Beyond Encoder Accumulation: Measuring Encoder Roles in Multi-Encoder VLMs

Wei Ding
Tsinghua University
Beijing, China
teresading999@gmail.com

Yudong Zhang*
Tsinghua University, Tencent Hunyuan
Beijing, China
btzhangyudong@gmail.com

Abstract

As foundation models integrate increasingly diverse visual inputs, understanding how vision encoders work together during training becomes important for efficient model design. We study these interactions by retraining all combinations of five vision encoders and evaluating them on 16 benchmarks from four domains, using approximately 20,000 GPU-hours. We report three findings. First, retraining each combination changes how encoders are ranked compared with masking them in an already trained model, even changing which encoder ranks first. Second, we measure each encoder along two dimensions: *Capacity*, how well it performs alone, and *Necessity*, how much performance drops when it is removed and the remaining encoders are retrained. These measures lead to different choices. Pairing a strong encoder with one that adapts well during joint training performs better than simply choosing the two strongest individual encoders. In particular, CLIP and ConvNeXt close 96.8% of the performance gap between CLIP alone and the full five encoder model, using roughly half the encoder parameters. Third, we analyze the effective rank of encoder features before the projector to understand how their representations change during joint training. Across pairs, ConvNeXt's mean rank stays near its singleton value, while CLIP shows the largest mean increase. The best pair, CLIP+ConvNeXt, combines these two behaviors. Together, our results show how retraining, individual encoder contributions, and changes in learned representations can guide the selection of compact vision encoder combinations.

1 Introduction

Large Vision-Language Models (LVLMs) [20, 45] increasingly combine heterogeneous vision encoders to improve multimodal performance. Eagle [33], for example, fuses ConvNeXt [23], EVA-02 [3], CLIP [30], Pix2Struct [16], and SAM [13] through channel concatenation. Other architectures, including Cambrian-1 [36] and BRAVE [10], use cross-attention aggregation. As encoder pools grow, selecting a compact combination that preserves the benefits of the full pool becomes an important design problem.

A central question is how to measure each encoder's contribution under joint training. Wang et al. [40] introduce the Conditional Utilization Rate (CUR), which measures performance drops from masking encoder branches in a fixed Eagle-X5 checkpoint. This captures encoder utilization after training, while architecture selection depends on how the remaining encoders perform when retrained together. We study this distinction through a systematic comparison of Inference-time Masking (IM) and Training-time Removal (TR).

We retrain all 31 non-empty subsets of the five Eagle-X5 encoders under a unified pipeline and evaluate them on 16 benchmarks from Cambrian-1, using approximately 20,000 GPU-hours. IM

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and TR produce different encoder rankings, including the top-ranked encoder, and favor different two-encoder combinations. These differences directly affect encoder selection.

Our subset results show that a well-chosen pair captures most of the full model's performance. CLIP+ConvNeXt closes 96.8% of the CLIP-to-full score gap with roughly half the encoder parameters. Larger pools offer limited overall gains, although the full configuration retains an advantage on Vision-Centric tasks. To understand these results, we distinguish two aspects of encoder contribution: *Capacity*, its standalone score relative to the full model, and *Necessity*, the score drop after removing it and retraining the remaining encoders. These dimensions reveal distinct encoder roles. In particular, ConvNeXt serves as a *Universal Core* and CLIP as an *Adaptive Complement*, forming a stronger pair than the two highest-Capacity encoders.

We further examine pre-projector effective rank to characterize encoder representations under joint training. After accounting for pool size, effective rank remains associated with score. Across pairs, ConvNeXt's mean rank stays near its singleton value, while CLIP shows the largest mean increase. The best pair combines this anchor with a rank-expanding complement.

Our contributions are threefold:

1. **Retraining-based Attribution.** We evaluate all 31 encoder subsets and show that training-time removal changes encoder rankings and selection decisions relative to inference-time masking.
2. **Capacity–Necessity Decomposition.** We separate standalone capability from contribution within the full pool, identifying encoder roles and a compact-pool selection rule based on complementary roles.
3. **Representation-Level Analysis.** We use pre-projector effective rank to characterize encoder adaptation under joint training and its relationship to pair performance.

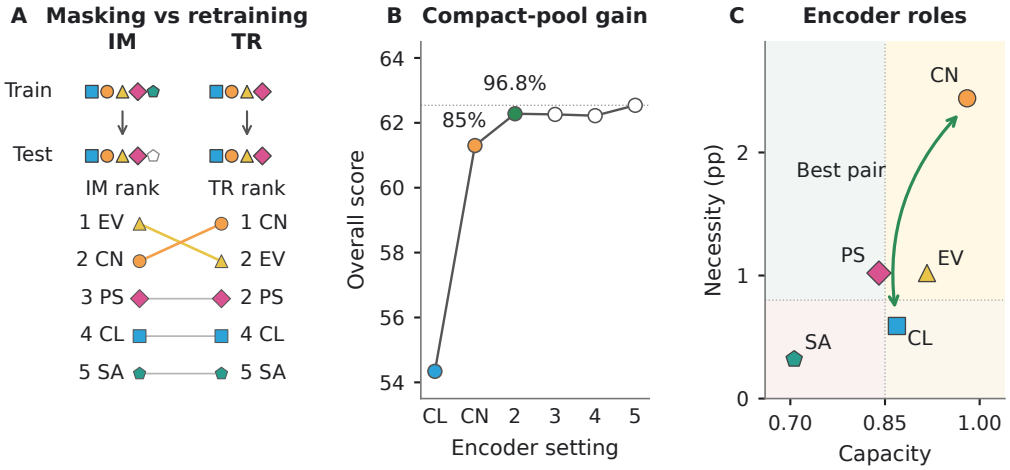

Figure 1: **Encoder selection depends on joint training.** (A) IM masks an encoder after training; TR removes it before retraining. The schematic removes SAM. Rank labels use leave-one-out drops; EVA-02 and Pix2Struct share second place under TR. (B) CL and CN are separate singletons; 2–5 denote best subsets by size. Labels give CLIP-to-full gap closure (54.34 to 62.54; dotted line). (C) Capacity and Necessity; the arrow connects the best pair. Background divisions are at Capacity 0.85 and Necessity 0.80 points. CN = ConvNeXt, EV = EVA-02, CL = CLIP, PS = Pix2Struct, SA = SAM.

