports the alignment diagnostics introduced in Section 3.1 for Swin-T and ResNet-50. Across all evaluated stages, the 95th-percentile center-of-mass (COM) drift remains below the corresponding architecture-derived bound, supporting deterministic scale mapping as a local geometric anchor. On natural images, however, adjacent Swin-T stages retain near-zero global and local cosine similarity after deterministic alignment, whereas ResNet-50 exhibits substantially higher feature consistency. These findings motivate SCA's combination of geometry-centered local support and learned cross-scale exchange: the deterministic relation anchors the source neighborhood, while learned interaction addresses the residual representational mismatch. Per-stage drift statistics and visualizations are reported in Appendix B.3; additional natural-image consistency measures are reported in Appendix B.4.

Table 3: **Alignment-diagnostic summary.** Drift and bound values are in input pixels; per-stage statistics and measurement settings are provided in Appendix B.

| | Impulse locality: P95 COM $\leq$ bound | | | Global cosine | | Local cosine | |
|---|---|---|---|---|---|---|---|
| Backbone | S3 | S4 | S5 | S4→S3 | S5→S4 | S4→S3 | S5→S4 |
| Swin-T | $17.82 \leq 53.74$ | $54.86 \leq 263.04$ | $92.67 \leq 410.12$ | -0.0339 | 0.0463 | -0.0606 | 0.0221 |
| ResNet-50 | $1.43 \leq 2.83$ | $5.68 \leq 8.49$ | $16.66 \leq 19.80$ | 0.6950 | 0.3662 | 0.6419 | 0.3721 |

### 4.4 Computational Cost and Inference Efficiency

Across the three evaluated baseline–SCA pairs, SCA improves COCO AP while reducing measured model-forward latency. Table 4 reports these comparisons at $640 \times 640$ and batch size one on an A100 80GB GPU under compiled FP32 inference. Appendix C reports the full detector comparison, including D-FINE and memory usage; Appendix A.4 specifies the protocol.

Table 4: Accuracy and inference cost of selected detector configurations. AP is from COCO val2017; latency is the model-only mean across five trials.

| Model | Backbone | AP | Params (M) | Estimated GFLOPs | Latency (ms/image) |
|---|---|---|---|---|---|
| RT-DETR | Swin-T | 53.31 | 46.36 | 68.63 | 17.38 |
| SCA-DETR | Swin-T | 54.02 | 44.61 | 67.51 | **14.70** |
| DEIM-D-FINE | Swin-T | 54.33 | 44.68 | 60.54 | 15.82 |
| SCA-DEIM-D-FINE | Swin-T | 55.08 | 45.54 | 63.04 | **14.05** |
| DEIM-D-FINE | TinyViT | 55.12 | 37.73 | 58.73 | 17.05 |
| SCA-DEIM-D-FINE | TinyViT | **56.08** | 38.59 | 61.24 | **15.10** |

Across the three pairs, SCA raises AP by $0.71$, $0.75$, and $0.96$ while lowering model-forward latency by $15.5\%$, $11.2\%$, and $11.4\%$, respectively. Among all configurations in Appendix C, SCA-DEIM-D-FINE with TinyViT has the highest AP ($56.08$); the Swin-T variants of SCA-DEIM-D-FINE and SCA-D-FINE both run at approximately $14.05$ ms/image.

### 4.5 Ablation Studies

**Architecture ablations.** The gains in Section 4.2 motivate a closer look at SCA's encoder design. Table 5 examines the contributions of the encoder components under a controlled 12-epoch RT-DETR setting with Swin-T on COCO val2017. We begin by removing RT-DETR's HybridEncoder. The no-encoder row omits it entirely, while the $S_5$-only row retains its single-scale transformer on the coarsest feature, raising AP from 37.2 to 38.2. The first SCA configuration then introduces top-down structured local cross-scale attention, allowing each target location to attend within a mapped $3 \times 3$ neighborhood of the adjacent coarser feature level. It reaches 43.5 AP, 5.3 points above the $S_5$-only row and the largest increase in the reported design sequence.

Adding a learned per-offset attention bias raises AP from 43.5 to 44.0 (+0.5), allowing the model to favor positions within the fixed local support. Adding the residual refinement tail then raises AP to 46.0 (+2.0). This gain suggests that mapped local attention still benefits from refinement to address residual differences between feature levels. Adding the bottom-up update with centred $4 \times 4$ support raises AP to 48.6 (+2.6), with improvements in every reported metric, including +2.9 $AP_{75}$. This supports returning refined fine-level information toward coarser stages after the top-down pass. Together, these components bring SCA to 48.6 AP, surpassing RT-DETR's full HybridEncoder reference (47.5 AP) by 1.1 points.

**Additional ablations and comparisons.** Appendix D examines local-support geometry and the accuracy–compute tradeoff across top-down and bottom-up support configurations. Appendix E reports the preliminary refinement-depth and channel-expansion study supporting the retained capacity setting. Appendix F extends the backbone comparisons to PVTv2, NAT, and Hiera.

Table 5: SCA component ablations on COCO val2017 with Swin-T. Training and evaluation settings are given in Appendix A.2. Checkmarks indicate enabled components; n/a indicates inapplicable components.

| Variant | $S_5$ enc. | TD SLA | Pos. bias | Tail | BU SLA | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| No encoder | – | – | – | – | – | 37.2 | 54.5 | 40.0 | 20.0 | 40.4 | 53.9 |
| + $S_5$ encoder | ✓ | – | – | – | – | 38.2 | 56.0 | 41.2 | 20.9 | 41.4 | 55.5 |
| + structured local attention (TD) | ✓ | ✓ | – | – | – | 43.5 | 61.0 | 46.9 | 24.8 | 47.1 | 62.2 |
| + learned positional offset bias | ✓ | ✓ | ✓ | – | – | 44.0 | 61.7 | 47.6 | 25.3 | 47.9 | 62.8 |
| + DW–PW refinement tail | ✓ | ✓ | ✓ | ✓ | – | 46.0 | 63.5 | 49.5 | 27.2 | 50.1 | 64.7 |
| + bidirectional update | ✓ | ✓ | ✓ | ✓ | ✓ | 48.6 | 66.2 | 52.4 | 29.9 | 53.0 | 66.8 |
| RT-DETR (HybridEncoder) | ✓ | n/a | n/a | n/a | n/a | 47.5 | 65.1 | 51.1 | 28.5 | 51.8 | 66.0 |

### 4.6 Additional Results

We evaluate SCA-DETR on VisDrone2019-DET (Zhu et al., 2022) and PASCAL VOC (Everingham et al., 2010) to assess its performance beyond COCO, covering dense aerial imagery and standard object detection transfer. Table 6 compares SCA-DETR with Swin-T against reference detectors from literature. The reference results for VisDrone are drawn from UAV-DETR (Zhang et al., 2025), and the VOC references follow the evaluations reported by Xu et al. (2023) and Ma et al. (2023). The VOC models are trained on VOC2007+2012 trainval and evaluated on VOC2007 test.

