MIXTURE OF PRIMITIVES: FINE-GRAINED EXPERT SHARING FOR GENERATIVE VISUAL PERCEPTION IN UNIFIED MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unified multimodal models (UMMs) increasingly support dense visual perception through image generation, an emerging setting that we refer to as Generative Visual Perception (GVP). While GVP tasks share a common generative interface, they require distinct visual transformations to produce task-specific outputs, motivating task-dependent specialization while preserving shared representations. Mixture-of-experts (MoE) provides a natural mechanism for introducing this conditional specialization through routed experts, but conventional MoE parameterization couples routing decisions with expert-level parameter ownership, requiring each expert to maintain an independent copy of its full transformation. Such coupling restricts parameter ownership to the expert level. We propose Mixture of Primitives (MoP), a fine-grained expert-sharing mechanism that decouples routing identity from parameter ownership by selectively sharing internal channel-group primitives across experts, enabling fine-grained parameter sharing while preserving route-specific transformations. We mainly validate MoP on Cheers, a pre-trained UMM, forming Vision-Cheers, which achieves competitive performance with task-specific models while retaining image generation and multimodal understanding capabilities, and further validate MoP on Bagel for cross-architecture generality.

1 INTRODUCTION

Unified multimodal models (UMMs) combine visual understanding and image generation within a single framework (Chameleon Team, 2024; Xu et al., 2024; Xie et al., 2025; Diao et al., 2026; Zhang et al., 2026). Their shared generative interface also supports dense perception through generated visual targets (Han et al., 2026; Gabeur et al., 2026). We refer to this setting as Generative Visual Perception (GVP), where heterogeneous dense prediction tasks are unified through a common generative interface (Wang et al., 2023a; Xiao et al., 2024; Lin et al., 2025; Han et al., 2026).

Although GVP exposes heterogeneous dense prediction tasks through a common generative interface, adapting a single shared transformation to all of them forces different supervision signals to update the same parameters. Prior work on multi-task learning shows that such tasks can exhibit conflicting optimization signals and benefit from task-dependent specialization (Yu et al., 2020; Fifty et al., 2021; Chen et al., 2023b). This motivates introducing sparse routed experts on top of the shared UMM pathway to obtain specialized residual transformations while retaining the common backbone (Wu et al., 2022). Yet specialization need not imply fully disjoint internal computation. Prior multi-task vision studies have shown that related visual tasks exhibit nontrivial transfer relationships and often benefit from selectively sharing intermediate representations or network components (Misra et al., 2016; Zamir et al., 2018; Liu et al., 2019; Sun et al., 2020). This suggests that some internal computations may be reusable across task-dependent routes. Conventional MoEs nevertheless bind routing identity to parameter ownership: selecting a route also selects an entirely expert. As a result, potentially reusable transformations remain isolated across routes, while sharing entire experts would unnecessarily couple transformations that require route-specific adaptation. Figure 1 illustrates the granularity mismatch between expert-level routing with disjoint parameter ownership and the primitive-level sharing structure inherent in generative visual perception tasks,

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.

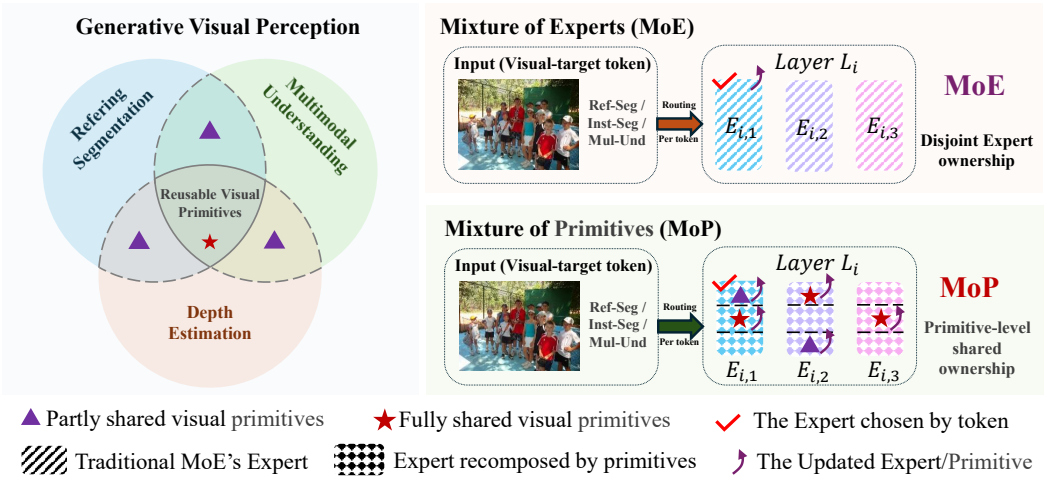

Figure 1: **From disjoint expert ownership to primitive-level shared ownership.** Left: conceptual motivation for reusing transformations across GVP tasks. Top right: conventional MoE uses per-token Top-1 routing with disjoint residual experts. Bottom right: MoP shares experts' parameter groups across routes. Expert routing identities and activated residual width remain unchanged.

and motivates our central question: *How should routed experts share internal parameters when routing identity and parameter ownership need not coincide?*

Our answer is to move the unit of sharing from whole experts to finer-grained internal components. We call these components primitives: fine-grained channel groups that provide the unit of parameter ownership and can be selectively shared across routes. MoP identifies corresponding primitives across experts and selectively shares their parameters according to their compatibility.

Built on pretrained UMMs, we scale Cheers (Zhang et al., 2026), a relatively compact UMM, with MoP to support dense GVP, native image generation, and multimodal understanding within a single checkpoint. We further validate MoP on Bagel (Deng et al., 2025) to demonstrate cross-architecture generalization. The resulting models are evaluated across referring segmentation, monocular depth estimation, image generation, and multimodal understanding, demonstrating broad capability without task-specific model variants. Our contributions are:

- We propose Mixture of Primitives (MoP), a fine-grained expert-sharing method that decouples routing from parameter ownership and improves GVP with fewer residual parameters.
- We apply MoP to Cheers, yielding Vision-Cheers, which achieves competitive dense perception while preserving generation and multimodal understanding. We further validate MoP on Bagel, demonstrating cross-architecture generalization.
- We construct a 38.27M multimodal training stream that unifies dense perception, image generation, and multimodal understanding, enabling the development of unified multimodal models with both generative and perceptual capabilities.

2 Related Work

2.1 Unified Multimodal Models and Generative Perception

Painter, SegGPT, and InstructCV formulate heterogeneous perception tasks as conditional image generation (Wang et al., 2023a;b; Gan et al., 2024), while Vision Banana and SenseNova-Vision extend this interface to structured, dense, and 3D outputs (Gabeur et al., 2026; Han et al., 2026). Unified multimodal models further combine visual understanding and generation within shared architectures and objectives (Chameleon Team, 2024; Wang et al., 2026; Xie et al., 2025; Zhou et al., 2025; Wu et al., 2025a; Liang et al., 2025; Diao et al., 2026). These works establish a shared generative interface for heterogeneous visual tasks, but do not address how conditional expert parameters

should be shared once route-specific specialization is introduced. Built on Cheers (Zhang et al., 2026), Vision-Cheers focuses on this parameter-sharing problem within the shared GVP pathway.

2.2 Routed Experts and Parameter Sharing

MoEs route tokens among expert transformations (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022), while multi-task experts and adapter mixtures combine shared computation with routed adaptation (Liang et al., 2022; Chen et al., 2023a;b; Pfeiffer et al., 2021; Wang et al., 2022; Shen et al., 2024). Prior work also explores sharing across experts: DeepSeekMoE introduces always-active shared experts, Residual MoE models experts as deviations from shared computation (Dai et al., 2024; Wu et al., 2022), MC-SMoE aligns and merges sparse experts (Li et al., 2024), and PCoMoE composes finer-grained projection paths at inference (Gan et al., 2026). In contrast, MoP preserves expert-level routing and active computation, and instead changes parameter ownership within each routed expert, allowing selected internal primitives to be shared while others remain route-specific.

3 Method

3.1 Problem Definition of Generative Visual Perception

Unlike conventional perception models that directly predict task-specific outputs, generative visual perception (GVP) represents each prediction target as an image and solves heterogeneous perception tasks through a shared visual-generation pathway (Gabeur et al., 2026; Han et al., 2026). For task a, a renderer ρ_a converts the annotation y_a into a visual target $x_a^{\mathrm{tar}} = \rho_a(y_a)$. Given an input image I and task instruction u_a, the unified multimodal model G_θ, parameterized by θ, generates the corresponding visual target,

$$\hat{x}_a^{\mathrm{tar}} = G_\theta(I, u_a), \qquad \hat{y}_a = \delta_a(\hat{x}_a^{\mathrm{tar}}), \tag{1}$$

where δ_a maps the generated visual target back to the task prediction. Thus, tasks share the same generative interface while differing in target rendering and decoding. Appendix H specifies the task-specific rendering and evaluation protocols like referring segmentation and depth estimation.

3.2 Overview and Two-stage Optimization

Adapting a UMM to heterogeneous GVP tasks may benefit from specialized visual transformations, as different tasks can induce conflicting optimization signals when sharing the same pathway. Meanwhile, these tasks operate on related visual representations and may benefit from reusable computations.

As shown in Figure 2, we build a residual-expert framework with Mixture of Primitives (MoP) to balance expert specialization and parameter sharing. Stage 1 learns specialized residual experts with disjoint parameter ownership, while Stage 2 reorganizes their ownership at the primitive level through MoP.

3.3 Stage 1: Residual Expert Specialization

3.3.1 Routed Expert Construction

At each routed layer, the pretrained FFN supplies shared computation, while a set of residual experts provides route-dependent responses. Let $\ell \in \mathcal{L}_r$ index routed layers, i index tokens, and $k \in [K] = \{1, \ldots, K\}$ index experts. We write $h_i^\ell = \mathrm{RMSNorm}(s_i^\ell)$ for the normalized token state and $m_i \in \{0, 1\}$ for the visual prediction target. The pretrained FFN is denoted by F_ℓ, and Δ_i^ℓ denotes the added residual response. Appendix D gives the complete token update. Each residual expert $E_{\ell,k}$'s gate/up parameters are randomly initialized. These parameters are independently owned across experts in Stage 1. For each token, the router produces probabilities p_i^ℓ, selects $z_i^\ell = \arg\max_k p_{i,k}^\ell$, and yields the one-hot vector $q_{i,k}^\ell = \mathbf{1}[k = z_i^\ell]$.

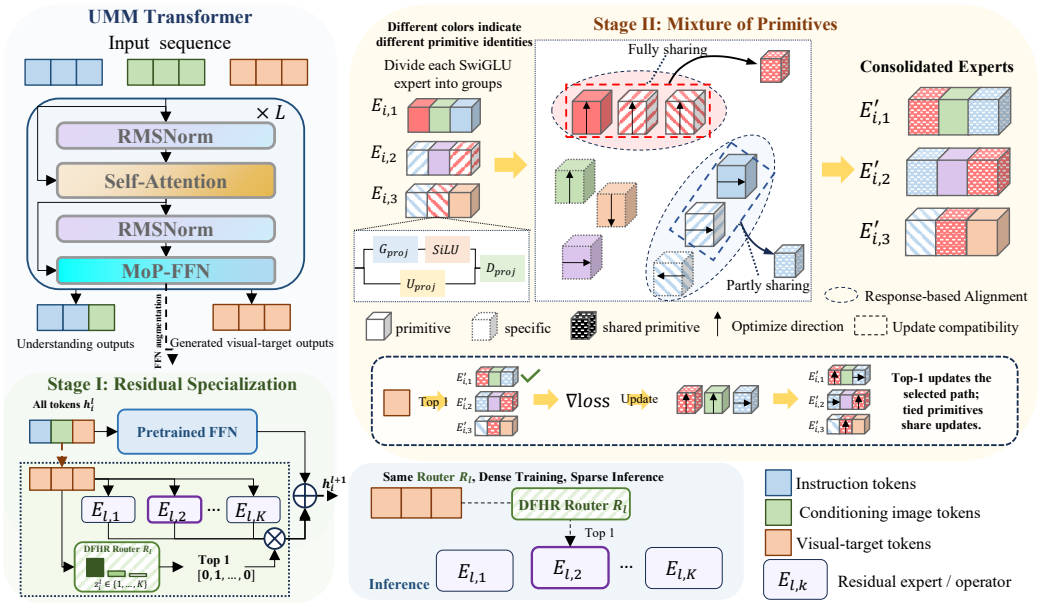

Figure 2: **Overview of our two-stage residual-expert framework with Mixture of Primitives (MoP).** Residual experts are inserted alongside the pretrained FFNs at routed layers of the UMM. In Stage I, sparse routing learns specialized residual experts with disjoint parameter ownership. In Stage II, MoP takes these specialized experts as input and reorganizes their ownership at the primitive level by aligning corresponding primitives and evaluating their update compatibility. Compatible primitives are consolidated into shared components, while incompatible ones remain route-specific. The resulting experts retain the same router and are used for sparse inference.

3.3.2 Expert Specialization

Stage 1 jointly trains the disjoint residual experts and their router using dense-feedback hard routing (DFHR), inspired by (Bengio et al., 2013). For candidate response $e^{\ell}_{i,k} = E_{\ell,k}(h^{\ell}_i)$, an output-side straight-through estimator gives

$$\hat{q}^{\ell}_i = q^{\ell}_i + p^{\ell}_i - \mathrm{sg}(p^{\ell}_i), \qquad \Delta^{\ell}_i = m_i \sum_{k=1}^{K} \hat{q}^{\ell}_{i,k} e^{\ell}_{i,k}, \tag{2}$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient. All K candidate responses inform the router during training, while the forward value and expert task gradients follow the selected path. The visual-target indicator restricts the residual response to visual tokens. Appendix D derives the router gradient and defines the routing controls.

