ADAPTERS DO MORE THAN ADAPT: COUPLING PARAMETER-EFFICIENT FINE-TUNING WITH RECOVERY-AWARE STRUCTURED PRUNING FOR VISUAL GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual grounding aims to localize an image region specified by a natural-language expression. Large pretrained vision-language models have substantially improved grounding accuracy but incur considerable adaptation and inference costs. Parameter-efficient fine-tuning reduces trainable parameters while leaving the deployed backbone intact, whereas structured pruning produces smaller models but is typically decoupled from task adaptation, leading to representation shifts and degraded task-relevant responses. To bridge this gap, we propose Adapter-Conditioned Recovery Pruning (ACRP), which couples parameter-efficient adaptation with recovery-aware structured pruning in a unified optimization framework. The key idea is to counterfactually supervise adapters to approximate compact signatures of candidate-removed responses, thereby providing recovery-aware guidance for pruning. ACRP combines modality-aware joint budget allocation across attention and MLP structures with a three-stage optimization strategy, allowing adaptation and compression to interact before physical pruning. Moreover, parallel multi-level feature branches integrate hierarchical visual semantics, while an evidence-preserving token condenser retains expression-relevant tokens and aggregates the remaining tokens into regional context. Experiments on four public datasets demonstrate that ACRP achieves effective task adaptation and retains competitive localization performance, while updating merely 0.38% of the pre-trained encoder parameters and physically pruning approximately 40% of the backbone parameters.

Keywords: Visual Grounding, Vision-Language Models, Parameter-Efficient Fine-Tuning, Recovery-Aware Structured Pruning, Adapter-Based Recovery

1 INTRODUCTION

Visual grounding aims to localize the image region referred to by a free-form natural-language expression (Deng et al., 2023; 2021). Unlike object detection over a predefined category space, visual grounding requires a model not only to recognize the appearance and attributes of the target but also to reason about spatial relations and distinguish among multiple visually similar candidates (Liu et al., 2025). This capability is fundamental to embodied intelligence, human-robot collaboration, and general-purpose vision-language understanding (Shridhar et al., 2020; Kamath et al., 2021). In recent years, Transformer-based methods (Xiao et al., 2024b; Zhu et al., 2023; Shi et al., 2024; Yao et al., 2025; Liu et al., 2024b;c) have substantially advanced visual grounding through fine-grained cross-modal interaction and end-to-end localization. Meanwhile, large-scale pretrained models such as CLIP (Radford et al., 2021), Grounding DINO (Liu et al., 2024a), and DINOv3 (Siméoni et al., 2026) have provided increasingly transferable linguistic representations and dense visual features. However, as pretrained encoders continue to grow in scale, balancing downstream adaptation accuracy with computational efficiency has become an increasingly important challenge for visual grounding.

Full fine-tuning requires updating and storing all backbone parameters, incurring substantial optimization memory, training cost, and task-specific storage overhead. Parameter-efficient fine-tuning (PEFT)

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.

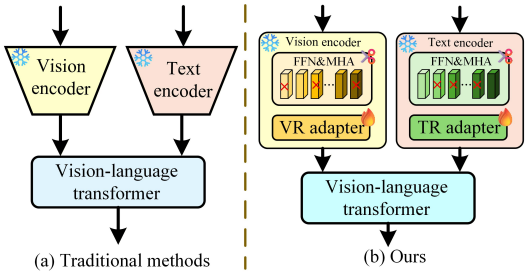


Figure 1: The comparison of visual grounding frameworks.

alleviates these costs by freezing the pretrained backbone and optimizing only a small number of parameters (Li et al., 2026a; Lin et al., 2026; Dong et al., 2026). However, parameter efficiency during training does not necessarily translate into efficiency at deployment. At inference time, the frozen backbone must still be stored and executed in full, leaving its parameter count, computational cost, and activation memory largely unchanged. Parallel adapter branches may even introduce additional inference overhead. Structured pruning (Zhang & Yang, 2026; Wang et al., 2026; Berisha et al., 2025) offers a complementary solution by physically removing redundant structures, such as attention heads and MLP channels, thereby producing a genuinely smaller model. Nevertheless, existing pruning methods (Shi et al., 2023; Shen et al., 2024) are typically developed independently of downstream adaptation, with importance estimated using task-agnostic, unimodal, or post-hoc criteria.

Recent studies have investigated joint structured pruning and PEFT, cross-modal sparsity allocation, and representation-preserving compensation (Zhang & Yang, 2026; Shi et al., 2023; Shen et al., 2024; Zhao et al., 2024). However, these components remain weakly coupled: adapters are not explicitly trained to recover candidate pruned responses, while pruning decisions rarely exploit adapter recoverability. Consequently, post-hoc pruning or a shared training schedule may still cause substantial representation shifts. Existing allocation strategies also overlook the different roles and redundancy patterns of attention and MLP structures across modalities. This is particularly problematic for visual grounding, where visual and textual encoders must preserve distinct spatial and semantic cues (Deng et al., 2021). We therefore ask: Can adapters do more than adapt and enable safer structured pruning of pretrained vision-language models?

To address these challenges, we propose ACRP, a visual grounding framework that bidirectionally couples parameter-efficient adaptation with recovery-aware structured pruning in a unified training pipeline, as illustrated in Figure 1. ACRP employs frozen DINOv3 and CLIP encoders augmented with lightweight adapters in their Transformer layers. Beyond learning task-specific residuals, these adapters are counterfactually supervised to approximate compact signatures of candidate-removed responses. The coupling is bidirectional: candidate removed responses provide recovery supervision for the adapters, while adapter recovery errors guide global budget allocation and the selection of attention heads and MLP channels before physical pruning. ACRP is optimized in three stages: adaptation warm-up, joint compression exploration, and post-pruning recovery. These stages respectively establish task-specific representations, coordinate adapter recovery with structural selection, and restore the performance of the physically compressed model. The main contributions of this work are summarized as follows:

(1) We propose ACRP, a visual grounding framework that couples parameter-efficient adaptation with recovery-aware structured pruning within a unified optimization process. ACRP enables adapters to serve the dual roles of task adaptation and pruning recovery, thereby balancing parameter efficiency during training with structural efficiency at deployment.

