Visual Expert Skipping for MoE-MLLMs via K-Armed Bandit based Expert Estimation

Hongbo Zhou, Jiapeng Zhang, Yuqi Liu, Qiong Wu,
Jun Peng, Xiao Chen, Yiyi Zhou*, Rongrong Ji

Key Laboratory of Multimedia Trusted Perception and Efficient Computing,
Ministry of Education of China, Xiamen University, 361005, P.R. China.
{zhouhongbo,zhangjiapeng,liuyuqi,qiong}@stu.xmu.edu.cn,
{pengjun,zhouyiyi,rrji}@xmu.edu.cn

Abstract

Despite great success, existing MoE-MLLMs still suffer from substantial visual expert redundancy, making visual expert skipping a critical solution. However, deriving an effective and fixed skipping policy is still an open problem, mainly due to the exponentially large layer-wise policy space and the dependency between expert allocations across layers. In this paper, we propose a novel and training-free approach for MoE-MLLMs, termed *K-armed bandit based expert redundancy estimation* (**KAB-MoE**). KAB-MoE defines the retained visual expert number at each MoE layer as a discrete action, and then conducts efficient one-shot RL estimations to obtain the layer-action reward matrix. Based on this reward matrix, KAB-MoE can quickly derive the optimal skipping policy that satisfies the predefined computation budgets, *e.g.*, expert skipping ratio. The obtained skipping policy can be directly applied to MoE-MLLMs for all examples, which not only facilitates high-throughput deployment but also contributes to practical acceleration. Extensive experiments on Kimi-VL-A3B and Qwen3-VL-MoE show that KAB-MoE effectively accelerates MoE-MLLMs while keeping strong multimodal performance, *e.g.*, retaining 97.09% average performance on Kimi-VL-A3B under the skipping ratio of 83%. Moreover, KAB-MoE also achieves competitive or superior performance over dynamic expert skipping SOTAs with practical inference speed-up. The source code is publicly available at `https://github.com/chowhongbo/KAB-MOE`.

1 Introduction

Driven by the great success in *large language models* (LLMs) (1, 2, 3), *Mixture-of-Experts* (MoE) has also become a popular paradigm for advanced multimodal LLMs (MLLMs) (4, 5, 6, 7). Compared with the *dense* structure (8), MoE (9) can achieve a better trade-off between model capacity and inference efficiency by only activating a small proportion of experts. To this end, recent endeavors often extend MoE-LLMs (10, 11, 12) to the vision-language tasks (13) via multimodal SFT training.

Despite the great success, existing MoE-MLLMs often suffer from a more severe visual redundancy issue. Specifically, to remedy the visual shortcoming (7, 14), advanced MLLMs (15, 16, 17) often resort to large token patches to represent the visual input, *e.g.*, $16k$ tokens for a high-resolution image or more than $100k$ tokens for a minute-long video (1FPS) (18, 19), which inevitably results in obvious redundant visual information (20). This case becomes more pronounced in MoE-MLLMs, since both visual and textual tokens activate the same number of experts across all MoE layers. However, recent studies reveal that visual tokens are less significant than the text ones in MLLMs (21, 22), *e.g.*,

*Corresponding Author.

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

skipping up to 50% of experts barely degrades performance (23), while they account for the major computation overhead. In this case, how to skip redundant visual experts becomes a critical task for the research of efficient MoE-MLLMs (5, 7).

However, the estimation of visual expert redundancy remains an open problem. Firstly, directly estimating all experts in MoE-MLLM is intractable, since visual tokens might route to all experts, especially *expert load-balancing* (9) is implemented during training (24, 25). Thus, the estimation objective becomes *"how to reduce the number of experts activated for each layer"*, namely *expert skipping policy*. Nevertheless, solving this skipping policy will encounter an excessively large optimization space for existing MoE-MLLM. For instance, Kimi-VL-A3B (26) has up to 7^{26} candidate skipping policies. Furthermore, existing thresholding-based methods still need to perform all routing comparisons during inference, limiting their actual speed-up. Besides, the setting of layer-wise thresholds (27) is hard to achieve a global optimum, which is also very labor-intensive. Overall, efficiently achieving a proper skipping policy for MoE-MLLMs is still a daunting challenge.