Table 6: Additional detection results on VisDrone2019-DET validation (left) and PASCAL VOC2007 test (right). We report COCO-style AP and $AP_{50}$.

| Method | Backbone | AP | $AP_{50}$ |
|---|---|---|---|
| DETR | ResNet-50 | 24.1 | 40.1 |
| Deformable DETR | ResNet-50 | 27.1 | 42.2 |
| Sparse DETR | ResNet-50 | 27.3 | 42.5 |
| RT-DETR | ResNet-50 | 28.4 | 47.0 |
| UAV-DETR | ResNet-50 | 31.5 | 51.1 |
| **SCA-DETR** | **Swin-T** | **33.2** | **57.1** |

| Method | Backbone | AP | $AP_{50}$ |
|---|---|---|---|
| SMCA-DETR | ResNet-50 | 56.7 | 83.7 |
| DETR | ResNet-50 | 59.5 | 83.3 |
| Deformable DETR | ResNet-50 | 61.1 | 83.1 |
| H-Deformable DETR | ResNet-50 | 63.8 | 82.4 |
| RT-DETR | ResNet-50 | 67.3 | 87.7 |
| **SCA-DETR** | **Swin-T** | **68.7** | **88.8** |

On VisDrone2019-DET validation, the selected SCA-DETR checkpoint achieves 33.2 AP and 57.1 $AP_{50}$, exceeding the UAV-DETR reference by 1.7 and 6.0 points, respectively. On PASCAL VOC2007 test, SCA-DETR reaches 68.7 AP and 88.8 $AP_{50}$, exceeding the RT-DETR-R50 reference by 1.4 AP. These results extend the COCO findings to both dense aerial imagery and the VOC detection setting.

## 5 Conclusion

In this paper, we introduce SCA-DETR, a Detection Transformer that uses Structured Cross-Scale Attention to integrate hierarchical backbone features through geometry-guided local exchange. Alignment diagnostics motivate this design by showing that deterministic scale relationships provide a geometric anchor while representational differences remain across adjacent stages. SCA combines this structure with learned cross-scale interaction and bidirectional hierarchy updates, preserving a common multi-scale feature interface that enables its integration across hierarchical backbones and modern DETR architectures. Experiments demonstrate this applicability across backbone families and modern detector pipelines, with the strongest configuration achieving 56.08 AP on COCO val2017; additional results on VisDrone2019-DET and PASCAL VOC extend the evaluation to distinct detection settings. In the evaluated deployments, these accuracy improvements accompany lower model-forward latency, supporting SCA as a practical cross-scale fusion module. Together, these findings support structured cross-scale correspondence as a reusable principle for encoder-side fusion, connecting the spatial organization of hierarchical features with learned information exchange.

## AI USE STATEMENT

Generative AI tools were used to suggest edits for grammar, clarity, and phrasing in author-written manuscript text, and to review, edit, and complete portions of the SCA-DETR research code. The authors reviewed the AI-assisted text and code, checked the code as part of the experimental workflow, and take responsibility for the final manuscript, implementation, claims, and results.

## REPRODUCIBILITY STATEMENT

The research code used for SCA-DETR is provided as supplementary material. Experimental configurations and evaluation protocols are described in Appendix A.

## REFERENCES


Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, pp. 213–229, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58452-8. doi: 10.1007/978-3-030-58452-8_13.

Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group detr: Fast detr training with group-wise one-to-many assignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6633–6642, October 2023.

Qiang Chen, Xiangbo Su, Xinyu Zhang, Jian Wang, Jiahui Chen, Yunpeng Shen, Chuchu Han, Ziliang Chen, Weixiang Xu, Fanrong Li, Shan Zhang, Kun Yao, Errui Ding, Gang Zhang, and Jingdong Wang. LW-DETR: A transformer replacement to YOLO for real-time detection, 2024. URL https://arxiv.org/abs/2406.03459.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88 (2):303–338, Jun 2010. ISSN 1573-1405. doi: 10.1007/s11263-009-0275-4. URL https://doi.org/10.1007/s11263-009-0275-4.

Jiawei Fan, Chao Li, Xiaolong Liu, and Anbang Yao. ScaleKD: Strong vision transformers could be excellent teachers. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2022.

Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6185–6194, June 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, June 2016.

Shihua Huang, Zhichao Lu, Xiaodong Cun, Yongjun Yu, Xiao Zhou, and Xi Shen. Deim: Detr with improved matching for fast convergence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15162–15171, June 2025.

Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M. Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13619–13627, June 2022.

Feng Li, Ailing Zeng, Shilong Liu, Hao Zhang, Hongyang Li, Lei Zhang, and Lionel M. Ni. Lite detr: An interleaved multi-scale encoder for efficient detr. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18558–18567, June 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1. doi: 10.1007/978-3-319-10602-1_48.

Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2117–2125, July 2017.

Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: Dynamic anchor boxes are better queries for DETR. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=oMI9PjOb9Jl.

Shilong Liu, Tianhe Ren, Jiayu Chen, Zhaoyang Zeng, Hao Zhang, Feng Li, Hongyang Li, Jun Huang, Hang Su, Jun Zhu, and Lei Zhang. Detection transformer with stable matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6491–6500, October 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10012–10022, October 2021.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11976–11986, June 2022b.

Wenyu Lv, Yian Zhao, Qinyao Chang, Kui Huang, Guanzhong Wang, and Yi Liu. RT-DETRv2: Improved baseline with bag-of-freebies for real-time detection transformer, 2024. URL https://arxiv.org/abs/2407.17140.