(2) We design parallel multi-level feature branches and an evidence-preserving token condenser to fuse multi-level visual semantics, retain tokens relevant to the referring expression, and aggregate the remaining tokens into regional context, thereby reducing redundancy while preserving the background and relational cues required for localization.

(3) Experiments on four public benchmarks show that ACRP maintains competitive grounding performance while introducing trainable parameters amounting to only approximately 0.38% of the pretrained encoders' total parameter count and physically removing approximately 40% of the backbone parameters.

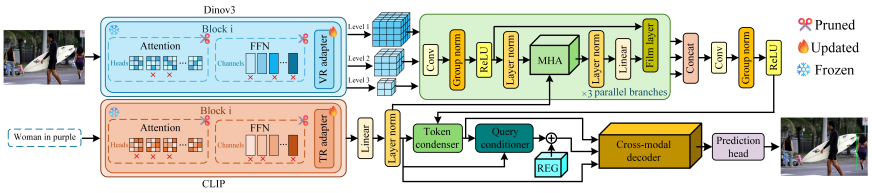


Figure 2: Overview of the ACRP framework.

2 Related Work

2.1 Visual grounding

Visual grounding aims to localize the image region referred to by a natural-language expression and requires joint reasoning over the target's appearance and attributes, spatial relations, and contextual dependencies. Early methods (Yu et al., 2018; Yang et al., 2019) typically followed proposal-based pipelines. Yu et al. decomposed a referring expression into subject, location, and relation cues and matched each of them against candidate regions. Subsequent Transformer-based methods (Wang et al., 2025a; Li et al., 2026b; Zhu et al., 2026; Wang et al., 2025b; Kang et al., 2025; Xiao et al., 2024a; Huang & Satoh, 2024) simplify this pipeline through end-to-end cross-modal modeling. Deng et al. (Deng et al., 2021) formulated visual grounding as direct bounding-box regression. Kamath et al. (Kamath et al., 2021) performed text-conditioned object detection. More recently, large-scale pretrained grounding models have leveraged extensive image-text supervision to improve transferability and open-vocabulary localization. Despite their strong performance, these accuracy-oriented methods typically rely on large pretrained encoders, dense multi-level visual features, and extensive cross-modal interactions, thereby incurring substantial adaptation and inference costs. Moreover, processing all visual tokens indiscriminately introduces considerable redundancy, whereas directly discarding low-scoring tokens risks removing the background and relational evidence required for accurate grounding. In contrast, our method couples parameter-efficient adaptation with recovery-aware structured pruning and employs parallel multi-level feature branches and an evidence-preserving token condenser to reduce token redundancy while retaining expression-relevant visual and contextual evidence.

2.2 Parameter-Efficient Adaptation and Compression of Vision-Language Models

Parameter-efficient fine-tuning (PEFT) adapts large pretrained models by freezing their backbones and optimizing a small number of task-specific parameters (Li et al., 2026a; Lin et al., 2026; Dong et al., 2026). Common approaches include adapters, low-rank updates, and learnable prompts. In visual grounding, Liu et al. (Liu et al., 2024b) introduced domain-aware and relation-aware adapters. Liu et al. (Liu et al., 2024c) incorporated multimodal priors and local convolutional adapters. Although these methods reduce training and storage costs, the complete backbone must still be executed during inference.

Structured pruning physically removes redundant attention heads and MLP channels while token compression reduces computation by shortening visual sequences (Shi et al., 2023; Fang et al., 2023; Rao et al., 2021). However, these techniques are generally optimized independently of PEFT, and directly removing structures or low-saliency tokens may discard task-relevant responses and relational evidence. ACRP jointly optimizes parameter-efficient adaptation and recovery-aware compression. Its adapters support both task adaptation and pruned-response recovery, while the token condenser retains expression-relevant tokens and aggregates the remaining tokens into regional context. This design connects training-time parameter efficiency with deployment-time structural efficiency.

3 Methods

3.1 Framework Overview

Given an image I and a referring expression Q, ACRP predicts the normalized bounding box $\widehat{b}$ of the referred object. As shown in Figure 2, ACRP combines frozen DINOv3 and CLIP encoders with

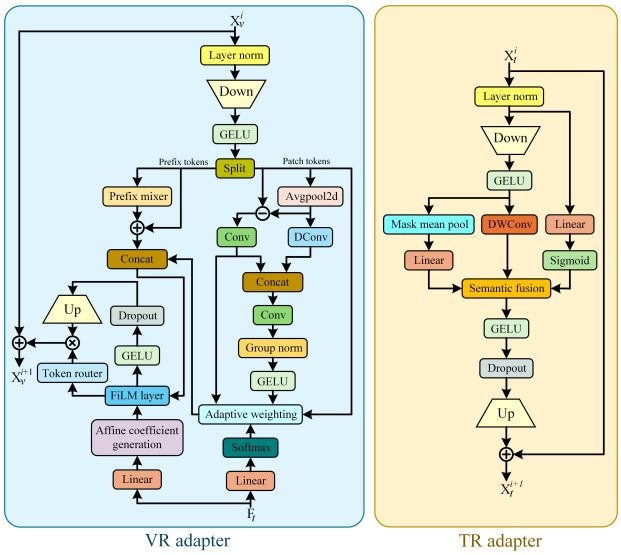


Figure 3: The architecture of the proposed VR and TR adapters.