In this paper, we focus on the fixed visual expert skipping for MoE-MLLMs, and propose a *K-armed bandit* based expert estimation method termed **KAB-MoE**. Specifically, we regard the expert estimation as a *K-armed bandit problem* (28), and define the number of experts that need to be activated at each layer as the candidate actions. In this case, the objective of KAB-MoE is to obtain a visual expert skipping policy that yields the maximum reward given a predefined computation budget, *e.g.*, expert skipping ratio. In practice, inspired by previous one-shot network search (29, 30), we perform numerous policy samplings and step-wise reward updates on a small-scale calibration set. In this way, KAB-MoE can efficiently construct a layer-action reward matrix, based on which the optimal skipping policy can also be quickly derived in an offline manner.

Compared with existing expert skipping methods (10, 22), our KAB-MoE offers three main advantages. Firstly, via extensive estimations on the complete skipping policies, KAB-MoE can better model the sequential dependency across MoE layers than existing layer-wise estimation (thresholding) methods. Second, following the one-shot search principle (29), KAB-MoE can not only sufficiently learn a reliable layer-action reward matrix from sampled policies, but also greatly shorten the online search costs, *e.g.*, only 44.0 minutes for Kimi-VL-A3B. More importantly, KAB-MoE can generalize the obtained reward matrix across budgets, avoiding the repeated searches on the calibration set, *e.g.*, generalizing the 50% budget reward matrix to the policies of 67% or 83% budgets. Lastly, the fixed policy obtained by KAB-MoE can not only facilitate the proper allocation of both experts and computing resources, but also help to further speed up inference. For instance, the layers without expert allocations can be directly skipped, requiring no routing computations.

To validate KAB-MoE, we apply it to two advanced MoE-MLLMs, namely *Kimi-VL-A3B* (26) and *Qwen3-VL-MoE* (6), and conduct experiments on a variety of image and video understanding benchmarks. The experimental results not only confirm the effectiveness of KAB-MoE in reducing visual expert redundancy, e.g., achieving about 83% expert skipping ratio for Kimi-VL-A3B while retaining 97.09% average performance, but also show its strong performance against dynamic skipping methods (10, 12, 22). In particular, KAB-MoE consistently outperforms the SOTA method MoDES (22) on Kimi-VL-A3B across three skipping rates, while being up to $1.23\times$ and $1.62\times$ faster in TTFT and MoE expert latency than MoDES under the 83% skipping ratio for Kimi-VL.

Overall, our contributions are two-fold:

- We propose a K-armed-bandit-based expert redundancy estimation method termed KAB-MoE to quickly derive a fixed and effective visual expert skipping policy for MoE-MLLMs.
- KAB-MoE provides an effective fixed-policy derivation framework for improving the accuracy-efficiency trade-off of MoE-MLLMs, and yields great advantages over existing methods in both performance and efficiency.

2 Related Work

Driven by the great success of *large language models* (LLMs) (31, 18), multimodal LLMs (MLLMs) have achieved substantial progress across a wide range of vision-language tasks (32, 6, 7, 33). The dominant paradigm adopts an LLM-centric architecture, where a visual encoder extracts features from images or videos, a modality projector aligns them into the language embedding space, and a pre-trained LLM backbone performs unified reasoning and generation over concatenated multimodal