Yan Ma, Weicong Liang, Bohan Chen, Yiduo Hao, Bojian Hou, Xiangyu Yue, Chao Zhang, and Yuhui Yuan. Revisiting DETR pre-training for object detection, 2023. URL https://arxiv.org/abs/2308.01300.

Zhixiong Nan, Xianghong Li, Jifeng Dai, and Tao Xiang. Mi-detr: An object detection model with multi-time inquiries mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4703–4712, June 2025.

Yansong Peng, Hebei Li, Peixi Wu, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. D-FINE: Redefine regression task of DETRs as fine-grained distribution refinement. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 44015–44031, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/6cf58a87e3097e7d1f9be3e8693a93de-Paper-Conference.pdf.

Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse DETR: Efficient end-to-end object detection with learnable sparsity. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RRGVCN8kjim.

Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, Jitendra Malik, Yanghao Li, and Christoph Feichtenhofer. Hiera: A hierarchical vision transformer without the bells-and-whistles. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 29441–29454. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/ryali23a.html.

Ao Wang, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Repvit: Revisiting mobile cnn from vit perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15909–15920, June 2024.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. PVT v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, Sep 2022. ISSN 2096-0662. doi: 10.1007/s41095-022-0274-8. URL https://doi.org/10.1007/s41095-022-0274-8.

Kan Wu, Jinnian Zhang, Houwen Peng, Mengchen Liu, Bin Xiao, Jianlong Fu, and Lu Yuan. TinyViT: Fast pretraining distillation for small vision transformers. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision – ECCV 2022*, pp. 68–85, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-19803-8. doi: 10.1007/978-3-031-19803-8_5.

Sheng Xu, Yanjing Li, Mingbao Lin, Peng Gao, Guodong Guo, Jinhu Lü, and Baochang Zhang. Q-detr: An efficient low-bit quantized detection transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3842–3851, June 2023.

Jianwei Yang, Chunyuan Li, Xiyang Dai, and Jianfeng Gao. Focal modulation networks. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 4203–4217. Curran Associates, Inc., 2022. doi: 10.52202/068431-0304. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/1b08f585b0171b74d1401a5195e986f1-Paper-Conference.pdf.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3mRwyG5one.

Huaxiang Zhang, Hao Zhang, Kai Liu, Zhongxue Gan, and Guo-Niu Zhu. UAV-DETR: Efficient end-to-end object detection for unmanned aerial vehicle imagery. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 15143–15149. IEEE, October 2025. doi: 10.1109/iros60139.2025.11246176. URL http://dx.doi.org/10.1109/IROS60139.2025.11246176.

Chuyang Zhao, Yifan Sun, Wenhao Wang, Qiang Chen, Errui Ding, Yi Yang, and Jingdong Wang. Ms-detr: Efficient detr training with mixed supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17027–17036, June 2024a.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16965–16974, June 2024b.

Dehua Zheng, Wenhui Dong, Hailin Hu, Xinghao Chen, and Yunhe Wang. Less is more: Focus attention for efficient detr. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6674–6683, October 2023.

Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7380–7399, November 2022. ISSN 1939-3539. doi: 10.1109/tpami.2021.3119563. URL http://dx.doi.org/10.1109/TPAMI.2021.3119563.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=gZ9hCDWe6ke.

Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6748–6758, October 2023.

# A Experimental Protocols and Implementation Details

This appendix documents the training and evaluation settings for the main detector comparisons, ablations, external-dataset experiments, and efficiency measurements.

## A.1 Main COCO Experiments

**Refinement configuration.** The final SCA refinement tail uses three residual depthwise–pointwise blocks with a channel expansion ratio of 2.0. Its input and output projections map between the detector feature width $D$ and the internal width $2D$. Appendix E reports the preliminary capacity study that informed these settings.

**Training settings.** Our SCA configurations build on the RT-DETR Zhao et al. (2024b), D-FINE Peng et al. (2025), and DEIM-D-FINE Huang et al. (2025) detector pipelines, respectively. SCA replaces encoder-side cross-scale fusion while retaining the decoder and prediction heads of each detector family. All four reported runs train on COCO train2017 with total batch size 32, seed 0, AdamW, mixed precision, and exponential moving average (EMA) weights with configured decay $0.9999$. The three Swin-T configurations use pretrained Swin-T weights Fan et al. (2024). The TinyViT configuration uses pretrained TinyViT-21M weights from ImageNet-22K pretraining followed by ImageNet-1K fine-tuning Wu et al. (2022).

For the Swin-T configurations, layer-wise backbone learning rates range from $1.2005 \times 10^{-5}$ to $5 \times 10^{-5}$, with $2.5 \times 10^{-4}$ for the remaining model parameters. Ordinary backbone and remaining weights use weight decay $0.05$ and $10^{-4}$, respectively. The TinyViT configuration uses learning rates of $5 \times 10^{-5}$ for the backbone and $2.5 \times 10^{-4}$ elsewhere, with corresponding ordinary-weight decay $0.05$ and $1.25 \times 10^{-4}$. Designated normalization and bias parameters have zero weight decay. Table S1 summarizes the training schedules; augmentation operators follow the corresponding detector-family protocols.

Table S1: Training schedules for the SCA configurations reported in Table 2.

| SCA configuration | Epochs | Warmup | Learning-rate schedule | Refinement |
|---|---|---|---|---|
| RT-DETR / Swin-T | 72 | Linear; 2,000 steps | Constant | n/a |
| D-FINE / Swin-T | 80 | Linear; 500 steps | Constant | After epoch 72 |
| DEIM-D-FINE / Swin-T | 80 | Quadratic; 500 steps | Constant | After epoch 72 |
| DEIM-D-FINE / TinyViT | 58 | Quadratic; 2,000 steps | Same as DEIM-L Huang et al. (2025) | After epoch 50 |

**Evaluation and checkpoint selection.** We evaluate EMA weights on COCO val2017 with fixed $640 \times 640$ preprocessing and COCO bounding-box metrics. For each SCA configuration, we report the best EMA checkpoint selected by the highest unrounded validation AP over its training run. Metrics in Table 2 are rounded to two decimal places.