2 Related Work

Multi-Encoder LVLMs. Multi-encoder LVLMs combine heterogeneous visual streams via three fusion mechanisms [33]: token-sequence concatenation [19, 2], cross-attention aggregation [18, 36, 10, 27], and channel-level concatenation [11, 9]; Tong et al. [36] benchmark a broad range of vision encoders. Routing methods activate encoders conditionally for each input [46, 15, 43], with Zhang et al. [43] showing that one routed encoder can match four fused ones. Distillation compresses a multi-encoder pool into one backbone [31, 6, 1, 39], raising the question of which teacher capabilities

to preserve. We use Eagle-X5 [33] because channel concatenation keeps the visual token count fixed across encoder subsets. Retraining every subset then lets us measure which encoders contribute to the trained model, informing teacher selection for distillation.

Encoder Attribution and Representation Analysis. The closest work on encoder attribution, Wang et al. [40], formalizes this for Eagle-X5 via Conditional Utilization Rate (CUR) and Information Gap (IG), computed by Inference-time Masking (IM) on a fixed checkpoint; earlier inference-time subset removal appears in BRAVE [10], and broader feature-attribution methods [5, 38, 35, 26] measure interventions on a fixed checkpoint. Compact-pool design instead requires a *training-time* counterfactual: how the model would perform if trained with the chosen subset. We retrain every non-empty subset and find that IM and TR rank a different encoder first; our Capacity/Necessity decomposition separates standalone performance from the retrained leave-one-out contribution. To interpret subset-level differences we measure each encoder's pre-projector representation through its effective rank [32], complemented by centered kernel alignment (CKA) [14] for comparing projector representations across trained subsets in Section 5.5. The Platonic Representation Hypothesis [8] suggests that models can converge toward shared representations. Here we examine how an encoder's representation changes with its partners.

3 Experimental Setup

3.1 Architecture and subset enumeration

Architecture. We build on Eagle-X5 [33], which pairs a Vicuna-7B decoder [44] with five vision encoders: ConvNeXt [23], EVA-02 [3], CLIP [30], Pix2Struct [16], and SAM [13]. Spatially aligned visual features are concatenated across channels and mapped to the language model input space by a shared two-layer multilayer perceptron; the number of input channels ranges from 1024 to 7680 depending on the active subset, while its output has 4096 channels.

Subset enumeration. We retrain all 31 non-empty encoder subsets under the same training recipe, covering singletons, pairs, and larger pools. A pool is the active encoder subset, and k is its number of encoders. The same models support the attribution comparison in Section 3.4, the Capacity and Necessity analysis in Section 4, and the effective rank analysis in Section 5.

Table 1 gives all subset scores and encoder parameter counts; Table 3 compares pairs. CL = CLIP, CN = ConvNeXt, SA = SAM, PS = Pix2Struct, EV = EVA-02. Encoder parameters exclude the projector and language model.

Training recipe. Pretraining uses LLaVA-Pretrain 558K; finetuning uses Eagle-1.8M SFT. Recorded configurations use AdamW for one epoch, learning rates 10^{-3}/2×10^{-5} and per-device batches 8/4 (pretrain/finetune), accumulation 1, cosine decay, 3% warmup, zero weight decay, bfloat16, and a 2048-token limit.

3.2 Benchmarks

We evaluate on 16 benchmarks in four families following the standard Cambrian-1 task grouping [36]. General comprises MME [4], MMBench-EN [21], SEED-Bench [17], and GQA [7]. Knowledge comprises ScienceQA [24], MMMU-val [42], MathVista [25], and AI2D [12]. OCR & Chart comprises OCRBench [22], ChartQA [28], TextVQA [34], and DocVQA [29]. Vision-Centric comprises MMVP [37], RealWorldQA [41], and the CV-Bench-2D and CV-Bench-3D splits [36]. Unless stated otherwise, the overall score is the unweighted average across these benchmarks. Before averaging, we put scores on a percentage scale by dividing MME by 28 for its 2800-point normalization and OCRBench by 10. Each family contains four tasks, so averaging the family scores gives the same overall score up to rounding.

3.3 Attribution protocols: IM and TR

We compare two methods for measuring per-encoder contribution. *Inference-time Masking* (IM) replaces one encoder's output features with zeros on a fixed full-pool checkpoint. We compute CUR, the relative score drop, from the zero-output masking scores in Wang et al. [40, Table 7].

Table 1: **Results for all encoder subsets.** Subsets are grouped by size and sorted by Overall. Bold Overall scores identify the best subset at each size. Relative score divides Overall by the full-pool score; parameters count active vision encoders in millions. Abbreviations are defined in Section 3.1.