lightweight recovery adapters, multi-level grounding, token condensation, and structured pruning. The textual and multi-level visual representations are extracted as:

$$\left(T, \bar{t}\right) = \widetilde{E}_t(Q) \tag{1}$$

$$\{V_1, V_2, V_3\} = \widetilde{E}_v\left(I \mid \bar{t}\right) \tag{2}$$

where T and $\bar{t}$ are the contextual and global text representations, $\widetilde{E}_t$ is the CLIP encoder augmented with textual recovery adapters, $\widetilde{E}_v$ is the DINOv3 encoder augmented with text-conditioned visual recovery adapters, V_1, V_2, and V_3 are the multi-level visual representations extracted from the shallow, intermediate, and deep layers of the DINOv3 encoder. The visual features are fused and condensed as:

$$V_c = \mathcal{C}\left(\mathcal{F}\left(V_1, V_2, V_3, T\right), \bar{t}\right) \tag{3}$$

where $\mathcal{F}$ denotes multi-level language-conditioned fusion, and $\mathcal{C}$ is the evidence-preserving token condenser. A learned REG token q_{reg} is then initialized and progressively decoded:

$$q_0 = q_{reg} + \mathcal{Q}\left(\left[\bar{t}; \overline{v}_c\right]\right) \tag{4}$$

$$q_{l+1} = \mathcal{D}_l\left(q_l, T, V_c\right) \tag{5}$$

$$\widehat{b} = \mathcal{H}\left(q_L\right) \tag{6}$$

where q_0 is the initial decoder query, $\mathcal{Q}$ is the query-conditioning network, $\bar{t}$ is the concatenated global text representation, $\overline{v}_c$ is the mean-pooled condensed visual representation, $\mathcal{D}_l$ is the cross-modal decoder, V_c is the condensed visual tokens, $\mathcal{H}$ is the bounding box prediction head, q_L is the final REG query representation obtained after all L decoding layers. ACRP is optimized through adaptation warm-up, joint compression exploration, and post-pruning recovery, allowing adaptation, structural selection, and response recovery to interact before physical pruning.

3.2 Adapter-Conditioned Joint Adaptation and Pruning

Let $m \in \{v, t\}$ denote the modality and X_m^i the input to Transformer block i. The frozen backbone response and adapter residual are combined as:

$$X_m^{i+1} = X_m^i + \mathcal{B}_m^i\left(X_m^i; g_{m,a}^i; g_{m,f}^i\right) + \Delta X_m^i \tag{7}$$

$$\Delta X_m^i = \mathcal{A}_m^i\left(X_m^i; c_m\right) \tag{8}$$

where $\mathcal{B}_m^i$ denotes the gated attention and MLP responses, $g_{m,a}^i$ and $g_{m,f}^i$ are their soft gates, $\mathcal{A}_m^i$ is the recovery adapter, and c_m is its modality-specific condition. As shown in Figure 3, the visual recovery (VR) adapter combines regional, boundary, FiLM-modulated, and token-routed features. The text recovery (TR) adapter fuses local phrase patterns with global sentence semantics. Their up-projections are zero-initialized.

During joint exploration, complementary gates construct candidate removed responses:

$$S^{-}_{m,i} = \Psi\left(R^{-,a}_{m,i}\right) + \Psi\left(R^{-,f}_{m,i}\right) \tag{9}$$

$$S^{ada}_{m,i} = \Psi\left(\Delta X^{i}_{m}\right) \tag{10}$$

where $S^{-}_{m,i}$ is the target signature of the candidate-removed structures, $S^{ada}_{m,i}$ is the corresponding adapter-generated signature used in $\mathcal{L}_{rec}$, Ψ extracts a compact signature consisting of a global mean token and several anchor tokens, $R^{-,a}_{m,i}$ and $R^{-,f}_{m,i}$ are the candidate-removed responses of the attention and FFN sublayers. Recovery is supervised by:

$$\mathcal{L}_{rec} = E\big[w\big(0.7\left[1-\cos\big(S^{ada}_{m,i}, S^{-}_{m,i}\big)\right] + 0.3\ \mathrm{SmoothL1}\big(\log\big(1+\|S^{ada}_{m,i}\|_2\big), \log\big(1+\|S^{-}_{m,i}\|_2\big)\big)\big)\big]. \tag{11}$$

where w is normalized from the removed-response magnitude. The two terms constrain response direction and magnitude. The retention importance and joint objective are defined as:

$$I_u = \left[\lambda_T \mathcal{R}\left(T_u\right) + \lambda_D \mathcal{R}\left(D_u\right) + \lambda_G \mathcal{R}\left(G_u\right)\right]\left[1 - \lambda_A \mathcal{R}\left(A_u\right)\right] \tag{12}$$

$$\mathcal{L}_j = \mathcal{L}_g + \lambda_b \mathcal{L}_b + \lambda_r \mathcal{L}_{rec} + \lambda_p \mathcal{L}_p \tag{13}$$

where T_u, D_u, G_u, and A_u denote parameter transport cost, task contribution, Taylor sensitivity, and adapter recoverability, $\mathcal{R}$ is within-layer rank normalization, a larger I_u indicates a higher retention priority, $\mathcal{L}_g$, $\mathcal{L}_b$, and $\mathcal{L}_p$ are the grounding, budget, and gate-polarization losses, while the λ terms are their weights. The global allocator determines the final structure using I_u, parameter cost, and recovery confidence.