## A.2 Ablation Protocols

**Shared 12-epoch setting.** The SCA component ablations in Table 5 and the local-support configurations in Appendix D.2 follow the RT-DETR/Swin-T protocol of Appendix A.1, capped at 12 epochs. Unless noted otherwise, they use seed 0, total batch size 32, and epoch-12 EMA results on COCO val2017 at $640 \times 640$. The centred $4 \times 4$ local-support row reuses the main run's epoch-12 result. The $5 \times 5$ TD/$5 \times 5$ BU configuration instead uses batch size 16 with proportionally adjusted learning rates and update-count warmups; its result therefore characterizes the complete configuration rather than support size alone. Appendix E documents the earlier refinement-capacity protocol separately.

### A.3 External-Dataset Evaluation

**VisDrone2019-DET.** We fine-tune the COCO-trained SCA-DETR with Swin-T on VisDrone2019-DET train for 72 epochs with total batch size 16. We retain the RT-DETR/Swin-T optimizer, augmentation, and learning-rate schedule in Appendix A.1, including the same parameter-group learning rates despite the smaller batch size. We select the checkpoint by AP on VisDrone2019-DET val and evaluate at $640 \times 640$. Our VisDrone-aware evaluator accounts for ignored regions and computes AP over IoU thresholds $0.50$–$0.95$ with at most 500 detections per image.

**PASCAL VOC.** We initialize RT-DETR with ResNet-50 and SCA-DETR with Swin-T from their respective COCO-trained detector checkpoints, then fine-tune on VOC2007 and VOC2012 trainval with total batch size 16. Both use AdamW with a backbone learning rate of $10^{-5}$ and $10^{-4}$ for other parameters, a 2,000-step linear warmup, and constant learning rates thereafter. Both runs were configured for up to 72 epochs and trained through epoch 36. No separate validation split was reserved, so we use the common epoch-36 checkpoint for both models rather than selecting a peak on VOC2007 test. The main table reports COCO-style AP and $AP_{50}$ on VOC2007 test at $640 \times 640$.

### A.4 Efficiency Measurements

We benchmark each detector in its deployed inference configuration on an NVIDIA A100 80GB GPU at $640 \times 640$ resolution and batch size one, using compiled FP32 inference with TensorFloat-32 disabled. Latency is the mean and standard deviation of five trials, each comprising 100 warmup and 200 timed model-forward passes. Timing covers the backbone, encoder, and decoder, excluding compilation, preprocessing, data transfer, and postprocessing. We report pre-deployment parameter counts, estimated compute, and peak PyTorch-allocated GPU memory after warmup.

# B Alignment Diagnostics and Deterministic Cross-Scale Mapping

This appendix specifies the coordinate mapping and measurement protocols for the alignment diagnostics summarized in Section 4.3, and reports their per-stage statistics and supplementary visualizations.

## B.1 Deterministic Adjacent-Level Mapping

The exported Swin-T levels S3, S4 and S5 have input strides 8, 16 and 32, respectively. At level $\ell$, token $(i_\ell, j_\ell)$ has grid centre $(i_\ell + \frac{1}{2}, j_\ell + \frac{1}{2})$, corresponding to input-pixel centre $((i_\ell + \frac{1}{2})s_\ell - \frac{1}{2}, (j_\ell + \frac{1}{2})s_\ell - \frac{1}{2})$. Expressing adjacent levels in this shared coordinate frame gives the continuous source-grid correspondence for target position $p = (i_t, j_t)$ at level $t$ and source level $s$:

$$\phi_{t \leftarrow s}(p) = \left( \frac{(i_t + \frac{1}{2})s_t}{s_s} - \frac{1}{2}, \frac{(j_t + \frac{1}{2})s_t}{s_s} - \frac{1}{2} \right). \tag{7}$$

This correspondence defines the continuous centre of the source neighbourhood in Eq. 1. Appendix D.1 describes its direction-specific discretization, sampling offsets and boundary handling.

## B.2 Conservative Residual-Drift Bound

The deterministic map defines the nominal correspondence; the remaining question is how far the measured response can move around that point. Let $(x^\star, y^\star)$ be the input-referenced location predicted by the mapping and let $(\hat{x}, \hat{y})$ be a measured response location. The residual drift vector and magnitude are

$$\Delta = \begin{bmatrix} \hat{x} - x^\star \\ \hat{y} - y^\star \end{bmatrix}, \qquad \|\Delta\|_2 = \sqrt{(\hat{x} - x^\star)^2 + (\hat{y} - y^\star)^2}. \tag{8}$$

For Swin-T, the main architecture-conditioned source of residual displacement is shifted-window attention. Let $w$ be the window size, $\rho = \lfloor w/2 \rfloor$ the half-window shift in token units, $m_k$ the number of shifted blocks in stage $k$, and $s_k$ the input stride at that stage. A conservative input-referenced $L_\infty$ envelope through exported stage $K$ is

$$E_K^{(\infty)} = \sum_{k \le K} m_k \rho s_k + \sum_{k \le K} q_k^{(\infty)}, \tag{9}$$

where $q_k^{(\infty)}$ collects discretization and parent-assignment uncertainty. The corresponding $L_2$ envelope used for the drift tables is

$$E_K^{(2)} = \sqrt{2}\, E_K^{(\infty)}. \tag{10}$$

Table S2 reports the stage-wise contributions for Swin-T with window size $w = 7$ and shift $\rho = 3$.

Table S2: Swin-T conservative bound expansion in $L_\infty$ input-pixel units. The cumulative exported bound is converted to the stage-wise $L_2$ envelope used in the impulse-response tables.

| Stage | Stride | Depth | Shifted blocks | $\Delta_k^{(\infty)}$ | $q_k^{(\infty)}$ | Cumulative bound |
|---|---|---|---|---|---|---|
| $k = 0$ | 4 | 2 | 1 | 12 | – | – |
| $k = 1$ (S3) | 8 | 2 | 1 | 24 | 2 | 38 |
| $k = 2$ (S4) | 16 | 6 | 3 | 144 | 4 | 186 |
| $k = 3$ (S5) | 32 | 2 | 1 | 96 | 8 | 290 |

The bound is deliberately conservative. It is used to test whether empirical drift remains compatible with accumulated local window shifts and discretization effects, not to claim a tight receptive-field theorem. The resulting $L_2$ envelopes for S3, S4, and S5 are $53.74$, $263.04$, and $410.12$ input pixels.