Subset	k	General	Knowledge	OCR & Chart	Vision-Centric	Overall	Relative score	Encoder params (M)
Pool size $k = 5$								
CL+CN+SA+PS+EV	5	65.70	54.88	66.97	62.62	**62.54**	1.000	2274
Pool size $k = 4$								
CL+CN+PS+EV	4	65.48	55.19	66.44	61.76	**62.22**	0.995	1966
CN+SA+PS+EV	4	65.04	55.42	66.58	60.75	61.95	0.991	1970
CL+CN+SA+EV	4	65.14	54.49	65.90	60.58	61.52	0.984	1761
CL+CN+SA+PS	4	64.95	54.25	66.63	60.25	61.52	0.984	1970
CL+SA+PS+EV	4	64.90	55.11	61.00	59.41	60.10	0.961	1427
Pool size $k = 3$								
CL+CN+PS	3	65.56	55.50	67.12	60.88	**62.26**	0.996	1663
CL+CN+EV	3	65.58	54.79	66.39	60.91	61.92	0.990	1453
CN+SA+PS	3	65.27	54.82	66.65	59.93	61.67	0.986	1667
CL+CN+SA	3	64.87	54.61	66.26	60.74	61.62	0.985	1458
CN+PS+EV	3	65.62	54.34	65.81	60.70	61.62	0.985	1663
CN+SA+EV	3	64.70	54.93	65.86	60.49	61.49	0.983	1458
CL+PS+EV	3	64.93	54.22	60.74	60.14	60.01	0.960	1120
SA+PS+EV	3	64.01	54.01	59.41	60.84	59.57	0.953	1124
CL+SA+PS	3	64.08	54.26	60.17	58.86	59.34	0.949	1124
CL+SA+EV	3	64.83	54.65	54.10	60.34	58.48	0.935	915
Pool size $k = 2$								
CL+CN	2	65.37	55.02	67.25	61.50	**62.28**	0.996	1150
CN+EV	2	65.06	54.72	66.39	60.93	61.77	0.988	1150
CN+PS	2	65.04	54.83	66.30	59.82	61.50	0.983	1359
CN+SA	2	64.97	54.41	65.36	60.92	61.41	0.982	1154
CL+PS	2	64.76	55.38	59.94	59.52	59.90	0.958	816
PS+EV	2	64.46	54.07	59.45	60.65	59.66	0.954	816
CL+EV	2	64.81	55.18	54.28	61.21	58.87	0.941	607
SA+EV	2	63.86	53.88	51.94	62.04	57.93	0.926	611
SA+PS	2	57.86	52.54	56.22	58.00	56.15	0.898	820
CL+SA	2	64.45	54.05	41.67	58.11	54.57	0.873	611
Pool size $k = 1$								
CN	1	64.93	54.63	66.27	59.37	**61.30**	0.980	846
EV	1	63.99	53.43	51.99	59.90	57.32	0.917	303
CL	1	64.81	54.79	40.68	57.07	54.34	0.869	304
PS	1	50.10	51.90	55.39	52.87	52.56	0.840	513
SA	1	56.29	52.84	11.86	55.63	44.16	0.706	308

Training-time Removal (TR) measures the score drop between the full model and the subset retrained without the encoder. Section 4 gives its formal definition as Necessity.

Table 2: **Best-performing subset at each pool size.** Relative score divides Overall by the full-pool score. Parameters count active vision encoders in millions, excluding the projector and language model.

Subset	k	Overall	Relative score	Encoder params (M)
ConvNeXt	1	61.30	0.980	846
CLIP + ConvNeXt	2	62.28	0.996	1150
CLIP + ConvNeXt + Pix2Struct	3	62.26	0.996	1663
CLIP + ConvNeXt + Pix2Struct + EVA-02	4	62.22	0.995	1966
CLIP + ConvNeXt + SAM + Pix2Struct + EVA-02	5	62.54	1.000	2274

3.4 Protocol audit: IM and TR lead to different design decisions

We compare the rankings and pool choices obtained from IM and TR. The published IM results use a different checkpoint and benchmark aggregation from our retrained subsets; the comparison concerns encoder-selection decisions across these evaluations.

IM and TR agree on much of the ranking but identify different anchors: EVA-02 ranks first under IM, whereas ConvNeXt ranks first under TR, as shown in Figure 2A. This changes the pair suggested by the individual rankings. The IM top-two heuristic selects ConvNeXt+EVA-02; retraining identifies CLIP+ConvNeXt as the best pair. Their complete family scores appear in Table 3.

The family-specific differences in Figure 2B show that this contrast also depends on the task family.

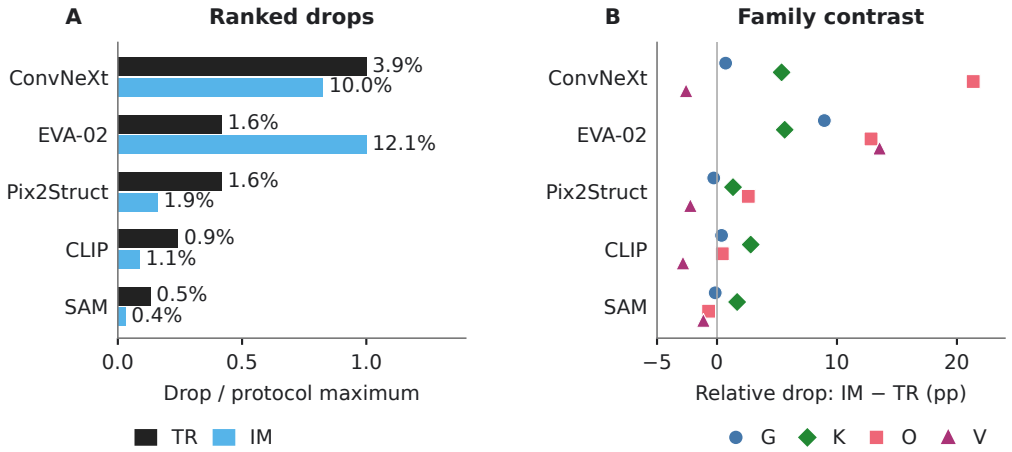

Figure 2: **Encoder attribution under masking and retraining.** (A) Relative leave-one-out drops scaled by each protocol's maximum; labels show unscaled percentages. EVA-02 and Pix2Struct tie under TR; Spearman $\rho = 0.82$ includes the tie. (B) IM minus TR relative drop by family, in percentage points; each drop is normalized by its own full-pool family score. IM CUR uses zero-output masking from Wang et al. [40, Table 7]. G = General, K = Knowledge, O = OCR & Chart, V = Vision-Centric.

4 Capacity and Necessity: Two Axes of Encoder Contribution

4.1 Two axes from the retrained subsets

We distinguish standalone performance from the contribution an encoder makes to the full pool. Let F denote the full encoder set and $\mathrm{score}(S)$ the mean benchmark score of the model trained with encoder subset S. Capacity is the encoder's singleton score relative to the full model:

$$\mathrm{Cap}(e) = \frac{\mathrm{score}(\{e\})}{\mathrm{score}(F)}. \tag{1}$$

Necessity is the score drop after retraining without that encoder:

$$\mathrm{Nec}(e) = \mathrm{score}(F) - \mathrm{score}(F \setminus \{e\}). \tag{2}$$

Necessity is measured in percentage points and corresponds to the TR comparison in Section 3.4. An encoder with high Capacity but low Necessity performs well alone but is partly substitutable within the pool. Capacity describes singleton training, while Necessity depends on the other encoders.