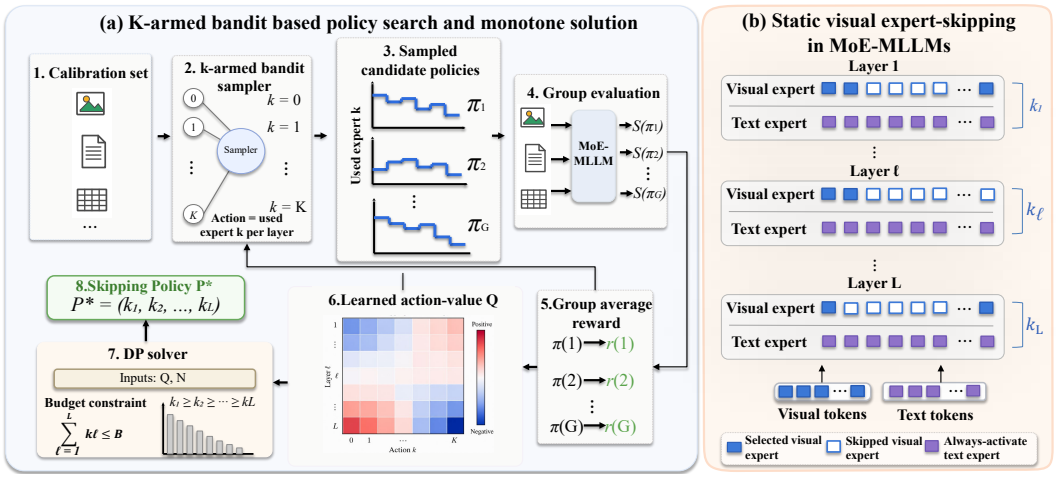


Figure 1: Illustration of the proposed KAB-MoE for MoE-MLLMs. (a) KAB-MoE formulates expert skipping policy search as a layer-action bandit problem. Candidate policies are sampled under a global expert skipping budget, and then *one-shot* estimations are extensively conducted on the calibration set to obtain the layer-action reward matrix $\mathbf{R}$. Based on $\mathbf{R}$, KAB-MoE further solves the optimal fixed policy $P = (a_1, a_2, \ldots, a_L)$ with both budget and monotone constraints, where a_ℓ denotes the number of retained visual experts at the ℓ-th MoE layer. (b) During inference, the skipping policy is applied to visual tokens of all examples, while text experts are kept unchanged.

tokens (34). This design effectively transfers the strong reasoning capabilities of LLMs to multimodal settings (35), but also introduces significant computational overhead (36) due to the large number of visual tokens, especially for high-resolution images and long videos.

To address the scalability challenge (37), Mixture-of-Experts (MoE) (9) has emerged as a promising solution for MLLMs (4, 26, 38). By activating only a sparse subset of experts for each token, MoE decouples model capacity from computation cost. However, existing MoE-MLLMs typically rely on load-balancing objectives during training (24, 25) to avoid expert collapse. While effective for optimization, such strategies implicitly encourage uniform expert utilization, which weakens sparsity and leads to substantial expert redundancy at inference time. To mitigate this, recent works explore training-free expert skipping in text-only MoE models. For instance, NAEE (10) dynamically prunes low-contribution experts based on routing probabilities, MC-MoE (11) incorporates attention-aware mechanisms to preserve critical experts, and DiEP (12) jointly considers routing probabilities and expert similarity to enable adaptive redundancy reduction. Despite their effectiveness in LLMs, applying these methods to MLLMs often leads to noticeable performance drops, as they overlook the visual computation redundancy caused by the massive number of visual tokens. To bridge this gap, MoDES (22) makes an early attempt to reduce expert redundancy in MoE-MLLMs, introducing a globally-modulated local gating mechanism and a dual-modality thresholding strategy to dynamically skip redundant experts in a modality-aware manner. However, existing methods remain dynamic, leading to unstable skipping patterns that hinder high-throughput deployment.

In this paper, we revisit expert redundancy from a different perspective and focus on visual expert skipping with a fixed policy. By explicitly modeling redundancy in visual tokens and formulating expert selection as a K-armed bandit problem, our approach enables stable and efficient inference, making it more suitable for practical deployment.