### B.3 Impulse-Response Locality Protocol

The impulse-response diagnostic probes frozen-backbone geometry without a detector loss. For a neutral input $I$ and an impulse-injected input $I^{(p)}$ at location $p$, the response energy at stage $\ell$ is

$$R_\ell^{(p)}(x, y) = \left\| F_\ell(I^{(p)})_{x,y,:} - F_\ell(I)_{x,y,:} \right\|_2, \tag{11}$$

where $F_\ell$ is the frozen backbone feature map. Each response map is summarized by the peak-response location and by the center of mass (COM) of the response energy. Drift is measured in input pixels relative to the deterministic mapped location. The locked protocol uses COM drift as the primary bound-validation statistic, with peak drift retained as a supporting diagnostic.

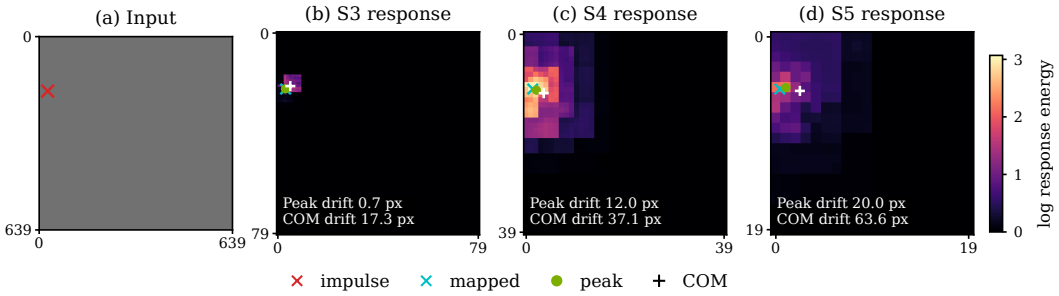


Figure S1: **Illustrative Swin-T impulse-response diagnostic.** A localized input impulse is propagated through the frozen backbone and measured at the exported feature stages. The figure illustrates the mapped-location and residual-displacement quantities used by the impulse-response diagnostic to evaluate local cross-scale support.

Table S3: Impulse-response locality under the locked protocol. Bounds and drift statistics are in input pixels. Swin-T uses the full sampled field; ResNet-50 summaries use an interior subset to reduce padding-boundary effects.

| Backbone | Stage | $n$ | $E^{(2)}$ | Within COM | Median COM | P95 COM |
|---|---|---|---|---|---|---|
| Swin-T | S3 | 101 | 53.74 | 1.000 | 9.40 | 17.82 |
| Swin-T | S4 | 101 | 263.04 | 1.000 | 21.26 | 54.86 |
| Swin-T | S5 | 101 | 410.12 | 1.000 | 28.54 | 92.67 |
| ResNet-50 | S3 | 32 | 2.83 | 1.000 | 1.43 | 1.43 |
| ResNet-50 | S4 | 32 | 8.49 | 1.000 | 5.22 | 5.68 |
| ResNet-50 | S5 | 32 | 19.80 | 0.969 | 8.88 | 16.66 |

**Typical displacement in source-grid units.** To relate the impulse-response measurements to local support, we normalize the median displacement by the corresponding feature stride. The Swin-T medians of 9.40, 21.26 and 28.54 input pixels correspond to approximately 1.18, 1.33 and 0.89 source tokens at S3, S4 and S5, respectively. These values characterize the typical response-centroid displacement across sampled impulse locations and motivate evaluating compact neighbourhoods around the mapped correspondence. The median provides a reference scale for compact attention support: at least half of the sampled impulse locations exhibit response-centroid displacement no greater than this value. This motivates concentrating attention around the typical response location while limiting the cost of broader spatial access. Appendix D.1 interprets this spatial scale in relation to the two sampling directions, and Appendix D.2 evaluates the resulting configurations.

Table S3 reports the per-stage drift statistics, with all sampled Swin-T response centroids falling within the conservative diagnostic envelope. Figure S2 shows the COM-drift distributions on a matched interior subset to reduce boundary effects; its sampling therefore differs from the full-field Swin-T summary in Table S3.

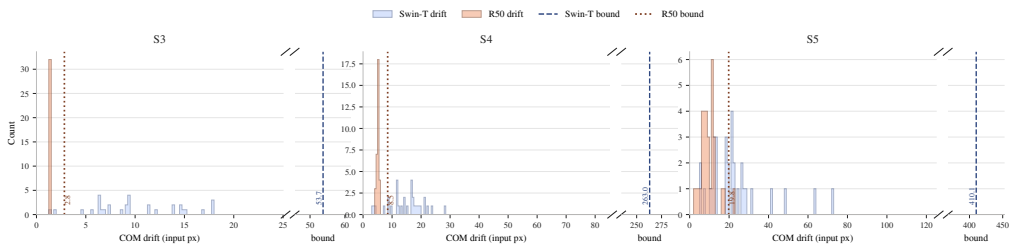


Figure S2: COM-drift histograms under the locked impulse-response protocol. The visualization uses a matched interior subset for Swin-T and ResNet-50 to reduce padding-boundary effects and keep the backbone comparison visually aligned. Broken x-axes expose the observed drift mass and the stage-wise Swin-T bound in the same panel.

### B.4 Natural-Image Consistency After Deterministic Alignment

The second diagnostic asks whether deterministic alignment is sufficient for natural-image feature agreement. Adjacent feature maps are aligned onto a common grid using the deterministic relation above and evaluated on 512 COCO images for the $S4 \rightarrow S3$ and $S5 \rightarrow S4$ pairs. Cosine similarity is the primary representation-consistency metric because it measures directional agreement in feature space after the geometric prior has already been applied. Local NCC is reported as a bounded local-recoverability check, and phase correlation is retained as a global drift diagnostic rather than as a substitute for cosine.