4.2 Per-encoder values, averaged and by family

ConvNeXt leads on both axes in Table 4; EVA-02 has high Capacity but moderate Necessity, and CLIP has moderate Capacity and the lowest Necessity among the top three. Family-specific scores distinguish the other encoders: EVA-02 has the highest Vision-Centric singleton score, while Pix2Struct has higher Capacity on OCR & Chart than on General.

Table 3: **Scores and family gains for all encoder pairs.** Pairs are sorted by Overall. Each Δ is the pair score minus the better constituent singleton score on that family, in percentage points. Highlighted rows compare CLIP+ConvNeXt with ConvNeXt+EVA-02.

Pair	Overall	ΔGeneral	ΔKnowledge	ΔOCR & Chart	ΔVision-Centric
CLIP+ConvNeXt	62.28	+0.44	+0.23	+0.98	+2.13
ConvNeXt+EVA-02	61.77	+0.13	+0.09	+0.12	+1.03
ConvNeXt+Pix2Struct	61.50	+0.11	+0.20	+0.03	+0.45
ConvNeXt+SAM	61.41	+0.04	-0.22	-0.91	+1.55
CLIP+Pix2Struct	59.90	-0.05	+0.59	+4.55	+2.45
Pix2Struct+EVA-02	59.66	+0.47	+0.64	+4.06	+0.75
CLIP+EVA-02	58.87	+0.00	+0.39	+2.29	+1.31
SAM+EVA-02	57.93	-0.13	+0.45	-0.05	+2.14
SAM+Pix2Struct	56.15	+1.57	-0.30	+0.83	+2.37
CLIP+SAM	54.57	-0.36	-0.74	+0.99	+1.04

Knowledge Capacity is high across encoders. This family is less sensitive to encoder choice, while General, OCR & Chart, and Vision-Centric better distinguish the encoders. Figure 3 shows these family profiles; pooled values are in Figure 1C and Table 4.

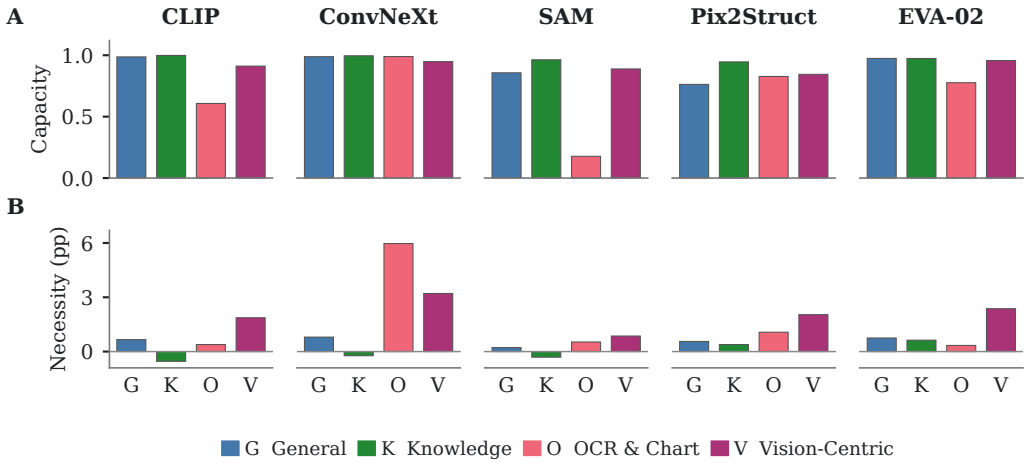

Figure 3: **Capacity and Necessity vary by task family.** (A) Singleton-to-full score ratios. (B) Retrained leave-one-out drops in percentage points. Negative values indicate that the subset scores above the full pool. The pooled view appears in Figure 1C.

4.3 Per-encoder roles and pair selection

Table 4 summarizes each encoder's role using its pooled Capacity and Necessity together with its family-specific scores. ConvNeXt is the Universal Core, with the highest pooled Capacity and Necessity. CLIP is an Adaptive Complement: its singleton Capacity is moderate, but it has the largest mean increase in pre-projector rank when paired with another encoder. EVA-02 is a Vision-Centric Specialist, with high singleton Capacity and the strongest singleton Vision-Centric score. Pix2Struct is a Niche Specialist concentrated on OCR & Chart, and SAM is Replaceable for the broad overall pool.

The best two-encoder model is CLIP+ConvNeXt rather than ConvNeXt+EVA-02, even though EVA-02 has higher singleton Capacity than CLIP; per-pair overall and per-family scores for all two-encoder subsets are listed in Table 3. Section 5 examines how each partner's effective rank changes in the trained pair.

4.4 Marginal contribution across subset sizes

We compare the highest-scoring subset at each pool size. Each subset is selected independently from all candidates of that size, rather than by extending the preceding pool.

Table 4: **Encoder roles from Capacity and Necessity.** Capacity is the singleton-to-full score ratio; Necessity is the retrained leave-one-out drop in percentage points (pp). Roles also reflect task-family profiles.

Encoder	Capacity	Necessity (pp)	Role
ConvNeXt	0.980	2.44	**Universal Core**
CLIP	0.869	0.59	**Adaptive Complement**
EVA-02	0.917	1.02	**Vision-Centric Specialist**
Pix2Struct	0.840	1.02	**Niche Specialist**
SAM	0.706	0.32	**Replaceable**

The largest gain comes from adding CLIP to ConvNeXt (Table 2). The best three- and four-encoder pools have similar overall scores at higher cost, while the full pool adds a small gain concentrated in Vision-Centric tasks, as shown in Figure 4.