3 Method

3.1 Overall

In this paper, we focus on the optimal visual expert skipping policy for an MoE-MLLM, and propose a *K-armed bandit* based expert estimation method termed **KAB-MoE**, as illustrated in Fig. 1.

Specifically, given an input token x_l, and let $\mathcal{E}_\ell = \{E_\ell^1, E_\ell^2, \ldots, E_\ell^N\}$ denote the candidate experts at the ℓ-th layer, where N denotes the number of experts, a standard MoE layer will use the router to predict routing probabilities $\rho_\ell = \{\rho_\ell^1, \rho_\ell^2, \ldots, \rho_\ell^N\}$ for all experts, and then activates the Top-K ones. Thus, an MoE layer's output is defined by

$$h_\ell^K(x_\ell) = \sum_{i \in \text{Top-}K(\rho_\ell)} \rho_\ell^i E_\ell^i(x_\ell), \tag{1}$$

where K is the default number of activated experts per token.

To reduce the visual redundancy, we aim to obtain a skipping policy to set the number of visual experts activated at each layer. Here, the skipping policy is defined by $P = \{a_1, \ldots, a_L\}$, where a_x denotes the number of experts activated. Thus, for a visual token representation v_ℓ, the l-th MoE layer with expert skipping policy can be defined by

$$h_\ell^{a_\ell}(v_\ell) = \sum_{i \in \text{Top-}a_\ell(\rho_\ell^v)} \rho_\ell^i E_\ell^i(v_\ell), \quad a_\ell \in \{0, 1, \ldots, K\}. \tag{2}$$

Here, ρ_ℓ^v denotes the routing probabilities of the visual token. The fixed policy $P = \{a_1, \ldots, a_L\}$ is shared by all inputs during inference, avoiding input-dependent redundancy estimation.

Besides, the skipping policy P should also meet the predefined computation budget, *e.g.*, the overall expert skipping ratio in this paper. Let $N_v(E)$ and $N_t(E)$ denote the number of visual and text tokens of a given example, respectively. The skipping ratio of policy P on E is measured by

$$S(E, P) = \frac{N_v(E) \sum_{\ell=1}^{L} (K - a_\ell)}{\big(N_v(E) + N_t(E)\big) LK}. \tag{3}$$

Then, the objective of KAB-MoE is to find an optimal skipping policy based on the given budget b:

$$P_b^* = \arg\min_P d\Big(G(E \mid \hat{P}_0), G(E \mid P)\Big), \quad \text{s.t.} \quad S(E, P) \geq b, \tag{4}$$

where $\hat{P}_0$ denotes the default policy without skipping, P_b^* denotes the optimal policy under the skipping budget b, and $d(\cdot, \cdot)$ measures the discrepancy between the original and skipped MoE-MLLM. However, directly optimizing Eq. (4) is expensive even on a small calibration set, since an MoE-MLLM with L layers has up to K^L action combinations. In this case, we aim to resolve this challenge via a K-armed bandit based one-shot estimation.

3.2 K-Armed Bandit Based Expert Redundancy Estimation

Given an MoE-MLLM $G(\cdot)$, a calibration set $\mathcal{D}_{cal}$ and a global skipping budget b, KAB-MoE aims to learn a reward matrix $\mathbf{R} \in \mathbb{R}^{L \times (K+1)}$ reflecting the redundancy of MoE-layers. Specifically, for each MoE-layer, we define the number of visual experts activated as an action:

$$a_\ell \in \mathcal{A} = \{0, 1, \ldots, K\}.$$

Thus, a candidate policy is a sequence of actions denoted by $P_i = \{a_1^i, a_2^i, \ldots, a_L^i\}$. During estimation, candidate policies are sampled under the global budget in Eq. (3). In practice, we resort to the *one-shot* search principle (29, 30) to estimate the rewards of all actions.