Task supervision and router regularization allow the independently parameterized experts to specialize under routing. The resulting checkpoint is the source for MoP construction: its expert responses support group alignment, and its gate/up update signatures guide ownership induction. The specialization experts also serve as frozen teachers during Stage-2 recovery. Initialization details of residual experts are provided in Appendix D.

3.4 Stage 2: Mixture of Primitives

3.4.1 Primitive Definition and Alignment

Conceptually, a primitive is the basic functional unit over which MoP reasons about parameter ownership. We instantiate it as a fixed-width group along the intermediate dimension of each SwiGLU residual expert, consisting of the corresponding gate/up rows and down-projection columns. We use $K = 3$ experts and $G = 4$ primitives per expert, yielding 1,120 channels per primitive (Appendix I).

Because the residual experts are specialized independently, primitive indices are not inherently aligned across experts: functionally corresponding primitives may appear at different group positions. We therefore establish primitive correspondence before comparing their update compatibility.

Specifically, we collect response statistics for each primitive on a calibration set and construct pairwise matching costs between each expert and a reference expert (expert 1), which is chosen arbitrarily. Hungarian matching (Gerards, 1995) finds the minimum-cost one-to-one assignment, yielding a permutation of primitive groups. The resulting permutation is applied jointly to the corresponding gate/up row blocks and down-projection columns, preserving the expert function and the channel order within each group. After alignment, primitives with the same group index represent corresponding computations across experts and can be meaningfully compared for ownership assignment. Alignment statistics are reported in Appendix I.2.

3.4.2 Primitive Ownership Induction

After alignment, MoP determines the ownership of each primitive. Experts in the same ownership group share the corresponding gate/up parameters, while their down projections remain route-specific.

Let K denote the number of experts. For each primitive, we group the K experts into disjoint ownership groups $\pi = \{B_1, \dots, B_m\}$, where every expert belongs to exactly one group. Each group B_j contains experts that share one gate/up parameter copy. We use Π_K to denote the set of all possible ownership partitions over K experts. The fully private partition is $\pi_{\text{priv}} = \{\{1\}, \dots, \{K\}\}$.

MoP infers ownership from gate/up update compatibility. Let $\mathcal{I}_{\text{cal}}$ index visual tokens in the calibration set and $r_i^\ell = \partial \mathcal{L}_{\text{task}} / \partial \Delta_i^\ell$ denote the upstream residual gradient. For primitive g of expert k at layer ℓ, we compute

$$v_{\ell,g,k} = \text{norm}\left(\text{vec}\left[\begin{array}{c} \sum_{i \in \mathcal{I}_{\text{cal}}} \nabla_{W_{\text{gate}}^{\ell,g,k}} \left\langle \text{sg}(r_i^\ell), P_{\ell,g,k}(h_i^\ell) \right\rangle \\ \sum_{i \in \mathcal{I}_{\text{cal}}} \nabla_{W_{\text{up}}^{\ell,g,k}} \left\langle \text{sg}(r_i^\ell), P_{\ell,g,k}(h_i^\ell) \right\rangle \end{array} \right] \right), \tag{3}$$

where $P_{\ell,g,k}(h_i^\ell)$ is the aligned primitive response, $W_{\text{gate/up}}^{\ell,g,k}$ are its gate/up parameters, sg denotes stop-gradient, vec concatenates the gradient blocks, and $\text{norm}(x) = x/(\|x\|_2 + \epsilon)$. Pairwise update compatibility is

$$C_{\ell,g}^{\text{upd}}(k, k') = \langle v_{\ell,g,k}, v_{\ell,g,k'} \rangle . \tag{4}$$

The down projection participates in $P_{\ell,g,k}$ but is excluded from the update signature because its ownership remains route-specific.

For a candidate partition $\pi \in \Pi_K \setminus \{\pi_{\text{priv}}\}$, we score its weakest compatibility among all shared blocks:

$$S_{\ell,g}(\pi) = \min_{\substack{B \in \pi,\ |B|>1 \\ k<k',\ k,k' \in B}} C_{\ell,g}^{\text{upd}}(k, k'), \tag{5}$$

where B denotes a block of experts that would share parameters. A partition is admissible when $S_{\ell,g}(\pi) \geq \tau(\pi)$, where $\tau(\pi)$ is its sharing threshold.

For each primitive, Select chooses the coarsest admissible partition, i.e., the partition with the fewest ownership blocks; ties are broken by $S_{\ell,g}(\pi)$. If no non-private partition is admissible, it assigns π_{priv}. A layerwise budget $b \leq G$ retains at most b non-private assignments and sets the remainder to π_{priv}:

$$(\pi_{\ell,1}, \dots, \pi_{\ell,G}) = \text{Select}_{\tau,b}(\mathbf{C}^\ell) \in \Pi_K^G, \tag{6}$$

where $\mathbf{C}^\ell$ collects the pairwise compatibility matrices $\left[C_{\ell,g}^{\text{upd}}(k, k')\right]_{k,k'=1}^K$ for $g = 1, \dots, G$. Appendix D gives the detailed selection procedure.

3.5 Primitive-level Parameter Sharing and Recovery

Given the ownership partitions $\{\pi_{\ell,g}\}$ selected above, we first instantiate the corresponding parameter sharing and then recover the consolidated model. For each block $B \in \pi_{\ell,g}$, experts in B share

one gate projection $W_{\text{gate}}^{\ell,g,B}$ and one up projection $W_{\text{up}}^{\ell,g,B}$, while each expert k retains its own down projection $W_{\text{down}}^{\ell,g,k}$. Let $B_{\ell,g}(k)$ denote the block containing expert k. The resulting residual expert is

$$E_{\ell,k}(h) = \sum_{g=1}^{G} W_{\text{down}}^{\ell,g,k} \left[\text{SiLU}\left(W_{\text{gate}}^{\ell,g,B_{\ell,g}(k)} h \right) \odot W_{\text{up}}^{\ell,g,B_{\ell,g}(k)} h \right]. \tag{7}$$

We instantiate this sharing by averaging the aligned Stage-1 gate/up parameters within each ownership block and using the average as the shared initialization. The aligned down projections are inherited directly from Stage 1 and remain route-specific.

Since parameter sharing changes the specialized experts learned in Stage 1, we further perform a recovery stage to adapt the consolidated model while keeping the ownership partitions fixed. In addition to the task objective and router regularization, we distill the residual responses of the frozen Stage-1 experts to preserve their learned specialization while adapting to the new sharing structure.

3.6 Training Objective and Inference

Training Objective. Our two-stage optimization first learns specialized residual transformations and then adapts them to the induced primitive-sharing structure. In Stage 1, we optimize

$$\mathcal{L}_{\text{S1}} = \mathcal{L}_{\text{task}} + \lambda_{\text{r}} \mathcal{L}_{\text{router}}, \tag{8}$$

where $\mathcal{L}_{\text{task}}$ provides task supervision and $\mathcal{L}_{\text{router}}$ regularizes sparse routing. Primitive ownership is then induced from the specialized experts and kept fixed thereafter. Stage 2 optimizes

$$\mathcal{L}_{\text{S2}} = \mathcal{L}_{\text{task}} + \lambda_{\text{r}} \mathcal{L}_{\text{router}} + \lambda_{\text{d}} \mathcal{L}_{\text{distill}}, \tag{9}$$

where $\mathcal{L}_{\text{distill}}$ preserves the residual responses learned in Stage 1 while adapting the model to the consolidated parameter-sharing structure. Both stages use DFHR (Section 3.3.2); full loss definitions are provided in Appendix E.

Inference. At inference, the router selects one residual expert for each visual token at every routed layer, and only the selected route is evaluated. Primitive-level sharing changes parameter ownership but not the active residual width, so the model retains Top-1 sparse inference with unchanged nominal residual FLOPs.

4 Experiments

We evaluate the breadth of the Vision-Cheers checkpoint in Table 1 and the effect of MoP under matched continuation in Section 4.4, followed by ownership-topology analysis. Furthermore, we also validate whether MoP generalizes beyond Cheers using Bagel in Table 3.

4.1 Experimental Setup

All experiments initialize from pretrained Cheers and follow the Stage-1 specialization and Stage-2 MoP recovery pipeline. Vision-Cheers is trained on a large-scale multimodal corpus covering dense perception, generation, and understanding tasks. Detailed training configurations, dataset composition, parameter accounting, evaluation protocols, and release information are provided in Appendix J and related appendices. The dataset and implementation will be released upon publication.

Evaluation protocols. Controlled comparisons use identical prediction and evaluation pipelines across variants. Referring segmentation is evaluated on RefCOCO and RefCOCOg validation splits, and depth estimation reports AbsRel and δ_1 on NYUv2/DIODE. External reference results follow their published protocols.

Implementation. Construction details, training procedures, and infrastructure configurations are provided in Appendices D and Appendices C.

Table 1: Capabilities of a single Vision-Cheers checkpoint across (a) image generation and multimodal understanding, (b) referring segmentation, and (c) monocular depth estimation. Panel (a) compares Vision-Cheers against general-purpose generation and multimodal systems, while panels (b,c) compare with task-focused perception models. Parameter superscripts denote total (T), active (A), or backbone-only (B) counts. ^{F} denotes Task-focused perception model, and ^{G} means generalist generative perception model.

(a) Image generation and multimodal understanding

		Image generation (↑)				Multimodal understanding (↑)					
Model	P_{active}	Single Obj.	Counting	Position	Overall	MMStar	MMMU	POPE	AI2D	OCR	Chart
SD3-Medium(Esser et al., 2024)	2B	0.99	0.72	0.33	0.74	–	–	–	–	–	–
Qwen2-VL(Wang et al., 2024a)	2B	–	–	–	–	48.0	41.1	–	74.7	80.9	73.5
Emu3(Wang et al., 2024b)	8B	0.99	0.42	0.49	0.66	–	31.6	85.2	70.0	68.7	68.6
Show-o2(Xie et al., 2026)	1.5B	0.99	0.55	0.46	0.73	43.4	37.1	–	69.0	24.5	40.0
JanusFlow(Ma et al., 2025)	1.3B	–	–	–	0.63	40.6	29.3	88.0	54.2	53.2	64.6
Janus-Pro(Chen et al., 2025)	1.5B	0.98	0.51	0.65	0.73	43.1	36.3	86.2	64.5	48.7	23.4
Harmon(Wu et al., 2025b)	1.5B	0.99	0.66	0.74	0.76	35.3	38.9	87.6	57.0	11.2	29.8
Tar(Han et al., 2026)	1.5B	0.99	0.76	0.57	0.76	–	36.0	88.4	–	–	–
Cheers(Zhang et al., 2026)	1.5B	0.98	0.65	0.63	0.78	50.9	36.0	87.9	74.4	58.4	75.7
Vision-Cheers	$3\mathrm{B}^{\mathrm{A}}$	0.98	0.73	0.85	0.85	53.0	37.8	88.6	75.5	64.2	79.1

(b) Referring segmentation (cIoU↑)

Method	Params.	RefCOCO	RefCOCOg
Open-source models			
LISA-7B^{F} (Lai et al., 2024)	$7.70\mathrm{B}^{\mathrm{T}}$	74.1	66.4
LISA-7B (ft)^{F} (Lai et al., 2024)	$7.70\mathrm{B}^{\mathrm{T}}$	74.9	67.9
PixelLM^{F} (Ren et al., 2024)	$13\mathrm{B}^{\mathrm{T}}$	72.7	65.8
Text4Seg^{F} (Lan et al., 2025)	$8\mathrm{B}^{\mathrm{B}}$	79.2	74.0
Seg-R1^{F} (You & Wu, 2025)	$3\mathrm{B}^{\mathrm{T}}$	58.7	57.9
PaDT^{F} (Su et al., 2026)	$3.8\mathrm{B}^{\mathrm{T}}$	76.1	70.5
Vision-Cheers^{G}	$3\mathrm{B}^{\mathrm{A}}$	73.5	69.0
Closed-source model			
Vision Banana^{G} (Gabeur et al., 2026)	–	–	73.8

(c) Monocular depth estimation on NYUv2

Method	Params.	AbsRel↓	δ_1 ↑
Open-source models			
Depth Anything^{F} (Yang et al., 2024a)	$0.34\mathrm{B}^{\mathrm{T}}$	4.3	98.1
Depth Anything V2^{F} (Yang et al., 2024b)	$0.34\mathrm{B}^{\mathrm{T}}$	4.5	97.9
Metric3D^{F} (Hariat et al., 2025)	$0.2\mathrm{B}^{\mathrm{T}}$	8.3	94.4
WorDepth^{F} (Zeng et al., 2024)	$0.3\mathrm{B}^{\mathrm{T}}$	8.8	93.9
DICEPTION^{F} (Zhao et al., 2026)	$2\mathrm{B}^{\mathrm{B}}$	6.1	96.0
Lotus-2 (FLUX)^{F} (He et al., 2025)	$17.59\mathrm{B}^{\mathrm{T}}$	4.1	97.6
Vision-Cheers^{G}	$3\mathrm{B}^{\mathrm{A}}$	8.5	93.2
Closed-source model			
Vision Banana^{G} (Gabeur et al., 2026)	–	8.1	94.8

4.2 Vision-Cheers as a System-Level Instantiation

Table 1 evaluates the Vision-Cheers checkpoint after 48,196 Stage-1 and 1,000 Stage-2 steps, alongside task-focused perception models and generation and understanding systems. Appendix J documents its training data.