CLIP+ConvNeXt closes 96.8% of the CLIP-to-full overall-score gap with roughly half the encoder parameters. Gap closure divides the score improvement over CLIP alone by the improvement from CLIP alone to the full pool. On OCR & Chart, it slightly exceeds the full pool. Family scores make the remaining design choice clear: the overall-best pair does not maximize every task family. Using 99% of the full-pool family score as a common target, Vision-Centric reaches that target at the full pool along the overall best-at-k sequence, but a family-specific choice, SAM+EVA-02, already reaches it with two encoders. Pool selection therefore depends on the intended workload as well as the parameter budget.

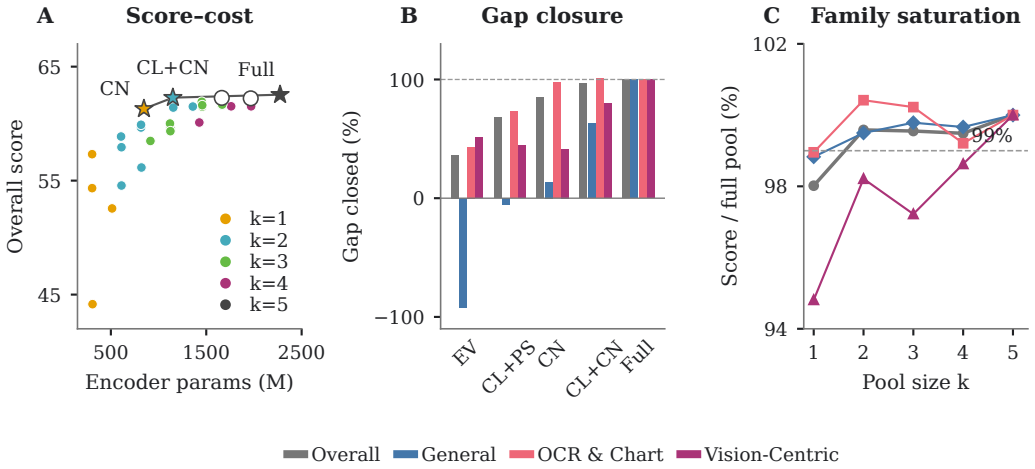

Figure 4: **A compact pair captures most of the full-pool gain.** (A) Score versus encoder parameters for all subsets. Stars mark non-dominated best-at-k models; hollow circles mark the other best-at-k models. (B) CLIP-to-full gap closure; Knowledge is omitted because its baseline gap is near zero. (C) Scores relative to the full pool along the overall best-at-k sequence; the dashed line marks 99%. CL = CLIP, CN = ConvNeXt, EV = EVA-02, PS = Pix2Struct.

5 Effective Rank and Encoder Composition

To examine why encoder pairs differ, we measure the effective rank of each encoder's representation at the shared projector input. For each task, we average visual tokens within each sampled example and stack these vectors by example. Centering each channel across examples gives the feature matrix X. With singular values s_i, we use the spectral-energy form of effective rank [32]:

$$r_{\text{eff}}(X) = \exp\left(-\sum_i p_i \log p_i\right), \qquad p_i = \frac{s_i^2}{\sum_j s_j^2}. \tag{3}$$

We compute this quantity separately for each encoder's channel slice, before the projector combines the features into the shared language input space. We average task-level ranks equally within each

family and across all tasks. Total rank sums these per-encoder ranks within a pool; mean rank divides that sum by pool size. Total rank is not the effective rank of the concatenated feature matrix. For pair comparisons, the higher-Capacity encoder is the anchor; partner rank change is the other encoder's rank in the trained pair minus its singleton rank.

5.1 From single encoders to larger pools

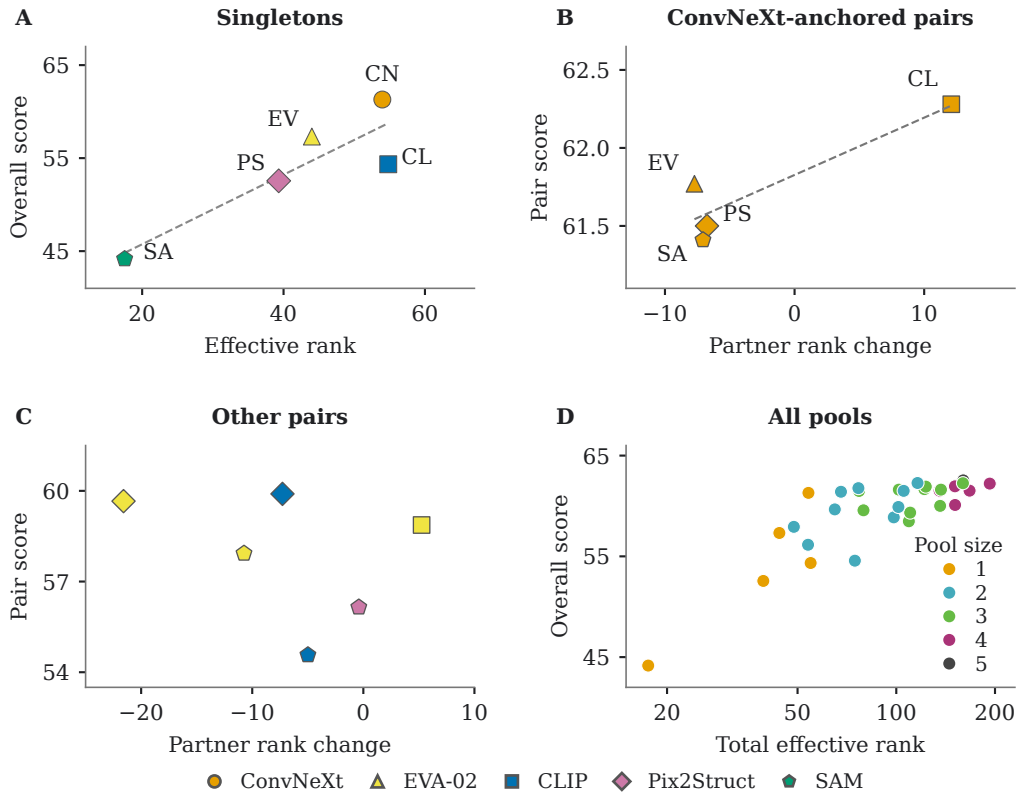

Figure 5: **Rank and score across encoder-pool sizes.** (A) Singletons. (B) ConvNeXt-anchored pairs, with an ordinary least squares fit; labels identify the partner. (C) The other six pairs. In B and C, the partner has lower Capacity; color identifies the anchor and marker shape the partner. Rank change is relative to singleton training. (D) All subsets, colored by pool size. Total rank uses a log axis; correlations use linear values. Numerical summaries appear in Section 5.1.