Concretely, for a calibration example $E \in \mathcal{D}_{cal}$, we sample a group of m non-duplicated policies $\{P_i\}_{i=1}^m$ satisfying the budget b. Each policy is evaluated by the ***KL divergence*** between the original model and the skipped model:

$$\mathcal{L}(E, P_i) = \text{KL}\Big(G(E \mid P_0^*) \parallel G(E \mid P_i)\Big). \tag{5}$$

A smaller $\mathcal{L}(E, P_i)$ means that policy P_i better preserves the behavior of the original MoE-MLLM.

To make rewards comparable within the same sampled group, the actual reward of each policy r_i is compared to a baseline one:

$$r_i(E, P_i) = \frac{1}{m} \sum_{j=1}^{m} \mathcal{L}(E, P_j) - \mathcal{L}(E, P_i). \tag{6}$$

Here, $\frac{1}{m} \sum_{j=1}^{m} \mathcal{L}(E, P_j)$ denotes the baseline reward of this group. Thus, a policy receives a positive reward if it produces a smaller discrepancy than the group average, and a negative reward otherwise.

Reward matrix update. Akin to one-shot RL search (30), KAB-MoE assigns the policy-level advantage reward to all selected layer-action pairs in the policy. In practice, we maintain a reward matrix $\mathbf{R}$ initialized by zeros. For each sampled policy $P_i = \{a_1^i, \ldots, a_L^i\}$, the selected action at each layer is updated by

$$r_\ell^i \leftarrow r_\ell^i + r_i(E, P_i), \quad \ell = 1, \ldots, L. \tag{7}$$

This cumulative update estimates the expected contribution of each expert-number action from extensively complete-policy evaluations.

Advantage sampling. Following the principle of the K-armed bandit algorithm (28), KAB-MoE also performs advantage sampling to speed up the convergence of the reward matrix. For the ℓ-th layer, the probability of choosing action i is defined by

$$\pi_\ell(i) = \frac{\exp(r_\ell^i/\tau)}{\sum_{i' \in \mathcal{A}} \exp(r_\ell^{i'}/\tau)}, \tag{8}$$

where r_ℓ^i denotes the accumulated advantage reward of selecting action i at the l-th layer, and τ controls the exploration strength. Thus, high-reward actions are more likely to be sampled, while exploration is preserved by the temperature-controlled softmax distribution. Afterwards, we then sample policies layer by layer and retain policies that satisfy the global budget constraint in Eq. (3).

3.3 Monotone Policy Solving

After sufficient one-shot estimations, KAB-MoE obtains a reward matrix $\mathbf{R} \in \mathbb{R}^{L \times (K+1)}$. Given a global skipping budget b, we solve a fixed policy by maximizing accumulated rewards under the global token-weighted budget and monotonicity constraint.

Monotone policy solving. Let $\bar{N}_v$ and $\bar{N}_t$ denote the average visual and textual token numbers estimated from the calibration set. Since the reward matrix is estimated from sampled policies, the learned layer-action rewards may still contain local estimation noise. To regularize the final policy, we impose a monotone non-increasing constraint on the retained visual expert numbers. This design is consistent with the observation that visual information is more important in shallow layers and becomes increasingly redundant in deeper layers (21) of MLLMs. The final policy is solved by

$$\begin{aligned} P_b^* = \arg\max_{\mathbf{a}} \quad & \sum_{\ell=1}^{L} \mathbf{R}_{\ell, a_\ell} \\ \text{s.t.} \quad & \frac{\bar{N}_v \sum_{\ell=1}^{L} (K - a_\ell)}{(\bar{N}_v + \bar{N}_t) L K} \geq b, \\ & K \geq a_1 \geq a_2 \geq \cdots \geq a_L \geq 0, \\ & a_\ell \in \mathbb{Z}, \quad \ell = 1, \ldots, L. \end{aligned} \tag{9}$$

The first constraint enforces the global expert skipping budget over all token-expert activations, not merely over visual experts. The second constraint imposes a monotone non-increasing structure over layers, preventing the solver from assigning more visual experts to deeper layers than to earlier ones. Since all variables are discrete, Eq. (9) can be solved offline by *dynamic programming*.