4.2.1 Native Generation and Multimodal Understanding

Vision-Cheers achieves GenEval Overall 0.85 through the visual residual pathway. The same checkpoint scores 53.0 on MMStar, 37.8 on MMMU-Val, 88.6 on POPE, 75.5 on AI2D, 64.2 on OCR-Bench, and 79.1 on ChartQA (Table 1, panel a). Text-target understanding disables residual experts ($m_i = 0$), so these benchmarks measure backbone capability after continued training. Appendix F gives the benchmark breakdowns and published reference comparisons.

4.2.2 Referring Segmentation and Monocular Depth

Vision-Cheers achieves 73.5/69.0 cIoU on RefCOCO and RefCOCOg, respectively. On monocular depth estimation, it obtains 8.5/93.2 and 34.3/73.0 AbsRel/δ_1 on NYUv2 and DIODE, respectively (Table 1, panels b,c). These results show that a single checkpoint supports both segmentation and depth prediction through the same generative interface.

Table 2: **Ablation studies of the two-stage framework under the controlled 1-M setting.** Stage-1 studies routed specialization, while Stage-2 compares ownership strategies starting from the same Disjoint checkpoint.

Method	Stage-1 Steps	Stage-2 Steps	Logical Params	cIoU	gIoU	AbsRel	δ_1
Stage 1: Routed specialization							
Direct FT	1,953	–	–	46.5	46.3	12.1	86.7
Disjoint checkpoint	1,953	–	1.7B	56.7	57.0	10.9	88.3
Stage 2: Ownership comparison							
Continued Disjoint	1,953	1,000	1.7B	56.6	61.2	10.8	87.9
Random MoP	1,953	1,000	1.4B	38.7	38.4	10.7	88.5
Fixed Partial	1,953	1,000	1.3B	27.2	27.7	11.3	87.4
MoP(Vision-Cheers)	1,953	1,000	1.4B	**62.2**	**63.3**	**10.3**	**88.9**

4.3 Stage-1 Study: Effect of Routed Specialization

We compare direct continued fine-tuning with Stage-1 routed residual specialization from the same pretrained Cheers checkpoint under the controlled 1-M manifest setting described in Appendix A. Both variants use identical data and training budgets, while Stage-1 additionally introduces sparsely routed residual experts. As shown in Table 2, routed specialization improves RefCOCO validation performance from 46.5/46.3 to 56.7/57.0 cIoU/gIoU, demonstrating the benefit of route-specific residual transformations.

4.4 Stage-2 Study: Controlled Ownership Comparison

Table 2 compares MoP with continued disjoint training under the controlled 1-M continuation setting from a common Stage-1 Disjoint checkpoint. All continuation variants use identical data, optimization, trainable modules, DFHR, distillation, and evaluation settings. MoP and Random MoP additionally share grouping, response alignment, consolidation, down-projection treatment, binding, and sharing configuration, each retaining 1.414B residual parameters. They differ in ownership assignment: MoP uses gate/up update compatibility and Random MoP randomizes the assignment. Appendices D and G give construction and training details.

Capacity-controlled comparison. Starting from the same disjoint checkpoint, MoP reduces residual parameters from 1.734B to 1.414B (18.5% reduction), while improving RefCOCO cIoU/gIoU and NYUv2 depth metrics. This parameter-controlled comparison suggests that the observed improvement is associated with the induced ownership structure rather than additional expert capacity.

Ownership selection. Random MoP performs substantially worse than compatibility-based MoP despite the same logical parameter budget, showing that the ownership assignment, rather than sharing alone, is important.

Structural control. Fixed Partial shares all four groups per layer through a predetermined pair-sharing topology, retaining 1.349B residual parameters versus MoP's 1.414B with at most two shared groups per layer. Active residual width and recovery settings are identical. Appendix G.2 specifies the topology, and Table 7 gives optimization settings.

4.5 Cross-Architecture Validation on Bagel

To examine whether MoP relies on a specific UMM architecture, we apply the same primitive-level ownership mechanism to Bagel, another pretrained UMM, resulting in Vision-Bagel. We perform a matched 0.5M continuation experiment from a common Stage-1 disjoint specialization checkpoint, comparing continued disjoint training with MoP under identical data, optimization, and evaluation settings. The only difference is the ownership structure of residual experts: continued disjoint training preserves independent expert parameters, whereas MoP reorganizes expert ownership through primitive-level sharing. Table 3 shows that MoP improves dense visual perception while reduc-

Table 3: **Cross-architecture validation of MoP on Bagel under the controlled 0.5M setting.** Stage-1 disjoint specialization is trained for 1,400 steps, followed by 500-step continuation with either disjoint ownership or MoP from the same checkpoint.

Method	Stage-1 Steps	Stage-2 Steps	Logical Residual Params	RefCOCO val (cIoU)	NYUv2 (AbsRel/δ_1)
Direct FT	1,400	–	–	27.8	5.6/96.5
Stage-1 Disjoint	1,400	–	1.7B	34.9	6.0/96.2
Continued Disjoint	1,400	500	1.7B	33.8	5.5/96.8
MoP(Vision-Bagel)	1,400	500	1.4B	35.8	5.3/97.0

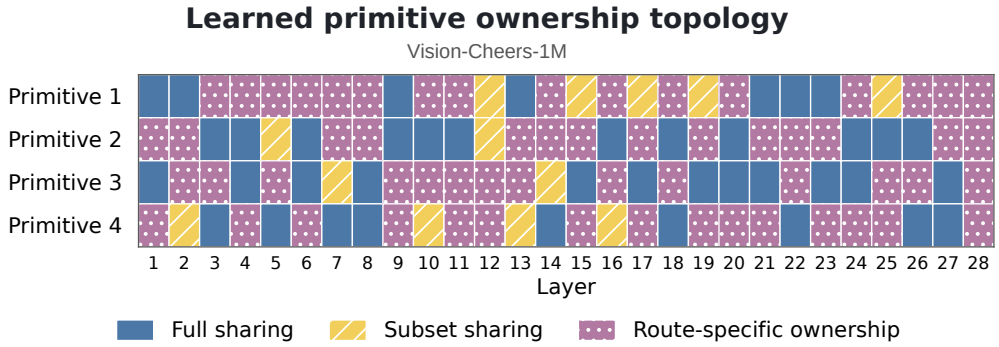

Figure 3: **Gate/up ownership in the Vision-Cheers-1M model.** Colors and patterns identify full, subset, or route-specific ownership, with at most two shared groups per layer. Indices are one-based, and primitive indices are local to each layer. Table 11 reports aggregate statistics.

ing residual parameters on Bagel. This cross-architecture validation suggests that primitive-level ownership sharing is not specific to Cheers, but can be applied to different UMM backbones.

4.6 Induced Ownership Topology

The offline-induced ownership topology of Vision-Cheers-1M contains 40 fully shared, 13 subset-shared, and 59 route-specific assignments across 28 layers and four groups per layer. The resulting topology satisfies the layer-wise sharing budget $b = 2$, with shared and route-specific primitives coexisting rather than collapsing to either fully shared or fully independent experts.

These ownership assignments reduce logical gate/up copies from 336 expert-specific copies to 243 while keeping route-specific down projections. The residual parameter count decreases from 1.734B to 1.414B (18.5%), while active residual parameters remain unchanged. Detailed parameter accounting and topology statistics are provided in Appendix I and Table 11.

Figure 3 visualizes the induced layer–group ownership structure. Importantly, this parameter reduction does not trade off perception quality: MoP achieves higher performance than the larger disjoint expert baseline (Section 4.4), suggesting that the gains arise from the induced ownership structure rather than increased residual capacity.

5 Conclusion

We present Mixture of Primitives (MoP), a fine-grained expert-sharing mechanism that decouples routing identity from parameter ownership in sparse MoE models. MoP identifies reusable primitives across specialized experts and selectively shares compatible components while preserving route-specific transformations.

Applied to unified multimodal models, MoP enables Cheers to support dense visual perception while maintaining generation and understanding capabilities. Controlled experiments and cross-architecture validation show that primitive-level ownership improves perception performance over disjoint experts while reducing residual parameters and preserving sparse execution.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and revision, language polishing, consistency checking, LaTeX preparation, and code assistance for visualization and analysis scripts. Generative AI was not used to fabricate or substitute experimental results, data, or citations. All AI-assisted text, code, figures, and analyses were reviewed by the authors; AI-assisted code was inspected and validated against the underlying experimental outputs, and manuscript claims and references were manually checked for correctness and consistency. We take full responsibility for the content of this work, including all text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human-subject experiments or the collection of new personal data. Our experiments use established vision and multimodal benchmarks and are intended to study model architecture and parameter sharing for generative visual perception. We are not aware of additional ethical concerns specific to the proposed method.

REPRODUCIBILITY STATEMENT

Section 3 specifies the MoP parameterization, topology selection, consolidation, recovery, and routing procedure. Section 4.4 and Appendices E and G document the controlled comparisons, optimization settings, calibration procedure, and alternative configurations. Appendix J reports the final-model corpus and its contained targeted subset, and Appendix J.4 reports the final-model stage settings separately from the controlled-study hyperparameters. Evaluation protocols, parameter accounting, and topology analysis appear in Appendices H, I, and I.

The induced ownership structures used in our experiments are represented by explicit topology artifacts and calibration outputs. These artifacts, together with the alignment results and update-signature files, define the ownership assignments used during recovery. We provide these intermediate artifacts to enable verification and reuse of the reported topology. Reconstructing the exact topology from scratch additionally requires preserving the calibration samples and stochastic sampling states used during calibration.

The code is available in the anonymous repository: https://anonymous.4open.science/r/MoP-DF2D

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. doi: 10.48550/arXiv.1308.3432. URL https://arxiv.org/abs/1308.3432.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2024. URL https://arxiv.org/abs/2405.09818.

Tianlong Chen, Xuxi Chen, Xianzhi Du, Abdullah Rashwan, Fan Yang, Huizhong Chen, Zhangyang Wang, and Yeqing Li. AdaMV-MoE: Adaptive multi-task vision mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17346–17357, 2023a.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G. Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11828–11837, 2023b.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models, 2024. URL https://arxiv.org/abs/2401.06066.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining, 2025. URL https://arxiv.org/abs/2505.14683.

Haiwen Diao, Penghao Wu, Hanming Deng, Jiahao Wang, Shihao Bai, Silei Wu, Weichen Fan, Wenjie Ye, Wenwen Tong, Xiangyu Fan, et al. Sensenova-u1: Unifying multimodal understanding and generation with neo-unify architecture. *arXiv preprint arXiv:2605.12500*, 2026.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. URL http://jmlr.org/papers/v23/21-0998.html.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27503–27516, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e77910ebb93b511588557806310f78f1-Abstract.html.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, Kyle Genova, Nithish Kannen, Sherry Ben, Yandong Li, Mandy Guo, Suhas Yogin, Yiming Gu, Huizhong Chen, Oliver Wang, Saining Xie, Howard Zhou, Kaiming He, Thomas Funkhouser, Jean-Baptiste Alayrac, and Radu Soricut. Image generators are generalist vision learners, 2026. URL https://arxiv.org/abs/2604.20329.

Yulu Gan, Sung Woo Park, Alexander Schubert, Anthony Philippakis, and Ahmed Alaa. InstructCV: Instruction-tuned text-to-image diffusion models as vision generalists. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/ed9f00cb7dd5fbdc2175d55e2fdf1b05-Abstract-Conference.html.

Ziyan Gan, Fangxin Liu, Chenyang Guan, Junjie Wang, Ning Yang, Haomin Li, Xiang Li, Siran Yang, Jiamang Wang, Lin Qu, Zongwu Wang, Li Jiang, and Haibing Guan. PCoMoE: Shifting MoE inference from monolithic expert selection to fine-grained path composition, 2026. URL https://arxiv.org/abs/2609.01024.

AMH Gerards. Matching. *Handbooks in operations research and management science*, 7:135–224, 1995.

Xiaoyang Han, Jianhua Li, Kewang Deng, Zukai Chen, Xuanke Shi, Sihan Wang, Boxuan Li, Linyan Wang, Siyi Xie, Xin You, Jinsheng Quan, Zhongang Cai, Haiwen Diao, Ziwei Liu, Lei Yang, Dahua Lin, and Quan Wang. Vision as unified multimodal generation, 2026. URL https://arxiv.org/abs/2607.06560.

Marwane Hariat, Antoine Manzanera, and David Filliat. Improved monocular depth prediction using distance transform over pre-semantic contours with self-supervised neural networks. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21868–21879. IEEE, 2025.

Jing He, Haodong Li, Mingzhi Sheng, and Ying-Cong Chen. Lotus-2: Advancing geometric dense prediction with powerful image generative model. *arXiv preprint arXiv:2512.01030*, 2025.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9579–9589. IEEE, 2024.

Mengcheng Lan, Chaofeng Chen, Yue Zhou, Jiaxing Xu, Yiping Ke, Xinjiang Wang, Litong Feng, and Wei Zhang. Text4seg: Reimagining image segmentation as text generation. In *International Conference on Learning Representations*, volume 2025, pp. 1634–1661, 2025.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. GShard: Scaling giant models with conditional computation and automatic sharding, 2020. URL https://arxiv.org/abs/2006.16668.