3.3 Multi-Level Grounding and Evidence-Preserving Token Condensation

ACRP processes shallow, intermediate, and deep DINOv3 features using parallel language-conditioned branches:

$$U_l = \mathcal{P}_l\left(V_l\right) \tag{14}$$

$$T' = LN\left(W_t T\right) \tag{15}$$

$$C_l = MHA_l(LN\left(U_l\right), T', T') \tag{16}$$

$$[\gamma_l, \beta_l] = W^f_l C_l \tag{17}$$

$$\widetilde{U}_l = 2\sigma\left(\gamma_l\right) \odot U_l + \beta_l \tag{18}$$

$$V_f = \mathcal{P}_f\left(\left[\widetilde{U}_1 \parallel \widetilde{U}_2 \parallel \widetilde{U}_3\right]\right) \tag{19}$$

where $\mathcal{P}_l$ and $\mathcal{P}_f$ are the level and fusion projections, M_t is the text mask, C_l is the location-specific linguistic context, γ_l and β_l are the FiLM parameters. After flattening V_f into N tokens, their routing scores combine semantic similarity and a text-gated visual score:

$$g_t = \sigma\left(W_g LN\left(\bar{t}\right)\right) \tag{20}$$

$$[\omega_1, \omega_2] = Softmax(\xi) \tag{21}$$

$$s_n = \omega_1 \mathcal{Z}\left(\cos\left(LN\left(v_n\right), LN\left(\bar{t}\right)\right)\right) + \omega_2 \mathcal{Z}\left(f_s\left(v_n \odot (\ 1 + g_t)\right)\right) \tag{22}$$

where v_n and s_n are token n and its routing score, g_t is the text-conditioned channel gate, f_s is a learned scorer, $\mathcal{Z}$ denotes per-sample standardization, ω_1 and ω_2 are learned mixing weights. The highest-scoring tokens are retained, while the remaining tokens are pooled into regional context:

$$K = \max\left(K_{\min}, \lfloor \rho N \rfloor\right) \tag{23}$$

$$\delta_K = TopK\left(\{s_n\}^{N}_{n=1}, K\right) \tag{24}$$

$$c_r = \sum_{n \in \Omega^{-}_r} Softmax_{n \in \Omega^{-}_r}\left(s_n/\tau\right) v_n + e_c \tag{25}$$

$$V_c = \left[\{v_n\}_{n \in \delta_K} \parallel \{c_r\}^{R}_{r=1}\right] \tag{26}$$

where ρ is the keep ratio, $K_{\min}$ is the minimum retained-token count, and δ_K contains the selected indices. Ω^{-}_r denotes the unselected tokens in region r, τ is the pooling temperature, e_c is a context-type embedding, and c_r is the resulting context token. Positional embeddings are aggregated using the same weights.

3.4 Physical Recovery and Three-Stage Optimization

Before physical pruning, ACRP estimates removed responses on a small calibration subset. Building on CORP (Zhang & Yang, 2026), we extend its closed-form ridge recovery to removed attention-head

outputs. For attention layer i, let $\mathcal{K}_i$ and $\mathcal{R}_i$ denote the retained and removed heads:

$$A_i = \left(C_{\mathcal{K}_i\mathcal{K}_i} + \lambda I\right)^{-1} C_{\mathcal{K}_i\mathcal{R}_i} \tag{27}$$

$$\widehat{H}_{\mathcal{R}_i} = A_i^{\mathsf{T}} \left(H_{\mathcal{K}_i} - \mu_{\mathcal{K}_i}\right) + \mu_{\mathcal{R}_i} \tag{28}$$

where $C_{\mathcal{K}_i\mathcal{K}_i}$ and $C_{\mathcal{K}_i\mathcal{R}_i}$ are the retained-head covariance and retained-to-removed cross-covariance, λ is the regularization coefficient, A_i is the recovery mapping, and $\widehat{H}_{\mathcal{R}_i}$ is the estimated removed response. The estimate is folded into the attention output projection. For each MLP, the calibrated mean of removed channels is folded into the bias, while retained weights are corrected using the pre- and post-pruning variances. These operations introduce no additional inference branch. Counterfactual supervision and calibration folding thus provide staged, rather than explicitly orthogonal, compensation, with $\mathcal{L}_{rec}$ disabled after pruning. Training comprises adaptation warm-up, joint compression exploration, and post-pruning recovery. These stages first establish task-specific representations, then jointly optimize adapters and structural gates, and finally recover the physically pruned model.

4 Experiments

4.1 Experiment Setup

Datasets and Evaluation Metrics. We evaluate ACRP on four public visual grounding benchmarks: RefCOCO (Yu et al., 2016), RefCOCO+ (Yu et al., 2016), RefCOCOg (Mao et al., 2016), and ReferItGame (Kazemzadeh et al., 2014), following their standard training and evaluation splits. Following the standard visual grounding protocol (Deng et al., 2021), we adopt Intersection over Union (IoU) as the primary evaluation metric, where a prediction is considered correct if its IoU with the ground-truth box exceeds 0.5.

Implementation Details. We employ a DINOv3 ViT-L visual encoder and a CLIP-L text encoder, inserting a recovery-aware adapter with a reduction ratio of 64 into every transformer layer. The pretrained backbone weights remain frozen, while only the adapters, structural gates, and task-specific modules are optimized. Images are resized while preserving their aspect ratios and padded to 518×518, and the training epoch is set to 200. The initial target keep ratios for the visual/text MLPs are $0.50/0.25$, while those for the visual/text attention heads are $0.78/0.55$. The final layer-wise structures are determined by the global budget allocator. Further optimization, pruning, and reproducibility details are provided in Appendix A.

4.2 Comparison With the State-of-the-Art Visual Grounding Methods

To demonstrate the effectiveness of the proposed method, we compare ACRP with representative methods, with the experimental results shown in Table 1. ACRP achieves the best reported results among the listed encoder-based methods, although these published results involve different backbone configurations. On RefCOCO testA and testB, ACRP obtains 90.70% and 85.20%, outperforming the strongest previous results by 0.53 and 1.93 percentage points, respectively. The larger improvement on testB indicates stronger localization of non-person objects under complex contextual conditions. On RefCOCO+, ACRP achieves 85.54%/70.30% on testA/testB, surpassing the strongest compared method by 1.61/0.47 points. It also obtains 80.85% on RefCOCOg test-u and 75.31% on ReferItGame, outperforming MAP and HiVG by 1.53 and 0.09 points, respectively. Across five common splits, ACRP improves over MaPPER by 1.80–5.04 points. These results are achieved by updating only 0.38% of pretrained parameters while pruning about 40% of the backbone, demonstrating a favorable accuracy–efficiency trade-off.