Fixed-policy inference. Once $P_b^* = \{a_1^*, \ldots, a_L^*\}$ is obtained, it is fixed for all inputs. For visual tokens, the ℓ-th MoE layer activates only the Top-a_ℓ^* experts as in Eq. (2). For text tokens, KAB-MoE keeps the default Top-K routing. Therefore, the deployment stage introduces no extra predictor, no dynamic threshold search, nor input-dependent redundancy estimation.

Cross-budget policy derivation. The learned reward matrix can be reused for different global skipping budgets. After $\mathbf{R}$ is estimated once, KAB-MoE only needs to change b in Eq. (9) to derive policies for different accuracy-efficiency trade-offs, such as 67% and 83% global expert skipping. This avoids repeated calibration search and makes the method practical for deployment under different resource constraints. The detailed algorithm is given in the Appendix.

Table 1: Comparison to existing expert skipping methods on Kimi-VL-A3B and Qwen3-VL-MoE. *Avg.* denotes the average retention rate. * denotes the default setting without skipping.

Method	Image Understanding						Video Understanding					Avg.
	TextVQA	ChartQA	MMStar	MMB	MME	RealWorld	MVBench	EgoSchema	VMME	LVB	VMMMU	(%)
Kimi-VL-A3B												
$k = 6^*$	88.70	89.48	49.89	83.16	2207	65.36	61.80	78.18	66.59	63.13	49.33	100.00
Skip 50% Experts												
$k = 3$	85.41	86.20	51.21	80.67	2084	63.53	60.42	75.71	64.30	60.14	44.22	96.36
NAEE	86.14	85.74	50.82	80.58	2084	64.55	60.02	75.81	65.16	60.27	45.08	96.70
MC-MoE	86.28	87.94	51.61	81.32	2138	63.82	60.39	76.57	66.24	60.62	46.26	97.84
DieP	87.43	88.32	51.48	80.26	2159	64.74	61.06	77.32	65.96	61.04	47.83	98.57
MoDES	88.18	89.08	49.65	83.16	2203	65.62	61.95	78.41	67.19	62.83	49.00	<u>99.91</u>
Ours	87.67	89.36	50.25	82.99	2209	65.62	62.23	78.57	67.08	63.05	48.44	**99.97**
Skip 67% Experts												
$k = 2$	83.49	85.12	52.10	78.87	2084	62.99	59.35	70.80	62.15	57.67	41.44	94.06
NAEE	82.84	85.29	50.74	77.31	2083	64.54	59.68	72.29	63.74	58.36	43.68	94.75
MC-MoE	85.07	86.32	51.13	77.65	2104	63.61	59.86	74.36	64.22	59.73	45.21	95.96
DieP	84.21	85.56	50.76	78.94	2087	64.02	60.02	72.97	61.07	58.45	44.93	95.05
MoDES	85.57	88.24	49.25	82.73	2204	64.58	61.65	77.98	66.52	62.90	48.78	<u>99.08</u>
Ours	86.91	88.40	48.69	82.74	2206	65.75	61.95	78.22	66.63	62.53	49.00	**99.38**
Skip 83% Experts												
$k = 1$	77.17	76.68	42.65	54.55	2071	54.38	51.10	37.23	50.52	43.83	24.56	75.11
NAEE	75.73	78.41	41.48	69.14	1827	60.32	58.41	57.28	53.49	49.69	42.64	84.33
MC-MoE	79.41	78.31	43.57	73.42	2063	62.54	58.87	63.56	59.87	54.38	44.02	89.42
DieP	82.32	78.31	42.47	76.28	2071	61.34	59.15	61.27	57.49	52.41	43.81	88.83
MoDES	82.38	84.20	46.68	81.44	2162	64.84	61.30	76.98	65.48	62.60	47.11	<u>96.94</u>
Ours	85.22	86.48	46.72	82.05	2103	64.58	61.27	76.92	65.04	62.38	46.78	**97.09**
Qwen3-VL-MoE												
$k = 8^*$	83.41	85.08	59.67	86.60	2500	66.80	64.67	62.45	54.89	55.42	47.11	100.00
Skip 63% Experts												
$k = 3$	80.81	78.12	66.74	83.33	2326	45.88	62.02	57.96	53.48	54.60	50.44	95.20
NAEE	81.20	79.41	55.39	84.18	2348	59.67	61.31	58.32	51.08	55.12	48.32	95.61
MC-MoE	82.51	79.37	56.48	86.12	2438	62.01	62.61	58.73	54.22	54.13	48.54	97.09
DieP	82.04	80.23	57.26	85.07	2405	60.31	63.15	59.46	53.41	55.08	48.76	96.80
MoDES	81.82	82.48	58.61	86.17	2493	63.92	64.42	62.39	55.15	55.50	49.89	<u>99.22</u>
Ours	82.81	84.36	58.09	86.60	2475	66.01	64.10	62.57	54.89	55.80	50.33	**100.04**
Skip 75% Experts												
$k = 2$	77.54	69.60	62.38	80.50	2060	55.56	60.70	53.79	50.67	54.08	46.00	92.03
NAEE	78.42	77.28	54.64	81.34	2208	61.75	60.98	55.24	48.87	54.87	47.12	93.25
MC-MoE	80.13	78.41	57.02	85.32	2286	61.83	61.65	57.13	52.64	54.03	47.49	95.11
DieP	79.64	78.52	56.48	84.91	2243	60.94	62.78	57.86	52.38	54.62	48.16	95.21
MoDES	81.62	81.64	57.72	86.00	2459	65.36	63.62	62.43	52.93	55.12	48.89	<u>98.55</u>
Ours	82.46	81.92	57.62	86.08	2425	64.71	64.25	61.92	54.67	55.72	47.44	**98.58**