Effective rank separates three aspects of encoder composition, as shown in Figure 5. Among singletons, higher rank generally accompanies higher score. Among ConvNeXt-anchored pairs, the highest score comes from CLIP, the partner whose rank expands under joint training; the other partners' ranks contract. Thus, singleton strength and the representation learned with a partner provide different information for selecting a pool.

Larger pools separate rank from score. Along the best-at-k sequence, mean per-encoder rank peaks at the pair and then falls, while scores stay near the full-pool result. Across all subsets, total rank is positively associated with score, but pools with similar total rank can perform differently. Encoder identity matters alongside the amount of represented variation.

Ranks use 300 to 2,400 feature examples per task and are averaged across tasks before subsets. Table 5 shows that CLIP has the largest mean increase from singleton to pair and is the only encoder whose full-pool rank exceeds its singleton rank.

We use r for Pearson correlation, R^2 for the fitted coefficient of determination, and ρ for Spearman correlation. Singleton rank and score have $r = 0.886$. The four ConvNeXt-anchored pairs have an ordinary least squares slope of 0.0367 score points per rank dimension and $R^2 = 0.820$. Across all subsets, total rank and score have $r = 0.667$. After subtracting each pool-size group's mean from

score, total rank, and parameter count, score correlates with rank at $r = 0.421$ and with parameters at $r = 0.598$.

5.2 Effective rank and parameter count

Rank distinguishes similarly sized encoders such as CLIP, EVA-02, and SAM. Across the full subset set, both rank and parameter count remain associated with score after accounting for pool size, as reported in Section 5.1. Rank describes the learned representation; parameter count describes model size. Neither alone identifies the best partner.

5.3 Effective rank varies by task family

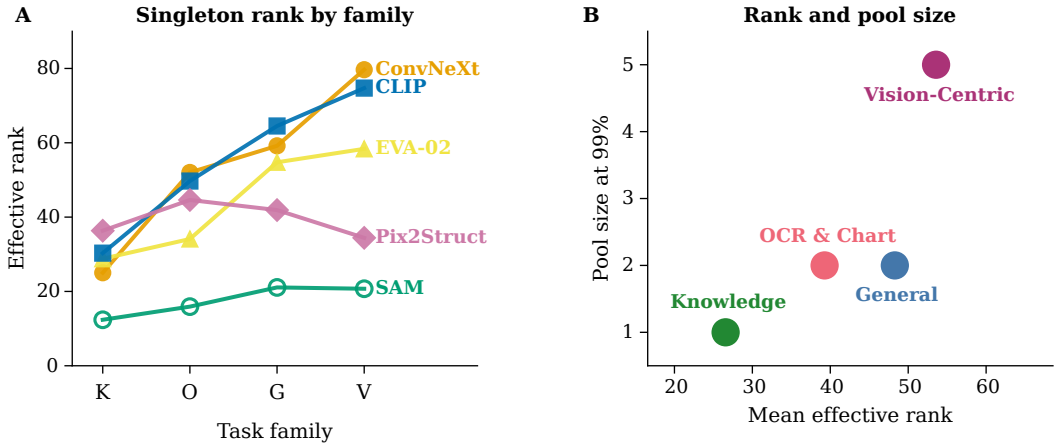

Figure 6: **Effective rank varies by task family.** (A) Singleton ranks, with families ordered by their mean. G = General, K = Knowledge, O = OCR & Chart, V = Vision-Centric. (B) Mean rank across singleton encoders versus the first pool size reaching 99% of the full-pool family score along the overall best-at-k sequence. This threshold crossing refers to that sequence, not the minimum over all subsets.

Knowledge has the lowest mean rank and Vision-Centric the highest; Pix2Struct instead peaks on OCR & Chart, as shown in Figure 6A.

The overall best-at-k sequence reaches the General and OCR & Chart targets with a pair, whereas its Vision-Centric score benefits from a larger pool. As the family-specific pair in Section 4 shows, this pattern motivates choosing partners for the target task, rather than assigning a required pool size from rank alone.

5.4 An anchor and an adaptable complement

Across pairings, ConvNeXt's mean rank stays near its singleton value, while CLIP shows the largest mean increase in Table 5. The best pair thus combines a stable anchor with a partner whose rank expands under joint training. In the full pool, ConvNeXt's rank falls from 54.0 alone to 27.5 with all five encoders, while CLIP rises from 54.8 to 66.9.

5.5 CKA across encoder-pool sizes

Linear centered kernel alignment (CKA) compares projector outputs between retrained subset models. We average visual tokens within each example, match example identifiers within each task, and normalize the inner product of the centered sample Gram matrices. Values closer to one indicate more similar representations.

We average tasks within each of 465 unordered model pairs, using all tasks for Overall and four for each family. Spearman ρ relates CKA to $\min(k_a, k_b)$, where k_a and k_b count active encoders. Table 6 shows increasing Overall CKA across same-size groups, with the clearest increase on Vision-Centric tasks. These pairwise measurements reuse model checkpoints and are statistically dependent.

Table 5: **Per-encoder effective rank by subset size.** Mean ± population SD across subsets containing each encoder, not training seeds. Change is full-pool minus singleton rank before rounding. The final row gives subset counts.