Table S4: Natural-image feature consistency after deterministic alignment. Values are medians over 512 images. Phase shift is the median absolute phase-correlation shift; local NCC is the median mean NCC under bounded local matching.

| Backbone | Pair | Phase shift | Global cos. | Local cos. | Local NCC |
|---|---|---|---|---|---|
| ResNet-50 | S4→S3 | 0.3953 | 0.6950 | 0.6419 | 0.6545 |
| ResNet-50 | S5→S4 | 6.5437 | 0.3662 | 0.3721 | 0.4667 |
| Swin-T | S4→S3 | 7.2638 | -0.0339 | -0.0606 | 0.2724 |
| Swin-T | S5→S4 | 10.2233 | 0.0463 | 0.0221 | 0.3369 |

Table S4 supplements the main-paper cosine results with phase-correlation and local NCC measurements. Under bounded local matching, Swin-T achieves median NCC values of 0.2724 and 0.3369 for $S4 \rightarrow S3$ and $S5 \rightarrow S4$, respectively, below the corresponding ResNet-50 values of 0.6545 and 0.4667. These measurements provide complementary evidence of limited cross-scale feature consistency after deterministic alignment.

# C Computational Cost and Inference Efficiency

This appendix reports the computational cost and inference efficiency of SCA-DETR and its reference baselines. The preliminary refinement-capacity study is reported separately in Appendix E.

We compare RT-DETR with ResNet-50, RT-DETR and SCA-DETR with Swin-T, and the D-FINE and DEIM-D-FINE configurations. We benchmark compiled FP32 inference on an NVIDIA A100 80GB GPU at an input resolution of $640 \times 640$ and a batch size of one. We report steady-state model inference latency and peak allocated GPU memory under the protocol described in Appendix A.4.

Table S5: Computational cost and inference efficiency of the PyTorch implementations. AP values are reproduced from the main COCO val2017 results. Latency is the mean $\pm$ standard deviation across five trials; the RT-DETR/SCA-DETR Swin-T pair is from the Hybrid-first session, while the ResNet-50 row is from the earlier measurement session. Memory is peak PyTorch-allocated GPU memory after warmup.

| Model | Backbone | AP | Params (M) | Estimated GFLOPs | Latency (ms/image) | Peak allocated memory (MiB) |
|---|---|---|---|---|---|---|
| RT-DETR | ResNet-50 | 53.1 | 42.89 | 66.99 | $16.04 \pm 0.01$ | 296.71 |
| RT-DETR | Swin-T | 53.31 | 46.36 | 68.63 | $17.383 \pm 0.028$ | 342.23 |
| SCA-DETR | Swin-T | 54.02 | 44.61 | 67.51 | $\mathbf{14.697 \pm 0.034}$ | 185.69 |
| D-FINE | HGNetv2-L | 54.0 | 31.24 | 45.86 | $15.028 \pm 0.253$ | 198.67 |
| D-FINE | Swin-T | 54.31 | 44.68 | 60.54 | $15.963 \pm 0.332$ | 329.83 |
| SCA-D-FINE | Swin-T | 54.82 | 45.54 | 63.04 | $\mathbf{14.048 \pm 0.008}$ | 188.81 |
| DEIM-D-FINE | HGNetv2-L | 54.7 | 31.29 | 45.97 | $14.789 \pm 0.121$ | 198.71 |
| DEIM-D-FINE | Swin-T | 54.33 | 44.68 | 60.54 | $15.818 \pm 0.075$ | 329.83 |
| SCA-DEIM-D-FINE | Swin-T | 55.08 | 45.54 | 63.04 | $\mathbf{14.046 \pm 0.025}$ | 188.81 |
| DEIM-D-FINE | TinyViT | 55.12 | 37.73 | 58.73 | $17.050 \pm 0.107$ | 266.22 |
| SCA-DEIM-D-FINE | TinyViT | 56.08 | 38.59 | 61.24 | $15.102 \pm 0.003$ | 176.35 |

Relative to RT-DETR with Swin-T, SCA-DETR increases COCO AP from 53.31 to 54.02 while reducing pre-deployment parameters from 46.36M to 44.61M (3.8%) and peak allocated GPU memory from 342.23 to 185.69 MiB (45.7%). Estimated computation is 67.51 versus 68.63 GFLOPs. In the Hybrid-first session, compiled inference latency decreases from 17.383 to 14.697 ms/image (15.45%). Reversing the measurement order gives 17.431 and 14.673 ms/image, respectively (15.82% reduction). The deployed SCA checkpoint retains 54.02 AP at reported precision after conversion, as checked on COCO val2017 with the eager graphs.

The same pattern appears in the D-FINE and DEIM-D-FINE integrations. With Swin-T, SCA-D-FINE raises AP from 54.31 to 54.82 while reducing latency from 15.963 to 14.048 ms/image. SCA-DEIM-D-FINE raises AP from 54.33 to 55.08 and reduces latency from 15.818 to 14.046 ms/image. With TinyViT, SCA-DEIM-D-FINE reaches 56.08 AP versus 55.12 for its baseline, while latency falls from 17.050 to 15.102 ms/image. Peak allocated GPU memory is lower for each of these SCA configurations, as reported in Table S5.

# D LOCAL-SUPPORT GEOMETRY AND CONFIGURATION ABLATION

This appendix explains the local support used in Eq. 1 and evaluates alternative top-down (TD) and bottom-up (BU) configurations, complementing the full-model efficiency comparison in Appendix C.

## D.1 DIRECTION-SPECIFIC SAMPLING GEOMETRY

The diagnostics in Appendix B motivate local cross-scale interaction around the correspondence induced by the feature strides. They support using a local neighbourhood to accommodate residual misalignment, while Appendix D.2 evaluates its extent and alignment. We consider the two paths separately because their source grids differ in spatial resolution.

**Top-down support.** Top-down interaction draws from the coarser S4 and S5 source grids, whose median response-centroid displacements are approximately 1.33 and 0.89 source tokens (Appendix B.3). Under the twofold stride relation, adjacent fine-grid indices $2m$ and $2m + 1$ map to coarse-grid positions $m - \frac{1}{4}$ and $m + \frac{1}{4}$, respectively. Both have the same nearest coarse token $m$, which is selected by floor anchoring. Consequently, each $2 \times 2$ group of fine targets shares a coarse anchor, with its mapped correspondences displaced by only a quarter-token along each axis (Figure S3a). We use offsets $\{-1, 0, 1\}$ along both axes to provide nine surrounding source samples. This combines a small anchoring displacement with compact contextual access; the configuration study in Appendix D.2 evaluates the benefit of expanding this neighbourhood.