Pingzhi Li, Zhenyu Zhang, Prateek Yadav, Yi-Lin Sung, Yu Cheng, Mohit Bansal, and Tianlong Chen. Merge, then compress: Demystify efficient SMoE with hints from its routing policy. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=eFWG9Cy3WK.

Hanxue Liang, Zhiwen Fan, Rishov Sarkar, Ziyu Jiang, Tianlong Chen, Kai Zou, Yu Cheng, Cong Hao, and Zhangyang Wang. M^{3}ViT: Mixture-of-experts vision transformer for efficient multi-task learning with model-accelerator co-design. In *Advances in Neural Information Processing Systems*, volume 35, pp. 28441–28457, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/b653f34d576d1790481e3797cb740214-Abstract-Conference.html.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://arxiv.org/abs/2411.04996.

Weifeng Lin, Xinyu Wei, Renrui Zhang, Le Zhuo, Shitian Zhao, Siyuan Huang, Junlin Xie, Gao Peng, and Hongsheng Li. Pixwizard: Versatile image-to-image visual assistant with open-language instructions. In *International Conference on Learning Representations*, volume 2025, pp. 38680–38711, 2025.

Shikun Liu, Edward Johns, and Andrew J. Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1871–1880, 2019.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai Yu, et al. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7739–7751. IEEE, 2025.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3994–4003, 2016.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503, 2021. doi: 10.18653/v1/2021.eacl-main.39.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26364–26373. IEEE, 2024.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL `https://openreview.net/forum?id=B1ckMDqlg`.

Ying Shen, Zhiyang Xu, Qifan Wang, Yu Cheng, Wenpeng Yin, and Lifu Huang. Multimodal instruction tuning with conditional mixture of LoRA. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 637–648, 2024. doi: 10.18653/v1/2024.acl-long.38.

Yongyi Su, Haojie Zhang, Shijie Li, Nanqing Liu, Jingyi Liao, Junyi Pan, Yuan Liu, Xiaofen Xing, Chong Sun, Chen Li, et al. Patch-as-decodable-token: Towards unified multi-modal vision tasks in mllms. In *International Conference on Learning Representations*, volume 2026, pp. 147893–147919, 2026.

Ximeng Sun, Rameswar Panda, Rogerio Feris, and Kate Saenko. AdaShare: Learning what to share for efficient deep multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024a.

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

Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. SegGPT: Towards segmenting everything in context. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1130–1140, 2023b.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024b.

Xinlong Wang, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Zhen Li, Yuqi Wang, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Chunlei Men, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Zhongyuan Wang, and Tiejun Huang. Multimodal learning with next-token prediction for large multimodal models. *Nature*, 650:327–333, 2026. doi: 10.1038/s41586-025-10041-x.

Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. AdaMix: Mixture-of-adaptations for parameter-efficient model tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5744–5760, 2022. doi: 10.18653/v1/2022.emnlp-main.388.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12966–12977, 2025a.

Lemeng Wu, Mengchen Liu, Yinpeng Chen, Dongdong Chen, Xiyang Dai, and Lu Yuan. Residual mixture of experts. *arXiv preprint arXiv:2204.09636*, 2022.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Zhonghua Wu, Qingyi Tao, Wentao Liu, Wei Li, and Chen Change Loy. Harmonizing visual representations for unified multimodal understanding and generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17739–17750. IEEE, 2025b.

Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation, 2024. URL https://arxiv.org/abs/2409.11340.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2408.12528.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *Advances in Neural Information Processing Systems*, 38:47490–47518, 2026.

Yifan Xu, Xiaoshan Yang, Yaguang Song, and Changsheng Xu. Libra: Building decoupled vision system on large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55371–55388. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/xu24ab.html.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10371–10381. IEEE, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in neural information processing systems*, 37:21875–21911, 2024b.

Zuyao You and Zuxuan Wu. Seg-r1: Segmentation can be surprisingly simple with reinforcement learning. *arXiv preprint arXiv:2506.22624*, 2025.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

Amir R. Zamir, Alexander Sax, William Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Ziyao Zeng, Daniel Wang, Fengyu Yang, Hyoungseob Park, Stefano Soatto, Dong Lao, and Alex Wong. Wordepth: Variational language prior for monocular depth estimation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9708–9719. IEEE, 2024.

Yichen Zhang, Da Peng, Zonghao Guo, Zijian Zhang, Xuesong Yang, Tong Sun, Shichu Sun, Yidan Zhang, Yanghao Li, Haiyan Zhao, et al. Cheers: Decoupling patch details from semantic representations enables unified multimodal comprehension and generation. *arXiv preprint arXiv:2603.12793*, 2026.

Canyu Zhao, Yanlong Sun, Mingyu Liu, Huanyi Zheng, Muzhi Zhu, Zhiyue Zhao, Hao Chen, Tong He, and Chunhua Shen. Diception: A generalist diffusion model for visual perceptual tasks. *Advances in Neural Information Processing Systems*, 38:40108–40155, 2026.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *International Conference on Learning Representations*, pp. 6446–6469, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/12678c3948153f4bc391f51e2082bd6e-Paper-Conference.pdf.

A Training Settings and Checkpoint Roles

Table 4 describes the large-scale system training. All Vision-Cheers system results in Table 1 use the checkpoint after 48,196 Stage-1 specialization steps and 1,000 Stage-2 recovery steps.

Table 4: Training settings and checkpoint roles. Data sizes describe corpus or manifest entries, not distinct images or optimization exposure. The Stage-2 pool is a subset of the Stage-1 corpus; The final Vision-Cheers checkpoint is trained on the full multimodal corpus, while the 1-M setting is used for controlled ablations under matched training conditions. Batch sizes are global, and continuation steps are per variant.

Setting / stage	Data pool	Steps	Batch	Role
Final: Stage 1	Approx. 38M corpus	48,196	800	Residual specialization
Final: Stage 2	1,770,538-record subset	1,000	800	Final checkpoint
1-M: specialization	1,000,000-entry manifest	1,953	512	Common disjoint source
1-M: continuation	Same 1-M manifest	1,000	512	MoP and matched controls

B Dataset Composition and Provenance

The final Vision-Cheers training corpus contains 38,268,990 JSONL training records. This count refers to record occurrences. The corpus consists of 28,268,990 main records and 10,000,000 replay entries inserted during stream construction.

The replay component reuses existing training records rather than introducing a separate annotation source. The final stream therefore contains repeated records by design. After exact normalized JSON-object deduplication, 22,098,946 unique records remain, while image-level deduplication is not performed because multiple records may correspond to different tasks, annotations, or instructions associated with the same image.

The final stream covers five task categories: understanding, generation, referring segmentation/grounding, instance segmentation, and depth estimation. The category distribution follows the intended 1:1:4:4:4 mixture ratio, with the replay stream preserving the same task proportions.

C Compute Infrastructure

All experiments are conducted on a cluster equipped with 400 MuXi Mars X203 AI accelerators. MuXi Mars X203 is a general-purpose AI accelerator designed for large-scale deep learning workloads, providing the computational resources used for training and evaluation in this work. All experiments are performed using distributed training across multiple accelerators with mixed-precision computation.

D Detailed Method Equations

Residual routing. Let $\bar{E}_{\ell,k} = E_{\ell,k}$ during specialization and $\bar{E}_{\ell,k} = E^{\Phi}_{\ell,k}$ after tying. At routed layer $\ell \in \mathcal{L}_r$, the token update is

$$\Delta_i^\ell = m_i \bar{E}_{\ell,z_i^\ell}(h_i^\ell), \qquad \tilde{s}_i^\ell = s_i^\ell + F_\ell(h_i^\ell) + \Delta_i^\ell, \tag{10}$$

where $h_i^\ell = \mathrm{RMSNorm}(s_i^\ell)$ and $m_i = 1$ only for visual tokens. The layer-specific router is

$$a_i^\ell = R_\ell(h_i^\ell), \qquad p_i^\ell = \mathrm{softmax}(a_i^\ell), \qquad z_i^\ell = \arg\max_{k\in[K]} p_{i,k}^\ell. \tag{11}$$

The one-hot vector used by DFHR is $q_{i,k}^\ell = \mathbf{1}[k = z_i^\ell]$.

Residual primitives and alignment. A specialized expert and its untied primitive g are

$$E_{\ell,k}(h) = W_{\mathrm{down}}^{\ell,k}\left[\mathrm{SiLU}\left(W_{\mathrm{gate}}^{\ell,k}h\right) \odot W_{\mathrm{up}}^{\ell,k}h\right], \tag{12}$$

$$P_{\ell,g,k}(h) = W_{\mathrm{down}}^{\ell,g,k}\left[\mathrm{SiLU}\left(W_{\mathrm{gate}}^{\ell,g,k}h\right) \odot W_{\mathrm{up}}^{\ell,g,k}h\right], \tag{13}$$

Here $\odot$ is elementwise multiplication. For hidden width d, the full gate/up matrices lie in $\mathbb{R}^{d_{\mathrm{op}}\times d}$ and the down matrix in $\mathbb{R}^{d\times d_{\mathrm{op}}}$. We use equal-width groups with $d_g = d_{\mathrm{op}}/G$, so every group permutation preserves parameter shape. Let $\mathfrak{S}_G$ be the permutations of $[G]$, set $\pi_{\ell,1} = \mathrm{id}$ for the reference expert, and let $\mathcal{I}_{\mathrm{cal}}$ index calibration token states. For $k \in \{2,\dots,K\}$, response alignment solves

$$\pi_{\ell,k}^* = \arg\max_{\pi\in\mathfrak{S}_G} \sum_{g=1}^{G} \mathbb{E}_{i\in\mathcal{I}_{\mathrm{cal}}}\left[\cos\left(P_{\ell,g,1}(h_i^\ell), P_{\ell,\pi(g),k}(h_i^\ell)\right)\right]. \tag{14}$$

The permutation acts on complete group blocks, synchronizing gate/up rows and down columns without changing the expert function or the ordering within a block. Gate/up averaging and gradient comparison use these retained within-group coordinates. The response score determines correspondence only. After alignment, gate/up update compatibility is

$$C_{\ell,g}^{\mathrm{upd}}(k,k') = \langle v_{\ell,g,k}, v_{\ell,g,k'}\rangle. \tag{15}$$

Equation 3 differentiates a linearized residual-response objective with respect to gate/up parameters, holding calibration states, upstream gradients, and down weights fixed. The accumulated gate/up gradients are concatenated and normalized. Their inner product measures local directional agreement over the calibration mixture, rather than the finite response change from averaging weights. This compatibility supplies the ownership selector.

Partition induction. The reported selector hyperparameters are $\tau_{\mathrm{full}} = -0.02$, $\tau_{\mathrm{pair}} = 0.02$, and $b = 2$. The thresholds serve different partition tests: full sharing tests all three pairwise relations against the permitted compatibility range, while pair sharing requires positive agreement for the selected pair. They specify the reported configuration, rather than an optimized parameter-reduction target.

For each routed layer, apply the following procedure:

1. Traverse groups in the order stored with their calibration signatures. Read c_{12}, c_{13}, c_{23} from gate/up update compatibility.
2. Compute $f_g = \min(c_{12}, c_{13}, c_{23})$. If $f_g \geq \tau_{\mathrm{full}}$, assign the full candidate 123 and skip the pair test.
3. Otherwise, compute $p_g = \max(c_{12}, c_{13}, c_{23})$. If $p_g \geq \tau_{\mathrm{pair}}$, assign the maximizing pair with the third route specific. Equal pair scores select the first pair in $(1,2), (1,3), (2,3)$ order.
4. If neither test passes, assign $1|2|3$.
5. For every non-specific candidate, compute $S_{\ell,g}(\widehat{\phi}_{\ell,g})$ from Equation 5.
6. Stably sort these candidates by decreasing S. Equal scores preserve group traversal order.
7. Retain the first $\min(b, N_{\mathrm{eligible}})$ candidates with their assigned partitions, where N_{eligible} is the number of non-specific candidates in that layer.
8. Set all other groups to $1|2|3$ and keep the resulting topology fixed during recovery.

The layerwise structural sharing constraint is

$$\sum_{g=1}^{G} \mathbf{1}[|\phi_{\ell,g}| < K] \le b, \qquad \ell \in \mathcal{L}_r. \tag{16}$$

Full and pair candidates each consume one shared-group slot, not an equal number of parameters. Their different gate/up copy counts enter Equation 29 after topology induction. With $G = 4$ and $b = 2$, every route retains at least two specific groups.

Random ownership selection. Random MoP samples ownership assignments using three topology seeds under the matched construction, sharing configuration, and recovery settings in Section 4.4. Each sampled topology undergoes recovery and evaluation, and the reported task scores are arithmetic means across the three runs. MoP uses the gate/up update signatures in Equation 3; both variants retain 1.414B residual parameters (Table 7).