4.3 Comparison With the State-of-the-Art PEFT Methods

As shown in Table 2, we compare ACRP with mainstream PEFT methods on RefCOCO. MAP, DARA, and MaPPER use published results, while PEFT-BoA, RMAdapter, and HI^{2}MA are reimplemented under the same backbone, grounding head, training schedule, and input resolution as ACRP. The update ratio denotes the number of trainable parameters introduced for adapting the pretrained encoders relative to their original parameter count. It can be observed that ACRP achieves 90.70% and 85.20% on testA and testB, attaining the best results among the compared methods. It outperforms the strongest compared method by 0.63 percentage points on testA and by 1.71 points on testB, while

Table 1: Comparison with state-of-the-art Visual Grounding methods.

Methods	RefCOCO		RefCOCO+		RefCOCOg	ReferItGame
	testA	testB	testA	testB	test-u	test
MAP (Yao et al., 2025)	89.35	82.01	82.55	66.42	79.32	74.95
DARA (Liu et al., 2024b)	82.76	76.72	69.83	57.22	67.67	–
MaPPER (Liu et al., 2024c)	88.90	81.19	81.12	65.68	75.81	–
FSVG (Wang et al., 2025a)	87.40	80.06	80.64	64.01	73.15	71.93
TLCN (Li et al., 2026b)	85.68	81.36	78.96	63.99	73.45	–
DCART (Zhu et al., 2026)	86.72	81.87	78.51	64.84	72.27	
RefDetector (Wang et al., 2025b)	85.01	76.98	74.21	61.07	71.73	–
AttBalance (Kang et al., 2025)	88.13	81.50	80.25	66.34	75.61	–
HiVG (Xiao et al., 2024a)	89.86	83.27	83.81	68.11	78.79	75.22
LoA-Trans (Huang & Satoh, 2024)	90.17	82.98	83.93	69.83	79.29	–
ACRP (Ours)	**90.70**	**85.20**	**85.54**	**70.30**	**80.85**	**75.31**

Table 2: Comparison with state-of-the-art PEFT methods. Update ratio represents the proportion of learnable parameters introduced by PEFT in the pre-trained encoder relative to the total number of parameters.

Methods	RefCOCO		Update ratio (%)
	testA	testB	
MAP (Yao et al., 2025)	89.35	82.01	0.44
DARA (Liu et al., 2024b)	82.76	76.72	1.63
MaPPER (Liu et al., 2024c)	88.90	81.19	1.41
PEFT-BoA (Li et al., 2026a)	89.61	83.16	0.78
HI^{2}MA (Dong et al., 2026)	90.07	83.14	0.69
RMAdapter (Lin et al., 2026)	90.03	83.49	0.52
ACRP (Ours)	**90.70**	**85.20**	**0.38**

using the lowest adaptation ratio of 0.38%. Compared with MAP, which has the closest parameter budget, ACRP reduces the update ratio from 0.44% to 0.38% while improving testA and testB by 1.35 and 3.19 points, respectively.

These results indicate that ACRP improves grounding performance through more effective use of a compact adaptation parameter budget rather than by introducing additional trainable parameters. By assigning the adapters the dual roles of task adaptation and pruning recovery, ACRP connects parameter-efficient training with deployment-oriented compression.

4.4 Comparison With the State-of-the-Art Pruning Methods

To evaluate the benefit of coupling adaptation with pruning, we conduct comparisons with standalone pruning methods, as summarized in Table 3. For a fair comparison, all pruning methods are matched in final parameter count, calibration and recovery data, recovery epochs, and hyperparameter-tuning budget. ACRP achieves 90.70% and 85.20% on RefCOCO testA and testB, substantially outperforming the other pruning methods. Compared with the strongest baseline, VBP, ACRP improves testA and testB by 4.31 and 8.46 percentage points, respectively. Because all methods use the same pruning ratio, these gains cannot be attributed to a looser compression target and instead demonstrate the advantages of ACRP in task-aware structure selection and pruned-response recovery. The larger improvement on testB further suggests that ACRP better preserves the contextual and relational evidence required for grounding non-person objects.

4.5 Ablation Study

Ablation Study of Adaptation and Pruning Components. Table 4 examines the contribution of each major component. The base model A1 obtains 86.25%/78.92% on testA/testB. Introducing the VR Adapter alone (A2) improves the results by 4.20/5.38 percentage points, while the TR Adapter alone (A3) yields gains of 3.89/4.02 points. Combining both Adapters (A4) further increases performance to 91.53%/85.73%, demonstrating their complementary roles in visual and textual adaptation. Adding

Table 3: Comparison of grounding accuracy with representative pruning methods.

Methods	RefCOCO testA	RefCOCO testB
CORP (Zhang & Yang, 2026)	63.73	55.21
GISP (Wang et al., 2026)	54.17	44.10
VBP (Berisha et al., 2025)	86.39	76.74
ACRP (Ours)	**90.70**	**85.20**

Table 4: Component-wise ablation of adaptation, token condensation, and structured pruning.

Method	VR adapter	TR adapter	Token condenser	Structure pruning	Joint allocation	RefCOCO testA	RefCOCO testB
A1	×	×	×	×	×	86.25	78.92
A2	✓	×	×	×	×	90.45	84.30
A3	×	✓	×	×	×	90.14	82.94
A4	✓	✓	×	×	×	91.53	85.73
A5	✓	✓	✓	×	×	91.71	85.89
A6	✓	✓	✓	✓	×	90.26	84.40
ACRP (Ours)	✓	✓	✓	✓	✓	**90.70**	**85.20**

the token condenser (A5) produces a further improvement to 91.71%/85.89%. Applying structured pruning without joint budget allocation (A6) reduces performance to 90.26%/84.40%, whereas joint allocation recovers 0.44/0.80 points. Compared with the unpruned A5 variant, full ACRP sacrifices only 1.01/0.69 points while physically removing approximately 40% of the backbone parameters. Additional ablations on adapter capacity and token-condenser design are provided in Appendix B.