**Bottom-up support.** Bottom-up interaction draws from the finer S3 and S4 source grids, whose median displacements are approximately 1.18 and 1.33 tokens. Here, a coarse target maps halfway between four fine-grid tokens. Floor anchoring selects one of these tokens, placing an odd window's centre half a token away from the continuous correspondence along each axis. Instead, offsets $\{-1, 0, 1, 2\}$ produce a $4 \times 4$ window whose samples lie at $\{-1.5, -0.5, 0.5, 1.5\}$ relative to the mapped centre along each axis. This gives symmetric support with sixteen sampling slots and a spatial extent consistent with the empirical median displacement scale. The floor-aligned $5 \times 5$ alternative uses twenty-five slots while remaining centred on the integer anchor (Figure S3b). The results in Appendix D.2 support retaining the centred $4 \times 4$ configuration through comparable overall accuracy, stronger small-object performance, and lower estimated compute and peak inference memory than this larger BU alternative.

At image boundaries, source coordinates are clamped to the valid grid, retaining repeated sampling slots. The diagnostic displacements motivate candidate neighbourhoods; the detection ablation evaluates the resulting support policies.

## D.2 LOCAL-SUPPORT CONFIGURATION ABLATION

We compare four TD/BU support configurations using the ablation protocol and configuration-specific training settings in Appendix A.2.

Table S6: Local-support configuration ablation. TD and BU denote top-down and bottom-up window dimensions. Floor and centred refer to the BU sampling geometry. All rows use the epoch-12 EMA readout.

| TD support | BU support | BU alignment | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|---|---|
| $3 \times 3$ | $3 \times 3$ | Floor | 48.19 | 65.51 | 52.14 | 29.06 | 52.86 | 66.47 |
| $3 \times 3$ | $4 \times 4$ | Centred | 48.59 | 66.17 | 52.40 | 29.94 | 52.98 | 66.75 |
| $3 \times 3$ | $5 \times 5$ | Floor | 48.60 | 66.06 | 52.63 | 29.50 | 53.01 | 66.70 |
| $5 \times 5$ | $5 \times 5$ | Floor | 48.83 | 66.36 | 52.80 | 29.28 | 53.25 | 67.01 |

**Support extent and detection accuracy.** With $3 \times 3$ top-down support, expanding floor-aligned bottom-up support from $3 \times 3$ to $5 \times 5$ increases AP from 48.19 to 48.60. The centred $4 \times 4$ alternative achieves comparable overall accuracy (48.59 AP) and the highest small-object accuracy among the evaluated configurations (29.94 $AP_S$), while reducing bottom-up sampling slots from

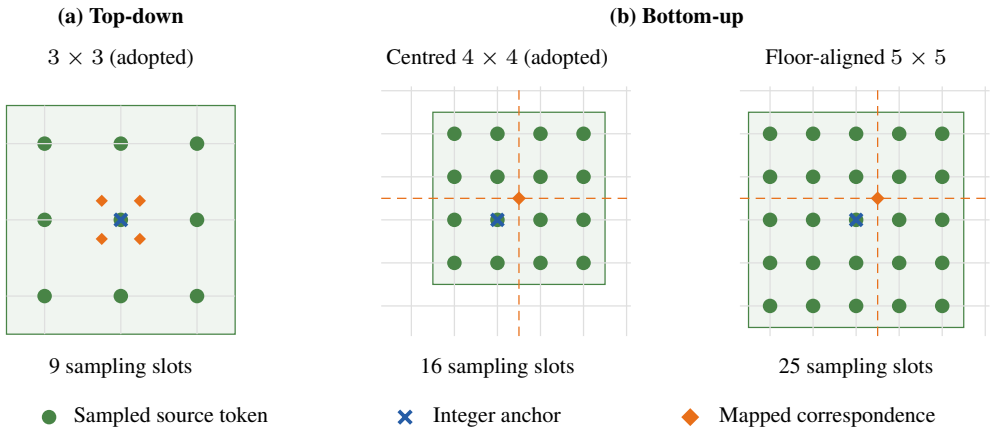


Figure S3: **Direction-specific local-support geometry.** (a) Four fine-grid targets map to positions $(\pm\frac{1}{4}, \pm\frac{1}{4})$ around a shared coarse anchor, supporting floor-anchored $3 \times 3$ top-down sampling. (b) The bottom-up correspondence lies at $(\frac{1}{2}, \frac{1}{2})$ relative to the integer anchor. The adopted $4\times4$ window is centred on this correspondence; the floor-aligned $5 \times 5$ window is centred on the anchor. Green dots denote sampled source tokens, blue crosses integer anchors, and orange diamonds mapped correspondences. Coordinates are in the respective source-grid units; all windows are shown before boundary clamping.

25 to 16. This comparison evaluates the complete support policy, including its extent, alignment, boundary sampling, and learned offset biases.

**Inference cost.** Table S7 reports inference costs under the protocol in Appendix A.4. Across the evaluated configurations, estimated arithmetic cost, peak allocated memory, and full-model latency vary. The $3 \times 3/4 \times 4$ and $5 \times 5/5 \times 5$ configurations report 67.509 and 68.317 GFLOPs, 14.697 and 17.925 ms/image, and 185.69 and 329.29 MiB of peak allocated memory, respectively. The larger configuration achieves the highest overall AP (48.83), although its small-object AP is lower (29.28).

Table S7: Computational cost of the local-support configurations under the protocol in Appendix A.4. Latency is the mean $\pm$ standard deviation across five trials; memory is peak allocated GPU memory after warmup.

| TD support | BU support | BU alignment | Estimated GFLOPs | Latency (ms/image) | Peak allocated memory (MiB) |
|---|---|---|---|---|---|
| $3 \times 3$ | $3 \times 3$ | Floor | 68.234 | $17.678 \pm 0.033$ | 324.20 |
| $3 \times 3$ | $4 \times 4$ | Centred | 67.509 | $14.697 \pm 0.034$ | 185.69 |
| $3 \times 3$ | $5 \times 5$ | Floor | 68.251 | $17.581 \pm 0.037$ | 324.96 |
| $5 \times 5$ | $5 \times 5$ | Floor | 68.317 | $17.925 \pm 0.032$ | 329.29 |

**Selected configuration.** We retain $3 \times 3$ top-down and centred $4 \times 4$ bottom-up support for its balance of overall accuracy, small-object detection, and inference cost. Its bottom-up window is symmetric about the continuous correspondence described in Appendix D.1, and its measured accuracy is comparable to the larger floor-aligned alternative. With this configuration, the full-model comparison in Appendix C reports higher AP than RT-DETR with Swin-T, fewer parameters, and lower peak inference memory, with lower measured model-forward latency.