Recovery and routing controls. Let $\mathcal{V}_\ell = \{i : m_i = 1\}$ be the visual token indices at layer ℓ in the current minibatch and $\mathcal{L}_+ = \{\ell \in \mathcal{L}_r : |\mathcal{V}_\ell| > 0\}$. Frozen teacher experts $E^T_{\ell,k}$ are loaded from the Stage-1 specialization checkpoint as buffers. They use the student's normalized states h^ℓ_i and selected expert identities z^ℓ_i. No teacher backbone or router is evaluated. The local distillation loss is

$$\mathcal{L}_{\text{dist}} = \frac{1}{|\mathcal{L}_+|} \sum_{\ell \in \mathcal{L}_+} \frac{1}{|\mathcal{V}_\ell|} \sum_{i \in \mathcal{V}_\ell} \left\| E^{\Phi}_{\ell,z^\ell_i}(h^\ell_i) - \text{sg}\Big(E^T_{\ell,z^\ell_i}(h^\ell_i)\Big) \right\|_2^2. \tag{17}$$

Squared errors are summed over hidden dimensions, averaged over visual tokens, and then averaged over eligible layers. The loss is zero if $\mathcal{L}_+$ is empty; its coefficient is applied once in Equation 25. Let $\bar{e}^\ell_i = \sum_j p^\ell_{i,j} e^\ell_{i,j}$. Differentiating the DFHR gate through the softmax gives

$$\frac{\partial \mathcal{L}_{\text{task}}}{\partial a^\ell_{i,k}} = m_i p^\ell_{i,k} \left\langle r^\ell_i, e^\ell_{i,k} - \bar{e}^\ell_i \right\rangle. \tag{18}$$

The selected-feedback control uses

$$\Delta^\ell_{i,\text{sel}} = m_i \left[1 + p^\ell_{i,z^\ell_i} - \text{sg}(p^\ell_{i,z^\ell_i})\right] e^\ell_{i,z^\ell_i}. \tag{19}$$

Unlike selected feedback, Hard MoE uses the unweighted output $\Delta^\ell_i = m_i e^\ell_{i,z^\ell_i}$ without a straight-through gate. Its router receives auxiliary and z-loss gradients but no task-loss gradient through the discrete choice. Both variants execute only the selected expert. Soft MoE uses $\Delta^\ell_i = m_i \sum_k p^\ell_{i,k} e^\ell_{i,k}$ and evaluates all three experts during training; task-loss gradients reach the router and all experts. Dense-feedback Top-1 instead evaluates all candidates for the router surrogate while updating only selected-path expert parameters.

Residual expert initialization and alignment. . Residual experts are instantiated as independent SwiGLU modules before Stage-1 specialization. The gate and up projections are independently initialized from $\mathcal{N}(0, 0.02^2)$, while the residual down projections are initialized to zero, making each residual branch function-preserving at initialization. The experts therefore share the same dimensional organization and primitive indexing convention, but do not inherit a common pretrained parameter basis. Stage-1 disjoint specialization allows the route-specific parameters to evolve independently while preserving the within-group coordinate convention.

After specialization, group-level Hungarian alignment jointly permutes gate/up row blocks and down column blocks without changing the ordering within each primitive group. Subsequent elementwise gate/up averaging is performed on these aligned within-group coordinates. Router weights are initialized with standard deviation 10^{-3}, and loading the Stage-1 specialization checkpoint for MoP construction preserves the trained residual parameters rather than reinitializing them.

Consolidation and recovery ownership. Let $W^{A,a}_{\ell,g,k}$ denote aligned weights from the Stage-1 specialization checkpoint for $a \in \{\text{gate}, \text{up}\}$, and let $W^{A,\text{down}}_{\ell,g,k}$ be the corresponding aligned down

projection. For initialization block $B^{\Psi}_{\ell,g}(k)$, consolidation applies

$$\widetilde{W}^{a}_{\ell,g,k} = \frac{1}{|B^{\Psi}_{\ell,g}(k)|} \sum_{j \in B^{\Psi}_{\ell,g}(k)} W^{A,a}_{\ell,g,j}, \qquad a \in \{\text{gate}, \text{up}\}. \tag{20}$$

The down projection is copied separately as $\widetilde{W}^{\text{down}}_{\ell,g,k} = W^{A,\text{down}}_{\ell,g,k}$, without averaging across experts. Rebinding averages the loaded gate/up copies within each recovery block $B \in \phi_{\ell,g}$ and makes its routes reference the same gate/up parameter objects. Aligned down projections remain separate, independently trainable parameter objects throughout recovery.

Here Ψ denotes initialization/consolidation blocks and Φ denotes recovery parameter ownership. MoP and Random MoP use the same mapping from an assignment to these partitions and the same averaging, binding, and recovery operations. Fixed Partial applies these operators to its predetermined assignment.

Serialization. Table 10 lists the 112 recovery assignments in Φ. Each tied gate/up object accumulates gradients from tokens routed to any expert in its block, while down projections adapt independently. The native distributed checkpoint preserves shared modules. HF export materializes expert tensors; reloading for shared execution uses the saved recovery topology to restore gate/up ownership.

E Training Objectives and Router Regularization

Let c denote the interleaved multimodal context and Θ all trainable model parameters. For text target $w_{1:M}$, Cheers uses

$$\mathcal{L}_{\text{text}} = -\frac{1}{M} \sum_{j=1}^{M} \log \Pr_{\Theta}(w_j \mid w_{<j}, c). \tag{21}$$

Let $x_1 \sim p_{\text{enc}}$ be the encoded visual target, $x_0 \sim \mathcal{N}(0, I)$ noise, and $\tau \sim \mathcal{U}(0, 1)$ flow time. Define $x_\tau = \tau x_1 + (1 - \tau)x_0$ and target velocity $d_\tau = x_1 - x_0$. The predictor $v_\Theta(x_\tau, \tau, c)$ outputs a velocity in the latent space, and the visual loss is

$$\mathcal{L}_{\text{vis}} = \mathbb{E}_{x_0, x_1, \tau} \left[\| v_\Theta(x_\tau, \tau, c) - d_\tau \|_2^2 \right]. \tag{22}$$

The target modality selects

$$\mathcal{L}_{\text{task}} = \begin{cases} \lambda_{\text{text}} \mathcal{L}_{\text{text}}, & \text{text token}, \\ \lambda_{\text{vis}} \mathcal{L}_{\text{vis}}, & \text{visual token}. \end{cases} \tag{23}$$

At fixed layer ℓ, let $\mathcal{I}^{\ell}_r = \{i : m_i = 1\}$ be the routed visual tokens and $N_r = |\mathcal{I}^{\ell}_r|$. Suppressing superscript ℓ, define $f_k = N_r^{-1} \sum_{i \in \mathcal{I}^{\ell}_r} \mathbf{1}[z_i = k]$ and $\bar{p}_k = N_r^{-1} \sum_{i \in \mathcal{I}^{\ell}_r} p_{i,k}$. We use

$$\mathcal{L}_{\text{bal}} = K \sum_{k=1}^{K} f_k \bar{p}_k, \qquad \mathcal{L}_z = \frac{1}{N_r} \sum_{i \in \mathcal{I}^{\ell}_r} \left(\log \sum_{k=1}^{K} \exp(a_{i,k}) \right)^2. \tag{24}$$

When $N_r = 0$, we set $\mathcal{L}_{\text{bal}} = \mathcal{L}_z = 0$. The complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{bal}} \mathcal{L}_{\text{bal}} + \lambda_z \mathcal{L}_z + \lambda_{\text{dist}} \mathcal{L}_{\text{dist}}, \tag{25}$$

The controlled and supplementary runs use $\lambda_{\text{text}} = \lambda_{\text{vis}} = 1$ and average router losses across routed layers. Router auxiliary and z-loss coefficients are 10^{-3} and 10^{-4}, respectively. Disjoint specialization uses $\lambda_{\text{dist}} = 0$. Continued disjoint training, Random MoP, and MoP use $\lambda_{\text{dist}} = 1$ with the same frozen specialization experts evaluated on each student's states and routes. Fixed Partial uses the same local residual-distillation settings, objective, and coefficient as MoP. For continued disjoint training, Equation 17 uses the all-specific partition. The distillation term sums hidden-coordinate errors before token and layer averaging.

Ablation optimization. All ablation runs use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon_{\text{opt}} = 10^{-8}$, zero weight decay, gradient-norm clipping at 1.0, and BF16 with FSDP/HSDP. The learning rate is constant after a 2% warmup; continuation-run expert and router parameter groups have no warmup. Pretrained parameters use a 0.1 learning-rate multiplier and routers use 1.0. Disjoint specialization uses base rate 1.5×10^{-4} and expert multiplier 1.0. Continued disjoint training, Random MoP, Fixed Partial, and MoP use base rate 3×10^{-5} and expert multiplier 0.2, with identical optimization schedules and trainable modules. With per-rank microbatch size 1, specialization uses 256 ranks and accumulation 2, while completed continuation runs use 128 ranks and accumulation 4, giving effective batch size 512. The trainable set includes the language model and residual modules, token embeddings/LM head, multimodal projectors, vision-representation module, and time embedding.

Ablation mixture and split scope. The ablation training manifest contains 1,000,000 entries with 606,981 distinct record IDs, separate from the final-model approximately 38M multimodal corpus and its contained 1,770,538-record targeted subset (Appendix J). The remaining 393,019 occurrences repeat an ID; entries therefore do not represent one million unique examples. Source metadata include COCO-derived depth and instance data, COCONut, VIPSeg, Flickr, Hypersim, LVIS, and RefCOCO. Training uses the training split of each source, and segmentation evaluation uses the standard validation splits. The ablation training mixture contains no images from the validation sets used for evaluation, including across source tasks. Referring validation reference IDs, annotations, and image–expression pairs are also disjoint from their corresponding training sources.

Ablation calibration sampling. For response alignment, we selected 256 calibration entries from the ablation manifest, of which 195 completed forward passes. For update-signature extraction, we selected 128 entries from the same manifest, of which 109 completed backpropagation of the image loss and entered signature accumulation.The source-stratified sampler uses seed 42 and metadata fields or ID prefixes to define 20 source buckets. Reservoir sampling retains up to 64 entries per bucket before shuffling the combined candidate pool; allocation is not equal across the three downstream tasks. Alignment uses at most the first 8,192 concatenated visual-target states per layer and Hungarian matching of tokenwise response cosines. Signature extraction uses at most 8,192 visual-target states per sample and layer, randomly subsampling longer sequences. It accumulates only gate and up gradient blocks from the image-loss upstream gradient, concatenates them, and normalizes each expert/group signature after accumulation. Down projections enter the primitive response but are held fixed for signature extraction.

F Complete Capability Comparisons

Table 5 extends the generation comparison in Table 1(c) with additional models and component categories. Table 6 reports the six understanding results from Table 1(d), alongside additional benchmark columns and reference models. Cheers (reported) reproduces published values.

G Supplementary Ablations and Analysis

All results in this section use the 1-M setting. Results labeled MoP are obtained with the Vision-Cheers-1M analysis model from Table 2.

G.1 Training Configurations

Table 7 gives optimization settings and logical residual parameter counts for the continuation variants in Section 4.4. Appendix D specifies construction operators, and Appendix E gives the data, teacher, and training protocol.

G.2 Ownership-Strategy and Structural Controls

Table 8 collects the MoP, Random MoP, and Fixed Partial results from the common continuation setting. The variants differ in their sharing assignments and use the construction operators in Appendix D. Fixed Partial's predetermined topology is specified below.

Table 5: Extended generation comparison and GenEval category breakdown for the final checkpoint in Table 1. Reference scores follow Zhang et al. (2026). TBD denotes pending category scores.

Model	P_{active}	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attr.	Overall
LlamaGen	0.8B	0.71	0.34	0.21	0.58	0.07	0.04	0.32
SDXL	2.6B	0.98	0.74	0.39	0.85	0.15	0.23	0.55
DALL-E 3	N/R	0.96	0.87	0.47	0.83	0.43	0.45	0.67
SD3-Medium	2B	0.99	0.94	0.72	0.89	0.33	0.60	0.74
Chameleon	7B	–	–	–	–	–	–	0.39
Show-o	1.3B	0.95	0.52	0.49	0.82	0.11	0.28	0.53
TokenFlow-XL	14B	0.95	0.60	0.41	0.81	0.16	0.24	0.55
Janus	1.3B	0.97	0.68	0.30	0.84	0.46	0.42	0.61
JanusFlow	1.3B	–	–	–	–	–	–	0.63
Transfusion	7.3B	–	–	–	–	–	–	0.63
D-DiT	2B	0.97	0.80	0.54	0.76	0.32	0.50	0.65
Emu3	8B	0.99	0.81	0.42	0.80	0.49	0.45	0.66
Janus-Pro	1.5B	0.98	0.82	0.51	0.89	0.65	0.56	0.73
Show-o2	1.5B	0.99	0.86	0.55	0.86	0.46	0.63	0.73
Harmon	1.5B	0.99	0.86	0.66	0.85	0.74	0.48	0.76
Tar	1.5B	0.99	0.91	0.76	0.81	0.57	0.51	0.76
Cheer	1.5B	0.98	0.92	0.65	0.86	0.63	0.65	0.78
Vision-Cheers	3B	0.98	0.93	0.73	0.90	0.85	0.73	0.85

Table 6: Extended multimodal-understanding comparison for the final checkpoint in Table 1. Vision-Cheers reports MMMU-Val; its OCRBench and ChartQA columns correspond to OCR and Chart in panel (d). Residual experts are disabled for text targets. Reference scores follow Zhang et al. (2026). TBD denotes pending benchmark scores; dashes denote unreported reference scores.