Figure 4 compares the attention maps of A1, A4, and ACRP. A1 exhibits diffuse attention, whereas the adapters in A4 improve focus on the referred targets. Despite approximately 40% physical pruning, ACRP preserves similar attention patterns and remains robust in distant or crowded scenes. Together with Table 4, these results indicate that recovery-aware pruning largely retains task-relevant adapter responses.

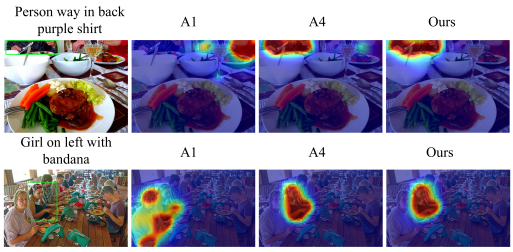


Figure 4: Attention visualization of the baseline (A1), the dual-adapter variant (A4), and the physically pruned ACRP.

Effect of the Pruning Ratio. Table 5 presents the accuracy–efficiency trade-off across pruning ratios. Increasing the ratio from 20% to 40% reduces testA/testB accuracy by only 0.25/0.17 points while decreasing parameters, storage, GFLOPs, peak memory, and latency by 25.5%, 21.9%, 24.3%, 21.6%, and 7.3%. Because general-purpose GPU kernels underutilize reduced widths and adapters and routing add fixed overhead, ACRP achieves greater savings in model size and memory than in wall-clock latency. At 60% pruning, accuracy drops by 6.42/9.22 points relative to 40%. We therefore adopt 40% as the default balance between accuracy and efficiency. Further analysis of the resource footprint and deployment efficiency is provided in Appendix C.

Table 5: Effect of the physical pruning ratio on grounding accuracy and inference efficiency.

Pruning ratio	RefCOCO testA	RefCOCO testB	Total params	Storage	GFLOPs	Peak memory	Inference time
0.60	84.28	75.98	181.94M	454MB	299.44G	442.94MiB	66.21ms
0.40	90.70	85.20	258.83M	601MB	458.44G	594.62MiB	69.51ms
0.20	90.95	85.37	347.40M	770MB	605.42G	758.46MiB	75.00ms

Ablation Study of Pruning Criteria and Recovery. Table 6 evaluates the pruning criterion and recovery objective. Taylor sensitivity or weight magnitude alone yields 88.05%/81.71% and 88.49%/81.12%. Removing the recoverability-aware score reduces testA/testB accuracy by 0.95/1.22 points. With calibration-based folding held fixed, replacing counterfactual adapter supervision with the standard task objective reduces accuracy from 90.70%/85.20% to 89.99%/83.77%, demonstrating its incremental benefit without assuming that the two recovery mechanisms are orthogonal.

Table 6: Ablation of pruning criteria and counterfactual adapter supervision.

Architecture	RefCOCO testA	RefCOCO testB
Ours w/ taylor-based importance only	88.05	81.71
Ours w/ magnitude-based importance only	88.49	81.12
Ours w/o recoverability-aware score	89.75	83.98
Ours w/ task loss only	89.99	83.77
ACRP (Ours)	**90.70**	**85.20**

4.6 Qualitative Results

Figure 5 presents qualitative results of ACRP, with green boxes indicating predictions. Across expressions involving attributes, scale, spatial relations, and similar instances, ACRP accurately localizes targets under small object sizes and cluttered backgrounds. These results show that ACRP preserves fine-grained visual and relational evidence after structural and token compression.

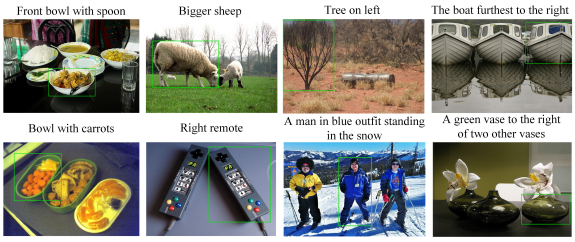


Figure 5: The qualitative results of ACRP.

5 Conclusion

This work bridges parameter-efficient adaptation and deployment-oriented structural efficiency in visual grounding. We propose ACRP, a unified framework that couples adaptation with recovery-aware structured pruning. Its visual and textual adapters support both task learning and pruning recovery, while multi-level feature branches and an evidence-preserving token condenser reduce structural and token redundancy without losing essential grounding cues. A three-stage strategy coordinates adaptation and pruning, with calibration corrections folded into existing parameters. Experiments on four benchmarks show that ACRP maintains competitive performance while updating only 0.38% of pretrained parameters and physically removing about 40% of backbone parameters. These results show that recovery-supervised adapters can facilitate effective structured pruning.

6 AI USE STATEMENT

Generative AI tools were used to draft and revise portions of the manuscript, refine language and organization, assist with code refactoring, and improve the textual presentation of the methods and results. They were not used to generate experimental data or quantitative results. All AI-assisted text and code were manually reviewed and verified by the authors, who take full responsibility for the final manuscript.

REFERENCES

U Berisha, J Mehnert, and A Condurache. Variance-Based Pruning for Accelerating and Compressing Trained Networks, 2025. Proceedings of the IEEE/CVF International Conference on Computer Vision, 4973-4982.

J Deng, Z Yang, T Chen, et al. TransVG: End-to-End Visual Grounding with Transformers, 2021. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 1769-1779.

J Deng, Z Yang, D Liu, et al. TransVG++: End-to-end visual grounding with language conditioned vision transformer, 2023. IEEE Trans. Pattern Anal. Mach. Intell, 45(11):13636-13652.