# E PRELIMINARY REFINEMENT CAPACITY STUDY

We examine how the depth and channel expansion of the residual refinement tail affect detection accuracy and model size. This preliminary study uses an earlier Swin-T SCA-DETR configuration with ImageNet initialization, floor-anchored $3 \times 3$ top-down and $5 \times 5$ bottom-up supports, and no learned per-offset attention bias. Its training and evaluation settings are specified below. The comparisons support the three-block, expansion-2.0 refinement setting retained in the final model.

**Study setup.** Each refinement tail comprises input and output channel projections and a stack of residual depthwise–pointwise blocks, with an outer residual connection around the tail. We vary the number of blocks and the internal channel expansion ratio while retaining the remaining architecture. All five configurations were trained for 12 epochs with seed 0, a total batch size of 16, and mixed precision on a single NVIDIA H100 GPU. Table S8 reports COCO val2017 accuracy at $640 \times 640$ from the final epoch, which also attained the best reported AP for each configuration. Parameter counts refer to the complete detector.

Table S8: Preliminary refinement-capacity study with Swin-T on COCO val2017. The retained depth and expansion setting is highlighted. Efficiency measurements follow Appendix A.4; latency is the mean $\pm$ standard deviation across five trials, and memory is peak allocated GPU memory.

| Blocks | Expansion | Params (M) | AP | $AP_{50}$ | $AP_{75}$ | Estimated GFLOPs | Latency (ms/image) | Peak allocated memory (MiB) |
|---|---|---|---|---|---|---|---|---|
| 2 | 2.0 | 43.54 | 45.89 | 63.07 | 49.62 | 65.57 | $20.61 \pm 0.16$ | 310.07 |
| **3** | **2.0** | 44.61 | 46.43 | 63.60 | 50.16 | 68.26 | $21.19 \pm 0.29$ | 314.21 |
| 3 | 3.0 | 49.11 | 46.44 | 63.73 | 50.15 | 79.51 | $23.05 \pm 0.06$ | 331.43 |
| 3 | 4.0 | 55.18 | 46.71 | 64.01 | 50.49 | 94.69 | $25.39 \pm 0.09$ | 381.83 |
| 4 | 2.0 | 45.69 | 46.43 | 63.83 | 50.07 | 70.95 | $21.89 \pm 0.14$ | 318.35 |

**Depth and expansion tradeoffs.** At expansion 2.0, increasing refinement depth from two to three blocks improves AP by 0.54 for an additional 1.08M parameters. A fourth block adds another 1.08M parameters while leaving overall AP unchanged at the reported precision. At depth three, increasing expansion from 2.0 to 3.0 adds 4.50M parameters for a 0.01 AP gain. Expansion 4.0 achieves the highest AP, 46.71, at a cost of 10.57M additional parameters relative to expansion 2.0. The three-block, expansion-2.0 setting therefore offers a compact capacity–accuracy compromise within this preliminary study.

**Inference cost.** We evaluate the preliminary architectures using the compiled FP32, batch-one protocol in Appendix A.4, with $640 \times 640$ inputs. All five capacity configurations were benchmarked within a common measurement session on a separate A100 80GB PCIe GPU from that used for Appendices C and D, following the same inference protocol. The results therefore characterize relative capacity tradeoffs within this study; cross-appendix latency differences combine architectural and measurement-session differences.

Increasing depth from two to three blocks raises estimated computation from 65.57 to 68.26 GFLOPs and latency from 20.61 to 21.19 ms/image, accompanying the 0.54 AP gain. A fourth block increases latency to 21.89 ms/image without improving overall AP at the reported precision. At depth three, expansion 4.0 raises computation to 94.69 GFLOPs, latency to 25.39 ms/image, and peak allocated memory from 314.21 to 381.83 MiB, for a 0.28 AP improvement over expansion 2.0. The retained three-block, expansion-2.0 setting balances the observed accuracy gains with the additional inference cost of deeper or wider refinement.

# F Additional Backbone Comparisons

We evaluate SCA across five hierarchical backbone families in the RT-DETR setting on COCO val2017. Each backbone comparison follows the shared 12-epoch training and evaluation setting in Appendix A.2, with backbone-specific initialization. For each backbone, Table S9 compares RT-DETR with its standard HybridEncoder against SCA-DETR, which replaces the encoder-side fusion module. This additional comparison complements the detector-pipeline and backbone results in Table 2.

Table S9: Additional hierarchical-backbone transfer screen on COCO val2017. RT-DETR rows use the standard HybridEncoder with the same backbone family, while SCA-DETR replaces the encoder fusion with Structured Cross-Scale Attention.

| | RT-DETR | | SCA-DETR | | |
|---|---|---|---|---|---|
| Backbone | AP | $AP_{50}$ | AP | $AP_{50}$ | $\Delta$AP |
| Swin-T Liu et al. (2021) | 47.5 | 65.1 | 48.6 | 66.2 | +1.1 |
| TinyViT Wu et al. (2022) | 48.7 | 65.3 | 49.9 | 66.5 | +1.2 |
| PVTv2-B2 Wang et al. (2022) | 49.1 | 66.6 | 49.5 | 66.9 | +0.4 |
| NAT-Tiny Hassani et al. (2023) | 48.6 | 66.3 | 49.1 | 66.8 | +0.5 |
| Hiera-T (SAM2) Ryali et al. (2023) | 44.3 | 60.1 | 44.9 | 60.8 | +0.6 |

SCA improves AP across all five backbones. The largest gains occur with TinyViT ($+1.2$ AP) and Swin-T ($+1.1$ AP), while PVTv2-B2, NAT-Tiny, and Hiera-T improve by $0.4$–$0.6$ AP. $AP_{50}$ also increases for every backbone. These results extend the evidence for structured cross-scale fusion beyond the Swin-T and TinyViT configurations emphasized in the main comparisons.