Model	P_{active}	SEEDBench	MMStar	MMBench	ChartQA	OCRBench	RealWorldQA	POPE	AI2D	MathVista	MMMU
MobileVLM-V2	1.4B	–	–	57.7	–	–	–	84.3	–	–	–
Qwen2-VL	2B	–	48.0	72.2	73.5	80.9	62.9	–	74.7	43.0	41.1
DeepSeek-VL	7B	70.4	–	73.2	–	45.6	–	88.1	–	36.1	36.6
mPLUG-Owl2	7B	57.8	–	64.5	22.8	25.5	50.3	86.2	55.7	–	32.7
Emu3	8B	68.2	–	58.5	68.6	68.7	57.4	85.2	70.0	–	31.6
Show-o	1.3B	51.5	–	–	–	–	–	80.0	–	–	26.7
Show-o2	1.5B	65.6	43.4	67.4	40.0	24.5	56.5	–	69.0	–	37.1
JanusFlow	1.3B	70.5	40.6	74.9	64.6	53.2	41.2	88.0	54.2	–	29.3
Janus-Pro	1.5B	68.3	43.1	75.5	23.4	48.7	52.6	86.2	64.5	–	36.3
Harmon	1.5B	67.1	35.3	65.5	29.8	11.2	49.8	87.6	57.0	–	38.9
Tar	1.5B	70.4	–	65.6	–	–	–	88.4	–	–	36.0
Cheers (reported)	1.5B	71.7	50.9	70.4	75.7	58.4	60.9	87.9	74.4	50.5	36.0
Vision-Cheers	3B	72.2	53.0	71.6	79.1	64.2	59.1	88.6	75.5	–	37.8

G.3 Primitive Alignment Analysis

Fixed Partial uses a predetermined topology repeated across all 28 routed layers. With the paper's one-based group and expert indices, groups $g = 1, 2, 3, 4$ use $12|3$, $13|2$, $23|1$, and $12|3$, respectively. These correspond to the zero-based patterns $01|2$, $02|1$, $12|0$, and $01|2$. All 112 layer-group assignments use pairwise sharing: 56 groups share experts 1–2, 28 share experts 1–3, and 28 share experts 2–3. There are zero fully shared groups and zero route-specific groups. Fixed Partial sets these assignments directly rather than using compatibility-based heterogeneous topology selection.

Fixed Partial applies the same averaging and tying rules with its predetermined $\Psi = \Phi$. This structural control retains 1.349B residual parameters at the same active residual width and recovery settings as MoP and Random MoP.

MoP exceeds Random MoP by 23.476 cIoU and 24.950 gIoU. On NYUv2, MoP reduces AbsRel from 10.70 to 10.29 and raises δ_1 from 88.50 to 88.85. Fixed Partial provides the complementary predetermined pair-sharing structural control.

G.4 Alignment Statistics

To verify that Hungarian matching performs non-trivial primitive correspondence estimation rather than simply preserving the original group ordering, we analyze the alignment assignments used in MoP construction. Across 28 transformer layers and two non-reference experts per layer, Hungarian

Table 7: Optimization settings and reported logical residual counts for 1-M continuation. Each run uses 1,000 steps, batch size 512, DFHR, pretrained/router learning-rate multipliers 0.1/1.0, and 0.578B active residual parameters. Random MoP uses three seeds for topology sampling. Training and calibration seed settings are given in Appendix E. Reductions are relative to 1.734B disjoint ownership; counts are rounded.

Method	Base LR	Expert mult.	λ_{dist}	$P_{\text{total}}^{\text{res}}$	Reduction
Continued disjoint	3×10^{-5}	0.2	1	1.734B	0.0%
Random MoP	3×10^{-5}	0.2	1	1.414B	18.5%
Fixed Partial	3×10^{-5}	0.2	1	1.349B	22.2%
MoP	3×10^{-5}	0.2	1	1.414B	18.5%

Table 8: Ownership selection and fixed pair-sharing on the 1-M mixture. MoP and Random MoP use the matched protocol in Section 4.4; Random MoP scores are means over three topology seeds. Fixed Partial shares their active residual width and recovery settings, with logical parameter counts given in Table 7. Metrics are percentages on RefCOCO val and NYUv2 depth estimation.

	RefCOCO val		NYUv2	
Method	cIoU↑	gIoU↑	AbsRel↓	δ_1 ↑
Random MoP	38.682	38.397	10.70	88.50
Fixed Partial	27.215	27.675	11.33	87.44
MoP	62.158	63.347	10.29	88.85

matching produces 55 out of 56 non-identity expert-to-reference assignments. At the primitive level, 175 out of 224 group correspondences are mapped to different reference-group indices. These results indicate that primitive correspondence after expert specialization generally differs from the original channel-group ordering, motivating explicit alignment before parameter sharing.

We emphasize that this analysis characterizes the structural effect of alignment, i.e., whether Hungarian matching changes primitive correspondence. It does not isolate the individual contribution of alignment to downstream performance, which would require a dedicated no-alignment ablation.

G.5 MoP Calibration Procedure

The MoP topology induction uses calibration data sampled from the same mixed multimodal training pool used for Stage-1 specialization. Calibration examples are sampled using a fixed seed and bucketed reservoir sampling based on task metadata.

Table 9: Calibration statistics for MoP construction.

Stage	Selected samples	Valid samples	Purpose
Response alignment	256	195	Hungarian matching
Update signature extraction	128	109	Ownership induction

The calibration loader uses seed 42 and performs bucketed sampling according to task and dataset metadata. The sampled examples are drawn from the 1M mixed training pool rather than validation benchmarks.

For alignment, routed visual responses are collected and used to compute primitive-level response similarity. For ownership induction, first-order gate/up update signatures are computed from calibration gradients.

The induced topology is stored explicitly as a topology artifact rather than being regenerated during recovery training.

MoP calibration provenance. The final MoP topology is constructed using the same calibration pipeline as the controlled topology studies. Calibration examples are sampled from the mixed Stage-

1 training pool rather than evaluation benchmarks. The dataset sampler uses a bucketed reservoir strategy with random seed 42. Response alignment uses 256 sampled examples, with 195 completing the alignment forward pass. Update-signature extraction uses a separate 128-example calibration draw, of which 109 samples produce valid image-loss gradients and contribute signatures. The resulting alignment and signature artifacts are stored explicitly and reused during topology construction. The final Row-8 model differs from learned-topology ablations only in the topology selection rule, while sharing the same calibration source and intermediate artifacts.

H Task Rendering and Standard Evaluation Protocols

The final Vision-Cheers checkpoint and the controlled continuation models share the referring- and instance-segmentation protocols below. Depth ablations use NYUv2, with dataset-specific evaluation settings described in Appendix H.2.

Generation protocol. Segmentation predictions use the image-generation target branch at 512×512 with 50 steps, text/image guidance scales 8.0/2.0, and $\alpha = 0.7$. The DIODE benchmark uses 512×512 predictions, 80 steps, guidance scales 7.5/1.5, and $\alpha = 0.3$. The sampling-schedule coefficient α maps time τ to $\tau' = \alpha\tau/[1 + (\alpha - 1)\tau]$.

H.1 Referring Segmentation

The standard RefCOCO validation evaluation traverses all 3,811 referring-query records in `refseg_refcoco_unc_val.jsonl`, with split `val`, no random subsampling, and evaluation limit 0 (unrestricted). Each JSONL record corresponds to one query. Distributed evaluation partitions the records deterministically and writes predictions as `<id>.png`. Both the final checkpoint and the controlled continuation models use preprocessing pipeline `refseg_t192_c0` and `benchmarks/RefSeg/eval_refseg.py`. The completed evaluation reports zero missing predictions.

The prompt requests a white foreground object on a black background; inference uses maximum sequence length 120. Ground-truth masks are rasterized from COCO annotations, with positive values denoting foreground. RGB predictions are averaged across channels and thresholded strictly above 192 on the 0–255 scale, with no small-component removal. Binary predictions are resized to the ground-truth dimensions by nearest-neighbor interpolation. For N evaluated queries with predicted and ground-truth foreground sets $\hat{Y}_n, Y_n$, cumulative IoU in percent is

$$\mathrm{cIoU} = 100 \times \frac{\sum_{n=1}^{N} |\hat{Y}_n \cap Y_n|}{\sum_{n=1}^{N} |\hat{Y}_n \cup Y_n|}. \tag{26}$$

The evaluator's gIoU is the arithmetic mean of query IoUs: $\mathrm{gIoU} = 100N^{-1} \sum_{n=1}^{N} |\hat{Y}_n \cap Y_n| / |\hat{Y}_n \cup Y_n|$. For an empty ground-truth mask, query IoU is one only when the prediction is also empty, and zero otherwise. RefCOCO validation contains no such negative queries. The evaluator skips missing predictions and excludes them from N, recording their count alongside evaluated and size-corrected predictions. The completed RefCOCO evaluation has zero missing predictions, so all 3,811 manifest queries contribute to the metrics.

H.2 Monocular Depth Evaluation

NYUv2 evaluation protocol. The controlled continuation, ownership, and routing comparisons reported in Tables 2 and 8 use NYUv2 depth evaluation and report AbsRel and δ_1 in percent. All variants share the same NYUv2 evaluation settings.

We evaluate on the 654-image NYUv2 Eigen test split. Raw 16-bit depth maps are decoded from millimeters to meters. Valid pixels satisfy $10^{-3} < d < 10$ m and are further restricted by the Eigen crop (rows 45:471 and columns 41:601). Predictions are resized to the 640×480 ground-truth resolution using nearest-neighbor interpolation. Predicted outputs are interpreted as relative disparity before per-image alignment. Following the relative-depth evaluation protocol, each prediction is independently aligned to inverse ground-truth depth using least-squares scale and shift estimation in disparity space. The aligned disparity is converted back to depth and clipped to the valid range.

AbsRel and δ_1 are computed per image over valid pixels and averaged across images. No hole-filled ground truth or additional filtering pipeline is used.

DIODE evaluation protocol. The DIODE evaluation in the final-checkpoint benchmark suite uses the complete `diode_val.jsonl` manifest with `env=all`, containing 771 records (325 indoor and 446 outdoor images). The protocol uses `benchmarks/DepthBench/eval_depth.py`, least-squares scale-and-shift alignment, and no additional preprocessing pipeline. The completed evaluation covers all 771 records with zero missing predictions.

Valid pixels are determined by the provided mask with threshold above 0.5 and finite positive ground-truth depth values; images with fewer than 100 valid pixels are skipped. Predicted RGB channels are averaged and divided by 255 to obtain a relative-depth scalar. Predictions are resized to the ground-truth resolution using nearest-neighbor interpolation. A least-squares scale and shift are fitted on valid pixels in depth space, followed by clipping predictions to $[0.6, 80]$ m.

For image n, let V_n denote valid pixels, $\hat{d}_{n,j}$ the aligned and clipped prediction, and $d_{n,j}$ the ground truth. The two reported metrics are

$$\mathrm{AbsRel}_n = \frac{1}{|V_n|} \sum_{j \in V_n} \frac{|\hat{d}_{n,j} - d_{n,j}|}{d_{n,j}}, \qquad \delta_{1,n} = \frac{1}{|V_n|} \sum_{j \in V_n} \mathbf{1}\left[\max\left(\frac{\hat{d}_{n,j}}{d_{n,j}}, \frac{d_{n,j}}{\hat{d}_{n,j}} \right) < 1.25 \right]. \quad (27)$$

Both metrics are averaged equally across images and multiplied by 100 for reporting. The result keys `abs` and `delta1_acc` correspond to these quantities. The evaluator skips missing or failed predictions rather than assigning zero. The completed DIODE evaluation contains no missing predictions and evaluates all 771 images under the shared protocol.

I Partition Induction and Analysis Protocols

The MoP topology analysis uses the seed-42 Vision-Cheers-1M checkpoint with $K = 3$, $G = 4$. The topology is induced using the calibration protocol described in Appendix E. Thresholds are -0.02 for full sharing and 0.02 for pair sharing, with at most two shared groups per layer. The analysis model uses the common initialization procedure in Appendix D, followed by 1,000 recovery steps with fixed Φ and $\lambda_{\mathrm{dist}} = 1$. Table 10 lists every saved recovery ownership decision in Φ used in the controlled continuation analysis. The MoP topology is constructed from the corresponding Stage-1 specialization checkpoint using the same alignment and ownership induction pipeline described above. The resulting ownership topology is fixed during Stage-2 recovery.

Parameter counts in the controlled continuation setting. The Vision-Cheers-1M analysis model has 3,985,570,313 unique parameters: 2,337,964,553 non-residual weights excluding embeddings, 233,373,696 tied token-embedding weights, 129,024 router weights, and 1,414,103,040 unique residual weights. Replacing its unique residual count by one 578,027,520-parameter Top-1 path gives 3,149,494,793 active parameters. Restoring all disjoint residual copies gives 4,305,549,833 unique model parameters. The difference is 319,979,520 parameters, or 7.432% of the disjoint whole-model count, corresponding to an 18.5% residual-only reduction. These topology-specific counts underlie the controlled comparison and ownership analysis.

Analysis definitions. Let d_g be the width of group g, so $\sum_g d_g = d_{\mathrm{op}}$. Define $\mathrm{scope}(\phi)$ as full, subset, or specific when $|\phi| = 1$, $1 < |\phi| < K$, or $|\phi| = K$, respectively. The width-weighted fraction of scope s at layer ℓ is

$$Q_s^\ell = \frac{1}{d_{\mathrm{op}}} \sum_{g=1}^{G} d_g \mathbf{1}[\mathrm{scope}(\phi_{\ell,g}) = s]. \quad (28)$$

Let $\kappa_{\ell,g}^{\mathrm{gu}}$ count one gate/up group and $\kappa_{\ell,g}^{\mathrm{down}}$ one expert-specific down group, excluding the router. For the gate/up-only topology,

$$P_{\mathrm{total}}^{\mathrm{res}} = \sum_{\ell \in \mathcal{L}_r} \sum_{g=1}^{G} \left(|\phi_{\ell,g}| \kappa_{\ell,g}^{\mathrm{gu}} + K \kappa_{\ell,g}^{\mathrm{down}} \right), \quad P_{\mathrm{active}}^{\mathrm{res}} = \sum_{\ell \in \mathcal{L}_r} \sum_{g=1}^{G} \left(\kappa_{\ell,g}^{\mathrm{gu}} + \kappa_{\ell,g}^{\mathrm{down}} \right). \quad (29)$$

Table 10: Ownership partitions of the Vision-Cheers-1M analysis model. All layer, group, and expert indices are one-based. Each entry uses the partition notation in Sec. 3.4.2; $1|2|3$ means independent gate/up copies. Down projections remain expert-specific.