C Dong, T Zhang, F Zhang, et al. Hierarchical intra-inter modal adaptation for vision-language models, 2026. Pattern Recognition, 180:114200.

G Fang, X Ma, M Song, et al. DepGraph: Towards Any Structural Pruning, 2023. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 16091-16101.

Z Huang and S Satoh. LoA-Trans: Enhancing Visual Grounding by Location-Aware Transformers, 2024. ECCV, 15065.

A Kamath, M Singh, Y LeCun, et al. MDETR - Modulated Detection for End-to-End Multi-Modal Understanding, 2021. International Conference on Computer Vision, 1760-1770.

W Kang, L Zhou, J Wu, et al. Visual Grounding with Attention-Driven Constraint Balancing, 2025. Proceedings of the 33rd ACM International Conference on Multimedia, 1637-1645.

S Kazemzadeh, V Ordonez, M Matten, et al. ReferitGame: Referring to objects in photographs of natural scenes, 2014. in Proc. Conf. Empirical Methods Natural Lang. Process, 787-798.

H Li, G Liu, X Wang, et al. PEFT-BoA: Parameter-Efficient Fine-Tuning with Bag-of-Adapters for Multi-Modal Object Re-identification, 2026a. Proceedings of the AAAI Conference on Artificial Intelligence, 40(8):6127-6135.

Z Li, C Wang, Y Pang, et al. Task-aware cross-modal refinement and liquid fusion for text-visual grounding, 2026b. Front. Artif. Intell, 9:1798247.

X Lin, W Li, S Guo, et al. RMAdapter: Reconstruction-based Multi-Modal Adapter for Vision-Language Models, 2026. AAAI.

S Liu, Z Zeng, T Ren, et al. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection, 2024a. European Conference on Computer Vision, 15105:38-55.

T Liu, X Liu, S Huang, et al. DARA: Domain- and Relation-Aware Adapters Make Parameter-Efficient Tuning for Visual Grounding, 2024b. International Conference on Multimedia and Expo, 1-6.

T Liu, Z Xu, Y Hu, et al. MaPPER: Multimodal Prior-guided Parameter Efficient Tuning for Referring Expression Comprehension, 2024c. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 4984-4994.

Y Liu, J Wang, L Xiao, et al. Foregroundness-Aware Task Disentanglement and Self-Paced Curriculum Learning for Domain Adaptive Object Detection, 2025. IEEE Transactions on Neural Networks and Learning Systems, 36(1):369-380.

J Mao, J Huang, A Toshev, et al. Generation and comprehension of unambiguous object descriptions, 2016. in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, 11-20.

A Radford, J Kim, C Hallacy, et al. Learning Transferable Visual Models from Natural Language Supervision, 2021. ICML, 8748-8763.

Y Rao, W Zhao, B Liu, et al. DynamicViT: Efficient Vision Transformers with Dynamic Token Sparsification, 2021. Advances in Neural Information Processing Systems.

B Shen, Z Lin, D Zha, et al. Pruning Large Language Models to Intra-module Low-rank Architecture with Transitional Activations, 2024. In Findings of the Association for Computational Linguistics: ACL, 9781-9793.

D Shi, C Tao, Y Jin, et al. UPop: Unified and Progressive Pruning for Compressing Vision-Language Transformers, 2023. Proceedings of the 40th International Conference on Machine Learning, 202:31292-31311.

F Shi, R Gao, W Huang, et al. Dynamic MDETR: A Dynamic Multimodal Transformer Decoder for Visual Grounding, 2024. in IEEE Transactions on Pattern Analysis and Machine Intelligence, 46(2):1181-1198.

M Shridhar, J Thomason, D Gordon, et al. ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks, 2020. Conference on Computer Vision and Pattern Recognition, 10737-10746.

O Siméoni, H Vo, M Seitzer, et al. DINOv3, 2026. Transactions on Machine Learning Research.

J Wang, W Zhang, D Huang, et al. A Simple and Better Baseline for Visual Grounding, 2025a. IEEE International Conference on Multimedia and Expo, 1-6.

Y Wang, Z Tian, Z Qin, et al. RefDetector: A Simple Yet Effective Matching-based Method for Referring Expression Comprehension, 2025b. Proceedings of the AAAI Conference on Artificial Intelligence, 39(8):8033-8041.

Z Wang, E Diao, Q Le, et al. From Local to Global: Revisiting Structured Pruning Paradigms for Large Language Models, 2026. In Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics, 1:35720-35739.

L Xiao, X Yang, F Peng, et al. HiVG: Hierarchical Multimodal Fine-grained Modulation for Visual Grounding, 2024a. In Proceedings of the 32nd ACM International Conference on Multimedia, 5460-5469.

L Xiao, X Yang, F Peng, et al. CLIP-VG: Self-paced Curriculum Adapting of CLIP for Visual Grounding, 2024b. IEEE Transactions on Multimedia, 26:4334-4347.

S Yang, G Li, and Y Yu. Cross-Modal Relationship Inference for Grounding Referring Expressions, 2019. Conference on Computer Vision and Pattern Recognition, 4140-4149.

R Yao, Y Rong, T Zou, et al. MAP: Parameter-Efficient Tuning for Referring Expression Comprehension via Multi-Modal Adaptive Positional Encoding, 2025. In Proceedings of the 33rd ACM International Conference on Multimedia, 2264-2273.

L Yu, P Poirson, S Yang, et al. Modeling context in referring expressions, 2016. in Proc. Eur. Conf. Comput. Vis, 69-85.

L Yu, Z Lin, X Shen, et al. MAttNet: Modular attention network for referring expression comprehension, 2018. in Proc. IEEE Conf. Comput. Vis. Pattern Recognit, 1307-1315.