Encoder	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	Change
CLIP	54.8	64.7 ± 3.7	62.6 ± 3.3	64.7 ± 5.5	66.9	$+12$
ConvNeXt	54.0	54.9 ± 11.9	52.0 ± 20.9	63.6 ± 12.4	27.5	-26
SAM	17.5	11.7 ± 3.7	8.3 ± 2.2	7.5 ± 1.5	5.6	-12
Pix2Struct	39.3	29.8 ± 7.2	29.5 ± 3.0	27.7 ± 1.7	26.6	-13
EVA-02	44.0	40.9 ± 4.2	40.1 ± 5.5	36.0 ± 6.6	33.4	-11
Subsets per encoder	1	4	6	4	1	

Table 6: **Projector CKA by encoder-pool size.** Mean CKA for equal-size ($1/1$–$4/4$) and unequal-size (Mixed) model pairs. Spearman ρ relates CKA to $\min(k_a, k_b)$ over 465 pairs. There is one full-pool model, so no $5/5$ pair.

Family	$1/1$	$2/2$	$3/3$	$4/4$	Mixed	ρ
Overall	0.754	0.830	0.830	0.852	0.821	0.322
General	0.759	0.831	0.838	0.862	0.818	0.397
Knowledge	0.746	0.812	0.774	0.808	0.795	0.121
OCR & Chart	0.699	0.800	0.806	0.829	0.796	0.311
Vision-Centric	0.810	0.876	0.901	0.908	0.874	0.526

Rank and CKA probe different interfaces: rank describes variation within pre-projector encoder features, whereas CKA compares projector-output geometry across models. High cross-model similarity can therefore coexist with different changes in the constituent encoder ranks.

6 Discussion

Encoder roles are joint-training properties. CLIP's performance with ConvNeXt illustrates the importance of joint training when selecting a partner. IM and TR distinguish dependence on a fixed checkpoint from contribution after training with a different encoder set. CLIP+ConvNeXt improves over its stronger singleton in every family (Table 3). CLIP+Pix2Struct has a larger OCR & Chart gain yet a lower overall score; its starting singleton baseline is weaker. Pair selection thus benefits from considering both baseline capability and incremental gain.

Implications for multi-encoder distillation. These measurements suggest what to preserve when distilling multiple encoders into one backbone [31, 1, 39]. For SAM, the relevant check is the score change when the branch is removed; for Pix2Struct, it is retention of OCR & Chart capability. CLIP raises a different question: whether the student preserves the representation learned jointly with its partner, rather than its singleton representation.

Scope and limitations. Our experiments use channel concatenation, Vicuna-7B, and the Eagle-X5 encoder pool, with rank measured at the shared projector input. Cross-attention and routing use different feature interfaces. The anchor and complement pattern is evaluated through CLIP and ConvNeXt. Each subset is trained once, so score differences are descriptive and repeated-seed uncertainty is not estimated.

7 Conclusion

This paper conducts retraining-based ablation experiments on encoder combinations in multi-encoder LVLMs to analyze the functional roles of individual encoders. Meanwhile, from the perspectives of capacity, necessity, and effective rank, it further proposes practical strategies for encoder selection. Future foundation models for world modeling and embodied perception will inevitably fuse heterogeneous perceptual information flows. Our method is not only applicable to multi-encoder LVLMs, but also contributes to the development and optimization of future world modeling models.

Acknowledgments and Disclosure of Funding

We thank Ruobing Xie, Xingwu Sun, and Yu Wang for insightful discussions, guidance, and support.

This work received no third-party funding. The authors declare no competing interests.

References

[1] Jiajun Cao, Yuan Zhang, Tao Huang, Ming Lu, Qizhe Zhang, Ruichuan An, Ningning Ma, and Shanghang Zhang. MoVE-KD: Knowledge distillation for VLMs with mixture of visual encoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. arXiv:2501.01709.

[2] Xiaoran Fan, Tao Ji, Changhao Jiang, Shuo Li, Senjie Jin, Sirui Song, Junke Wang, Boyang Hong, Lu Chen, Guodong Zheng, et al. MouSi: Poly-visual-expert vision-language models, 2024.

[3] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. EVA-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024. arXiv:2303.11331.

[4] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394*, 2023. URL `https://arxiv.org/abs/2306.13394`.

[5] Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching, 2024.

[6] Greg Heinrich, Mike Ranzinger, Hongxu Yin, Yao Lu, Jan Kautz, Andrew Tao, Bryan Catanzaro, and Pavlo Molchanov. RADIOv2.5: Improved baselines for agglomerative vision foundation models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22487–22497, 2025. arXiv:2412.07679.

[7] Drew A. Hudson and Christopher D. Manning. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. URL `https://arxiv.org/abs/1902.09506`.

[8] Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2405.07987.

[9] Dongsheng Jiang, Yuchen Liu, Songlin Liu, Jin'e Zhao, Hao Zhang, Zhen Gao, Xiaopeng Zhang, Jin Li, and Hongkai Xiong. From CLIP to DINO: Visual encoders shout in multi-modal large language models, 2024.

[10] Oğuzhan Fatih Kar, Alessio Tonioni, Petra Poklukar, Achin Kulshrestha, Amir Zamir, and Federico Tombari. BRAVE: Broadening the visual encoding of vision-language models, 2024.

[11] Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic VLMs: Investigating the design space of visually-conditioned language models, 2024.

[12] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A Diagram Is Worth A Dozen Images. *arXiv preprint arXiv:1603.07396*, 2016. URL `https://arxiv.org/abs/1603.07396`.

[13] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023.

[14] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, 2019. arXiv:1905.00414.

[15] Byung-Kwan Lee, Beomchan Park, Chae Won Kim, and Yong Man Ro. MoAI: Mixture of all intelligence for large language and vision models, 2024.

[16] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot

parsing as pretraining for visual language understanding. In *International Conference on Machine Learning (ICML)*, 2023. arXiv:2210.03347.

[17] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. SEED-Bench: Benchmarking Multimodal LLMs with Generative Comprehension. *arXiv preprint arXiv:2307.16125*, 2023. URL https://arxiv.org/abs/2307.16125.

[18] Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-Gemini: Mining the potential of multi-modality vision language models, 2024.

[19] Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, Jiaming Han, Siyuan Huang, Yichi Zhang, Xuming He, Hongsheng Li, and Yu Qiao. SPHINX: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models, 2023.