Layer	$g=1$	$g=2$	$g=3$	$g=4$
1	123	1|2|3	123	1|2|3
2	123	1|2|3	1|2|3	23|1
3	1|2|3	123	1|2|3	123
4	1|2|3	123	123	1|2|3
5	1|2|3	13|2	1|2|3	123
6	1|2|3	123	123	1|2|3
7	1|2|3	1|2|3	12|3	123
8	1|2|3	1|2|3	123	123
9	123	123	1|2|3	1|2|3
10	1|2|3	123	1|2|3	13|2
11	1|2|3	123	1|2|3	1|2|3
12	13|2	12|3	1|2|3	1|2|3
13	123	1|2|3	1|2|3	12|3
14	1|2|3	1|2|3	23|1	123
15	12|3	1|2|3	123	1|2|3
16	1|2|3	123	1|2|3	12|3
17	13|2	1|2|3	123	1|2|3
18	1|2|3	123	1|2|3	123
19	13|2	1|2|3	123	1|2|3
20	1|2|3	123	123	1|2|3
21	123	1|2|3	123	1|2|3
22	123	1|2|3	1|2|3	123
23	123	1|2|3	123	1|2|3
24	1|2|3	123	123	1|2|3
25	23|1	123	1|2|3	1|2|3
26	1|2|3	123	1|2|3	123
27	1|2|3	1|2|3	123	123
28	1|2|3	1|2|3	1|2|3	1|2|3

Let P_{base} include the non-residual model and routers. Then $P_{\text{active}} = P_{\text{base}} + P^{\text{res}}_{\text{active}}$, $P_{\text{total}} = P_{\text{base}} + P^{\text{res}}_{\text{total}}$, and fully materializing every expert gives $P_{\text{materialized}} = P_{\text{base}} + KP^{\text{res}}_{\text{active}}$. Here P_{total} counts distinct parameters under shared ownership; $P_{\text{materialized}}$ counts tensor elements after expanding shared expert references, with the same non-residual representation. The total logical residual-parameter reduction from disjoint ownership is

$$R_{\text{res}} = 1 - \frac{P^{\text{res}}_{\text{total}}}{KP^{\text{res}}_{\text{active}}}. \tag{30}$$

Let $B_{\ell,g}(k)$ denote the block containing expert k. Their width-weighted gate/up ownership overlap is

$$\Omega^{\ell}_{k,k'} = \frac{1}{d_{\text{op}}} \sum_{g=1}^{G} d_g \mathbf{1}[B_{\ell,g}(k) = B_{\ell,g}(k')]. \tag{31}$$

Layerwise ownership and overlap. For descriptive summaries, layers 1–9, 10–19, and 20–28 form bands of 9, 10, and 9 layers. Their full/pair/specific counts are $(15, 3, 18)$ out of 36 groups, $(10, 9, 21)$ out of 40, and $(15, 1, 20)$ out of 36. Fractions in Table 11 use these distinct denominators. Expert-overlap entries $(\bar{\Omega}_{1,2}, \bar{\Omega}_{1,3}, \bar{\Omega}_{2,3})$ are $(0.4444, 0.4444, 0.4444)$, $(0.3500, 0.3500, 0.2750)$, and $(0.4167, 0.4167, 0.4444)$ for the three bands. Full-model averages weight all 28 layers equally rather than weighting the three bands equally. Equal group widths make count fractions equivalent to width-weighted fractions. The 53 non-specific groups satisfy the layer-wise sharing budget $b = 2$, which allows at most $28 \times 2 = 56$ shared groups; compatibility selects their membership and ownership scope within this budget.

Table 11: Primitive ownership structure in the controlled continuation setting. Left: gate/up ownership fractions (%) by layer band. Right: shared gate/up width for each expert pair, expressed as $\bar{\Omega}$ averaged over all 28 layers on a 0–1 scale. Full, Subset, and Route-specific denote fully shared, subset-shared, and route-specific ownership. Figure 3 shows their layer–primitive positions.

Layerwise ownership (%)

Layers	Full	Subset	Route-specific
1–9	41.67	8.33	50.00
10–19	25.00	22.50	52.50
20–28	41.67	2.78	55.56
All	35.71	11.61	52.68

Expert overlap $\bar{\Omega}$

	E_1	E_2	E_3
E_1	1.0000	0.4018	0.4018
E_2	0.4018	1.0000	0.3839
E_3	0.4018	0.3839	1.0000

Table 12: Group-granularity scan at layer 13 on a fixed token pool. $G = 1$ is the whole-expert reference; D measures the relative pairwise correspondence after decomposition.

G	Group width	E_1–E_2	E_1–E_3	E_2–E_3	D
1	4480	0.212	0.279	0.112	–
2	2240	0.205	0.273	0.110	0.069
4	1120	0.195	0.263	0.107	0.067
5	896	0.191	0.256	0.104	0.065
8	560	0.179	0.245	0.100	0.066
10	448	0.174	0.238	0.096	0.064

The shared gate/up width fraction $\Omega^{\ell}_{k,k'}$ (Equation 31) averages to $\bar{\Omega}_{1,2} = \bar{\Omega}_{1,3} = 0.4018$ and $\bar{\Omega}_{2,3} = 0.3839$ across layers. These values quantify structural ownership overlap. Expert-specific down projections map the shared features into each route's residual response.

Parameter efficiency. We quantify parameter efficiency using P_{active}, P_{total}, and the logical residual-parameter reduction R_{res} in Equation 30, at unchanged active residual width and nominal inference FLOPs.

Group-granularity scan. With $K = 3$ fixed, Table 12 varies the within-expert group count G at layer 13 on a fixed token pool. The pairwise columns report mean per-group response cosine, distinct from the update compatibility in Equation 15. Let $A_G(k, k')$ denote this response statistic for experts k, k'; the relative correspondence margin is $D = A_G(1, 3) - A_G(1, 2)$. The reported margins range from 0.064 to 0.069 for $G = 2$–10. At the operating point $G = 4$, each group has 1,120 channels and the E_1–E_3 cosine is 0.263, retaining approximately 94.3% of the whole-expert value 0.279.

Group alignment and signatures. Alignment and signature extraction use the calibration protocol in Appendix E. Output responses determine group correspondence. Candidate experts receive common token states and upstream residual gradients, and Equation 3 concatenates only their gate/up gradient blocks for ownership selection. Recovery binds gate/up parameters while leaving each aligned down projection independent.

Continuation runs. The continued-disjoint, random, fixed-partial, and MoP runs use the settings in Table 7. Each retains its ownership structure for 1,000 training steps while adapting its parameters and router.

Analysis scope and reproducibility. We analyze calibration-set compatibility and the saved seed-42 MoP topology, including all 112 layer-group assignments, ownership counts, and unique residual parameters before and after tying. Calibration and recovery settings are given in Appendix E, and selector hyperparameters in Appendix D. The routing examples below provide qualitative illustrations of this model.

I.1 TASK-CONDITIONED HARD-ROUTING VISUALIZATION

Figures 4–5 qualitatively illustrate how the Vision-Cheers-1M analysis model uses its fixed expert-level routes during generation. The same conditioning image is paired with instructions for referring segmentation, category-conditioned instance segmentation, and monocular depth estimation.

For each visual token and expert, the heatmap records the fraction of 50 denoising steps selecting that expert. All maps use a 16×16 visual-token grid, a common $[0, 1]$ color scale, and one-based layer and expert indices. The visualizations use 50 steps for each task, while the DIODE benchmark protocol uses 80 steps.

I.2 HUNGARIAN ALIGNMENT STATISTICS

MoP performs Hungarian matching to establish primitive correspondence across independently specialized experts before ownership induction. To examine whether this alignment merely preserves the original group ordering, we report the resulting permutations across all routed layers.

We use expert 0 as the reference expert and align the other two experts to it. There are 28 routed layers, resulting in 56 expert-to-reference assignments. An assignment is considered identity only when the complete primitive permutation equals the original ordering $[0, 1, 2, 3]$.

Table 13: Hungarian alignment statistics.

Statistic	Value
Routed layers	28
Reference expert	Expert 0
Aligned experts	Expert 1, Expert 2
Expert-level assignments	56
Identity assignments	1
Non-identity assignments	55
Non-identity rate	98.21%
Primitive correspondences	224
Moved primitive groups	175
Primitive movement rate	78.13%

The complete expert-level assignment statistics are shown below.

Table 14: Expert-to-reference Hungarian assignment statistics.

Expert pair	Identity	Non-identity	Non-identity rate
E0–E1	0/28	28/28	100.00%
E0–E2	1/28	27/28	96.43%
Overall	1/56	55/56	98.21%

These statistics show that Hungarian alignment substantially changes primitive correspondence after specialization rather than simply recovering the original group ordering. This analysis verifies the structural effect of alignment; the contribution of alignment itself is evaluated separately through ownership induction experiments.

J TRAINING DATA COMPOSITION

This appendix details the final-model data pools summarized in Table 4. The approximately 38M corpus has aggregate stream statistics of 38.27M entries (Table 17); its 1,770,538-record targeted subset supplies Stage-2 continuation. Appendix J.4 gives final-model optimization settings, while Appendix E describes the ablation study.

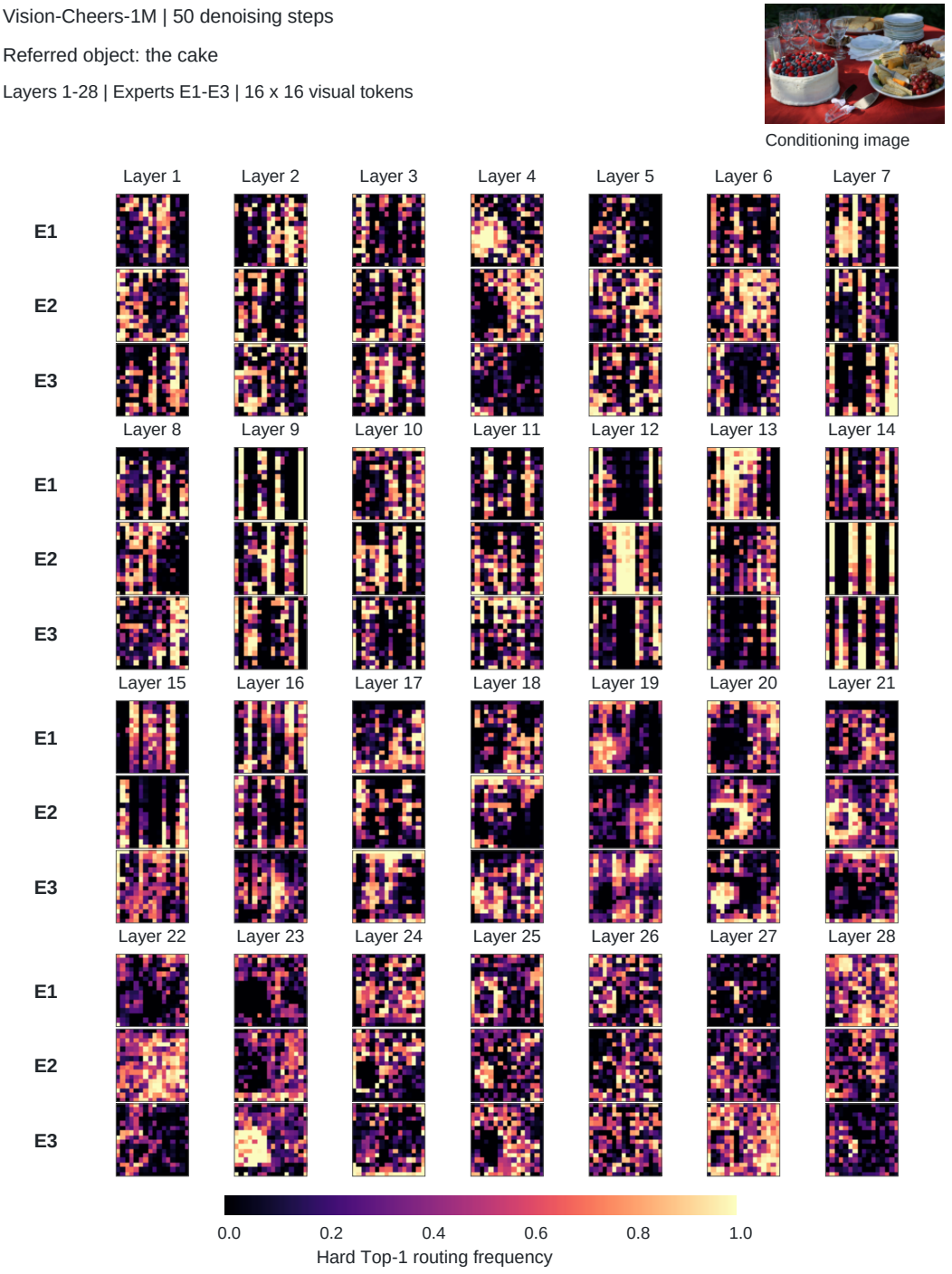

Figure 4: **Referring segmentation: qualitative hard Top-1 routing in the Vision-Cheers-1M analysis model.** Each map gives the fraction of 50 denoising steps selecting one expert at each visual token. All 28 layers and three experts are shown with one-based indices and a common $[0, 1]$ scale.