B Zhang and B Yang. CORP: Closed-Form One-shot Representation-Preserving Structured Pruning for Vision Transformers, 2026. , ArXiv, abs/2602.05243.

B Zhao, H Hajishirzi, and Q Cao. APT: Adaptive Pruning and Tuning Pretrained Language Models for Efficient Training and Inference, 2024. Proceedings of the 41st International Conference on Machine Learning, 235:60812-60831.

H Zhu, Q Lu, L Xue, et al. Visual Grounding with Joint Multi-modal Representation and Interaction, 2023. IEEE Transactions on Instrumentation and Measurement, 72:1-11.

Y Zhu, D Chen, H Wang, et al. DCART: A dual contrastive alignment residual transformer model for visual grounding, 2026. Pattern Recognition, 172:112688.

A Experimental and Reproducibility Details

A.1 Optimization Details

All models are trained for 200 epochs using AdamW, BF16 mixed precision, and cosine learning-rate decay. Before physical pruning, the learning rates for the adapters, structural gates, stabilization parameters, and task-specific modules are set to 1×10^{-4}, 2×10^{-3}, 2×10^{-5}, and 2×10^{-4}, respectively. After pruning, the structural gates are removed, and the learning rates for the adapters, stabilization parameters, and task-specific modules are set to 6×10^{-5}, 1×10^{-5}, and 8×10^{-5}, respectively.

A.2 Pruning Schedule and Calibration

The initial target keep ratios of the visual and textual MLP channels are set to 0.50 and 0.25, respectively, while those of the visual and textual attention heads are set to 0.78 and 0.55. These values specify the initial modality-dependent compression targets rather than fixed layer-wise pruning ratios. The final retained structures are determined by the global budget allocator according to task importance, structural cost, and adapter recoverability.

The first 20 epochs are used for adaptation warm-up, during which the structural gates remain inactive. The compression objectives are gradually introduced over the subsequent 20 epochs. Physical pruning is performed at epoch 110 using 64 calibration mini-batches with a batch size of 8. After calibration, the selected attention heads and MLP channels are physically removed, the soft gates are discarded, and the optimizer is rebuilt. Post-pruning recovery fine-tuning is then conducted for the remaining epochs.

A.3 Parameter Accounting and Reproducibility

The PEFT update ratio is defined as the number of trainable parameters introduced by the recovery adapters into the pretrained encoders divided by the original total parameter count of the visual and textual encoders. Structural gates and task-specific grounding modules are excluded from this ratio. Therefore, the reported 0.38% represents the adaptation overhead introduced by PEFT rather than the proportion of all trainable parameters in the complete model.

The physical pruning ratio is calculated from the backbone parameter counts before and after the removal of attention heads and MLP channels. Unless otherwise stated, the total parameter counts reported in Table 5 include the pruned encoders, recovery adapters, token condenser, cross-modal decoder, and prediction head. All experiments are implemented with Python 3.10 and PyTorch 2.6.0 using CUDA 12.4. Training is conducted on eight NVIDIA GeForce RTX 4090 GPUs, with the random seed fixed at 42.

B Additional Ablations

Effect of Adapter Capacity. Table 7 compares different adapter reduction ratios. A ratio of 32 achieves the highest accuracy but requires an update ratio of 0.68%. Increasing the reduction ratio to 64 lowers the update ratio to 0.38%, a relative reduction of approximately 44.1%, while decreasing testA/testB by only 0.44/0.23 points. A more restrictive ratio of 128 reduces the adaptation overhead further but degrades performance to 89.96%/84.16%. These results support 64 as a balanced choice between adaptation capacity and parameter efficiency.

Table 7: Effect of the adapter reduction ratio on accuracy and adaptation overhead.

Reduction ratio	RefCOCO		Update ratio (%)
	testA	testB	
32	91.14	85.43	0.68
64	90.70	85.20	0.38
128	89.96	84.16	0.24

Ablation Study of the Evidence-Preserving Token Condenser. As shown in Table 8, the complete token condenser achieves 90.70%/85.20%, outperforming dense-token processing by 0.71/1.05 points. This result suggests that removing token redundancy can suppress irrelevant visual information in addition to reducing computation. Compared with naïve top-k selection and the variant without

context pooling, the complete design improves testB by 2.04 and 1.45 points, respectively. Thus, directly discarding low-scoring tokens can remove useful background and relational cues, whereas aggregating unselected tokens into regional context better preserves grounding evidence.

Table 8: Ablation of the evidence-preserving token condenser.

Architecture	RefCOCO	
	testA	testB
Ours w/ dense token	89.99	84.15
Ours w/ naïve top-k	90.49	83.16
Ours w/o context pooling	89.92	83.75
ACRP (Ours)	**90.70**	**85.20**

C Efficiency and Deployment Analysis

Figure 6 presents the normalized resource footprint of ACRP before and after physical pruning. Pruning consistently reduces the total parameter count, model storage, computational cost, peak inference memory, and inference latency. Together with the results in Table 3, this comparison shows that ACRP more effectively preserves grounding accuracy under a fixed pruning budget while delivering practical deployment gains. This favorable trade-off is enabled by jointly optimizing the adapters and structural gates and explicitly recovering task-relevant responses during physical pruning, thereby mitigating the representation shift caused by structural removal.

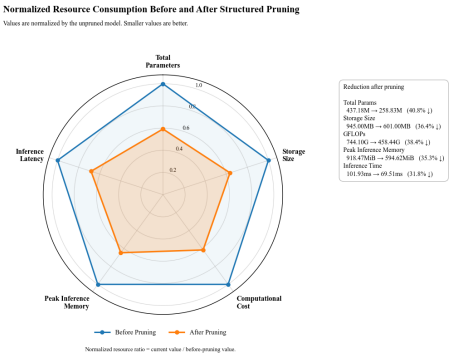


Figure 6: Normalized resource footprint of ACRP before and after physical pruning.