[20] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[21] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is Your Multi-modal Model an All-around Player? In *European Conference on Computer Vision*, 2024. URL https://arxiv.org/abs/2307.06281.

[22] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xucheng Yin, Cheng lin Liu, Lianwen Jin, and Xiang Bai. OCRBench: On the Hidden Mystery of OCR in Large Multimodal Models. *arXiv preprint arXiv:2305.07895*, 2023. URL https://arxiv.org/abs/2305.07895.

[23] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11976–11986, 2022.

[24] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2209.09513.

[25] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating Mathematical Reasoning of Foundation Models in Visual Contexts. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.02255.

[26] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. arXiv:1705.07874.

[27] Gen Luo, Yiyi Zhou, Yuxin Zhang, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Feast your eyes: Mixture-of-resolution adaptation for multimodal large language models. *arXiv preprint arXiv:2403.03003*, 2024.

[28] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2022. URL https://arxiv.org/abs/2203.10244.

[29] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A Dataset for VQA on Document Images. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021. URL https://arxiv.org/abs/2007.00398.

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021.

[31] Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. AM-RADIO: Agglomerative vision foundation model – reduce all domains into one. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12490–12500, 2024. arXiv:2312.06709.

[32] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *European Signal Processing Conference (EUSIPCO)*, pages 606–610, 2007.

[33] Min Shi, Fuxiao Liu, Shihao Wang, Shijia Liao, Subhashree Radhakrishnan, Yilin Zhao, De-An Huang, Hongxu Yin, Karan Sapra, Yaser Yacoob, Humphrey Shi, Bryan Catanzaro, Andrew Tao, Jan Kautz, Zhiding Yu, and Guilin Liu. Eagle: Exploring the design space for multimodal LLMs with mixture of encoders, 2024.

[34] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA Models That Can Read. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. URL `https://arxiv.org/abs/1904.08920`.

[35] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning (ICML)*, 2017. arXiv:1703.01365.

[36] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs, 2024.

[37] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal LLMs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9568–9578, 2024.

[38] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: A circuit for indirect object identification in GPT-2 small. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2211.00593.

[39] Yimu Wang, Mozhgan Nasr Azadani, Sean Sedwards, and Krzysztof Czarnecki. HAWAII: Hierarchical visual knowledge transfer for efficient vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.19072.

[40] Yizhou Wang, Song Mao, Yang Chen, Yufan Shen, Yinqiao Yan, Pinlong Cai, Ding Wang, Guohang Yan, Zhi Yu, Xuming Hu, and Botian Shi. Investigating redundancy in multimodal large language models with multiple vision encoders. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2507.03262.

[41] xAI. Grok-1.5 Vision Preview, 2024. URL `https://x.ai/news/grok-1.5v`. Introduces the RealWorldQA benchmark.

[42] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A Massive Multi-discipline Multimodal Understanding and Reasoning Benchmark for Expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL `https://arxiv.org/abs/2311.16502`.

[43] Tianyu Zhang, Suyuchen Wang, Chao Wang, Juan Rodriguez, Ahmed Masry, Xiangru Jian, Yoshua Bengio, and Perouz Taslakian. SCOPE: Selective cross-modal orchestration of visual perception experts, 2025.

[44] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

[45] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

[46] Zhuofan Zong, Bingqi Ma, Dazhong Shen, Guanglu Song, Hao Shao, Dongzhi Jiang, Hongsheng Li, and Yu Liu. MoVA: Adapting mixture of vision experts to multimodal context, 2024.

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: Sections 3.4–5 support the claims about protocol-dependent rankings, encoder roles, and effective rank in the Eagle-X5 setting.

2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: Section 6 describes the architectural scope, the tested encoder pool, and the absence of repeated-seed uncertainty estimates.

3. **Theory Assumptions and Proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [N/A]
   Justification: The paper is empirical; it does not prove theorems.

4. **Experimental Result Reproducibility**
   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper?
   Answer: [No]
   Justification: Section 3 describes the architecture, training data, configured recipe, and evaluation protocol. The repository supports regeneration of the tables and figures from processed results.

5. **Open Access to Data and Code**
   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?
   Answer: [No]
   Justification: The repository provides analysis code, processed result data, and instructions for reproducing the figures and tables.

6. **Experimental Setting/Details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?
   Answer: [Yes]
   Justification: Section 3 describes the architecture, training datasets, optimizer, learning rates, batch sizes, configured epochs, schedule, and benchmark aggregation.

7. **Experiment Statistical Significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [No]
   Justification: Each subset is trained once, and score differences are descriptive. Rank-table standard deviations measure variation across subsets, not training seeds.

8. **Experiments Compute Resources**
   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
   Answer: [No]
   Justification: The paper reports the evaluated encoder subsets and their parameter counts. Complete hardware and execution-time records for reproducing all experiments are not provided.

9. **Code of Ethics**
   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?
   Answer: [Yes]
   Justification: The work is empirical analysis of publicly available pretrained vision encoders and standard benchmarks. No human subjects, no new data collection, no models with elevated misuse risk.

10. **Broader Impacts**
    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
    Answer: [Yes]
    Justification: Compact encoder pools can reduce model size and deployment costs. Wider deployment may also increase privacy risks in applications such as surveillance.

11. **Safeguards**
    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?
    Answer: [N/A]
    Justification: The paper releases analysis data and plotting scripts, not new pretrained models or scraped datasets.

12. **Licenses for Existing Assets**
    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?
    Answer: [No]
    Justification: The models and benchmarks are credited in Section 3 and the references.

13. **New Assets**
    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
    Answer: [Yes]
    Justification: The release comprises analysis code and processed result data, with documentation in the repository README and data description. Retrained model checkpoints are not released.

14. **Crowdsourcing and Research with Human Subjects**
    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
    Answer: [N/A]
    Justification: No crowdsourcing or human-subjects work.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**
    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?
    Answer: [N/A]
    Justification: No human subjects.

16. **Declaration of LLM Usage**
    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?
    Answer: [N/A]
    Justification: LLMs assisted with language editing and figure preparation. Vicuna-7B is part of the model under study, rather than a tool used to develop the analysis methodology.