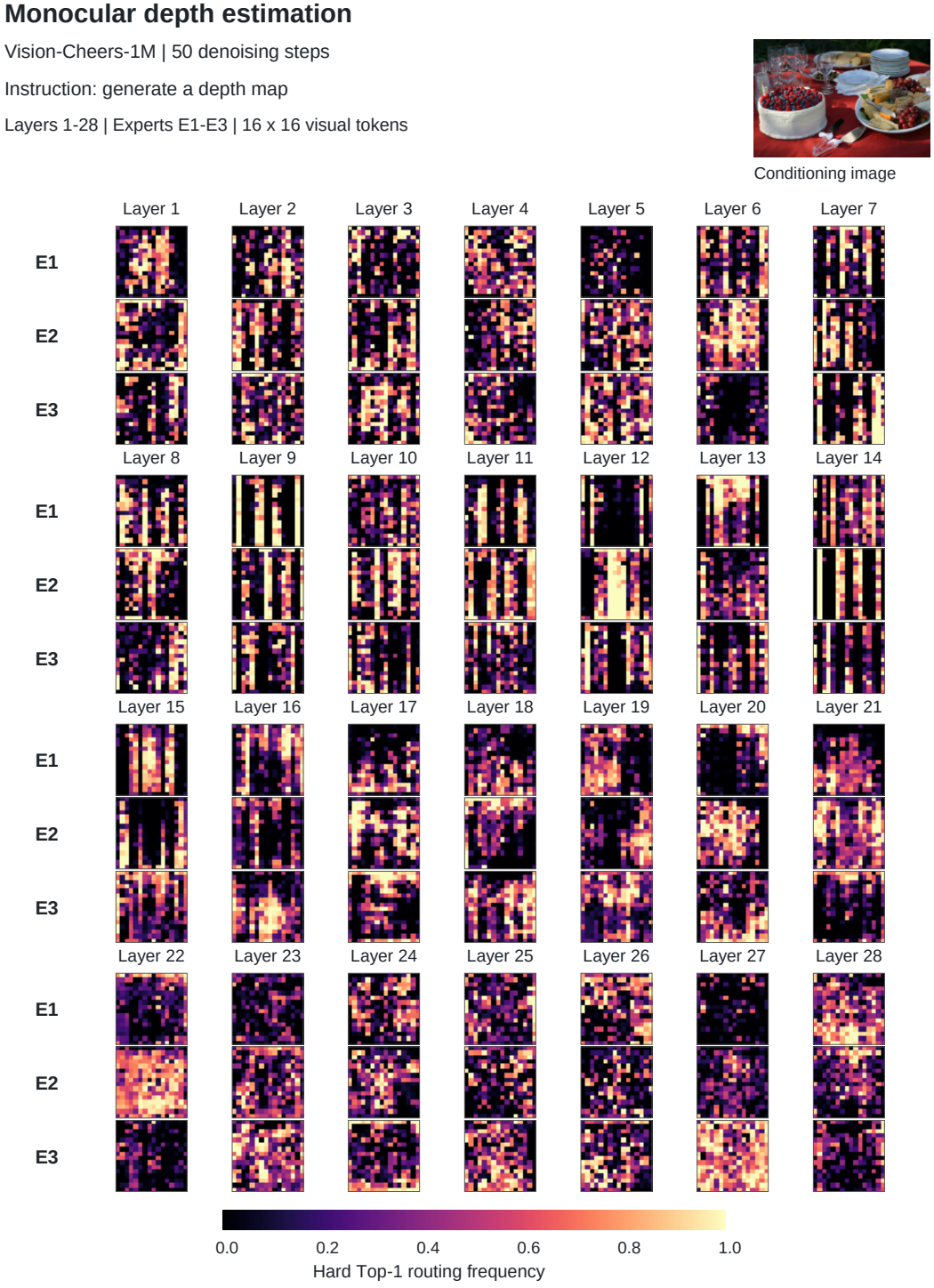

Figure 5: **Monocular depth estimation: qualitative hard Top-1 routing in the Vision-Cheers-1M analysis model.** Each 16×16 map shows expert-selection frequency over 50 denoising steps, with one-based layer and expert indices and a common $[0, 1]$ scale.

J.1 Training Stream Composition

The final Vision-Cheers training stream contains 38.27M training records. We use the term "record" because the count refers to JSONL training entries rather than unique images or unique annotations. Multiple records may correspond to the same image with different tasks, prompts, or annotations.

Table 15: Composition of the 38.27M training stream.

Component	Number of records
Main multimodal mixture	28,268,990
Replay stream	10,000,000
Total training records	38,268,990

The replay component is a fixed appended record stream rather than an online experience replay mechanism. It contributes 10M additional training records while preserving the original task mixture.

Table 16: Task composition of the final training stream.

Task category	Records
Understanding	2,758,521
Generation	2,758,460
Referring segmentation / grounding	10,682,039
Instance segmentation	11,035,092
Depth estimation	11,034,878
Total	38,268,990

Exact JSON-level duplicate analysis shows that the training stream contains repeated records introduced by the construction procedure. Therefore, the 38.27M number should be interpreted as training-record occurrences.

J.2 Stage 1: General Multimodal Training

The Stage-1 stream combines a 28.27M main pool and a 10M replay stream. The main portion approximately follows repeated cycles with the nominal understanding/generation/referring-segmentation/instance-segmentation/depth ratio $1 : 1 : 4 : 4 : 4$. Replay and a small residual tail produce the aggregate distribution in Table 17; the complete stream does not strictly follow the nominal cycle ratio. The replay stream reuses selected samples from its replay pool.

J.3 Stage 2: Targeted Dense-Prediction Subset

The Stage-2 sampling pool contains 1,770,538 records drawn from the approximately 38M corpus, organized into 126,467 deterministic cycles of 14 samples each (Table 18). Every cycle contains one understanding sample, one generation sample, and four samples from each of the three dense-prediction task families, giving the exact ratio $1 : 1 : 4 : 4 : 4$. Targeted continuation increases exposure to dense prediction while retaining understanding and generation examples.

J.4 Final-Checkpoint Training Procedure

Stage 1: Disjoint residual-expert specialization. Stage 1 trains these disjoint experts and their router on the multimodal corpus for 48,196 optimization steps with global batch size 500. The learning-rate schedule includes levels of 1×10^{-4} and 5×10^{-5}, a constant phase at 2×10^{-5}, and a subsequent cosine phase with a reported learning-rate value of 5×10^{-6}. The resulting Stage-1 specialization checkpoint supplies the source experts for MoP construction and frozen recovery teachers.

Table 17: Stage-1 general multimodal training stream. Sample counts and fractions are rounded; displayed task fractions sum to 99.9% because of rounding.

Task	Samples	Fraction
Multimodal understanding	2.76M	7.2%
Image generation	2.76M	7.2%
Referring segmentation	10.68M	27.9%
Category-conditioned instance segmentation	11.04M	28.8%
Monocular depth estimation	11.03M	28.8%
Total	38.27M	100%

Table 18: Stage-2 sampling-pool composition within the approximately 38M corpus. Counts are exact and sum to 1,770,538 records across 126,467 cycles.

Task	Samples	Samples per cycle
Multimodal understanding	126,467	1
Image generation	126,467	1
Referring segmentation	505,868	4
Category-conditioned instance segmentation	505,868	4
Monocular depth estimation	505,868	4

Stage 2: MoP construction and recovery. Starting from the Stage-1 specialization checkpoint, offline construction applies response alignment, gate/up update compatibility, and the hierarchical threshold selector. Selected gate/up weights are consolidated and bound while each route retains its aligned down projection. Fixed-ownership recovery then runs for 1,000 steps with global batch size 500 and learning rate 2×10^{-5}, giving 500,000 sample presentations drawn from the pool in Table 18. The resulting model is the final Vision-Cheers checkpoint.

K Qualitative Results

Figures 6 and 7 are qualitative outputs from the checkpoints reported in the main tables. Referring segmentation uses the 38-M training Vision-Cheers. The depth figure compares Vision-Cheers and Vision-Bagel on NYUv2.

K.1 Referring Expression Segmentation

Each row is one referring expression. Prediction is overlaid in red and ground truth in green. "Girl with mic" selects the child holding the microphone and leaves the surrounding children out. "Teddy bear in front" keeps the nearer bear and excludes the one behind it. "Extreme right side of the train" restricts the mask to the rightmost car. Boundaries are coarser than the ground-truth contours.

K.2 Depth Estimation

Columns are the image, the prediction, and the ground truth. Brighter values are closer. Each depth map is scaled on its own, so the figure compares structure rather than metric range. Vision-Cheers keeps the chair in front of the desk and the piano body with its bench. Vision-Bagel gives sharper chair silhouettes, while the bathroom window becomes a saturated bright region.

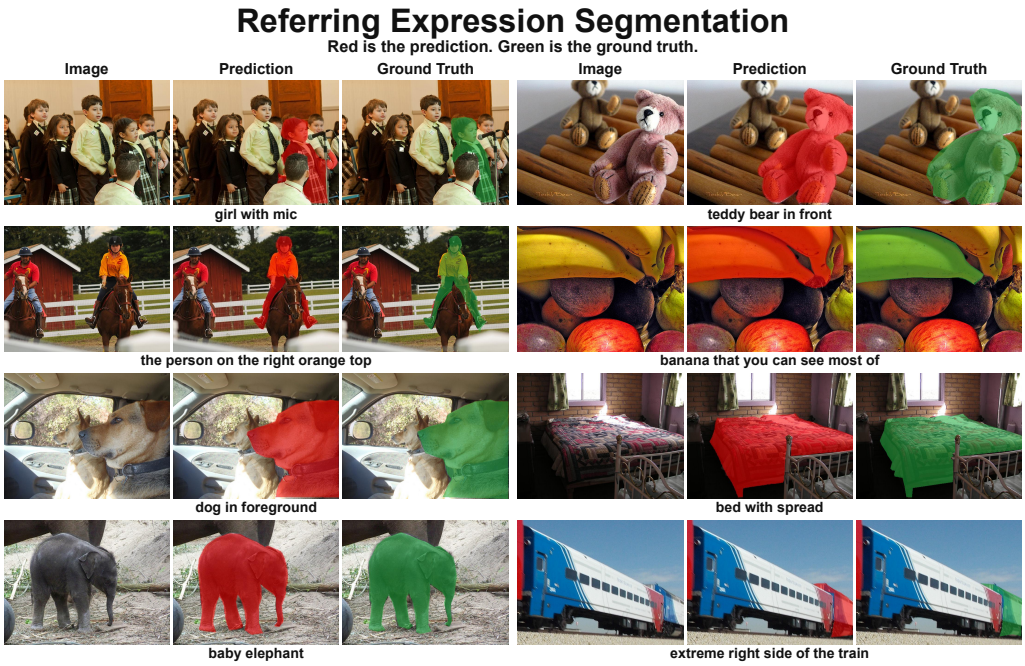

Figure 6: **Referring expression segmentation** on RefCOCO. Columns are the image, the prediction, and the ground truth.

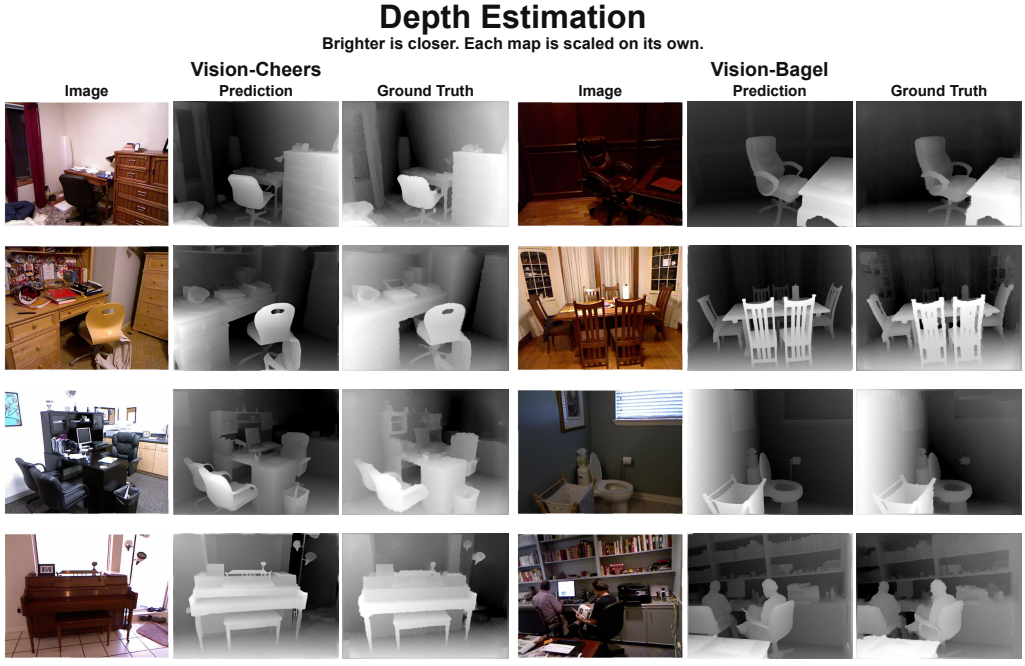

Figure 7: **Depth estimation** on NYUv2. The left four examples are Vision-Cheers and the right four are Vision-Bagel. Brighter means closer.