4 Experiments

4.1 Experimental Settings

Models and Datasets. We evaluate our method on two recent MoE-based multimodal large language models, namely Kimi-VL-A3B (26) and Qwen3-VL-MoE-30B-A3B-Instruct (6). The evaluation covers both image and video understanding tasks. For image benchmarks, we consider TextVQA(39), ChartQA(40), MMStar(41), MMB(42), MME(43), and RealWorldQA(44). For video benchmarks, we evaluate on MVBench(45), EgoSchema(46), Video-MME(47), LongVideoBench(48), and Video-MMMU(49).

Implementation Details. For Kimi-VL-A3B, we evaluate three expert skipping rates, *i.e.*, 50%, 67%, and 83%. For Qwen3-VL-MoE, we evaluate 63% and 75% skipping rates. We use 3,000 samples from GQA as the calibration set for policy search unless otherwise specified. In the search stage, the number of retained experts at each layer is treated as a discrete action, and the layer-wise action preferences are optimized through repeated policy sampling on the calibration set. Specifically, policies are sampled with a group size of m=16, and actions are drawn from a softmax distribution with temperature τ=0.8. For Kimi-VL-A3B, the policies for 67% and 83% skipping ratios are derived from the reward matrix obtained under the skipping ratio of 50% following Sec. 3.3. Similarly, for Qwen3-VL-MoE, the skipping ratio of 75% policy is derived from the reward matrix searched under the skipping ratio of 63%. The final static policy is then solved under the target skipping-ratio constraint and directly applied during inference. All experiments are conducted on 8 NVIDIA A800 GPUs, and the remaining settings follow the default configurations.

4.2 Experimental results

Comparison with existing methods. We first compare the proposed KAB-MoE with existing expert skipping methods (12, 22) on Kimi-VL-A3B and Qwen3-VL-MoE under different expert

Table 2: Cross-budget policy derivation. The reward matrix of KAB-MoE estimated under a lower skipping ratio can be reused for higher-ratio policy solving without repeated calibration search.

Setting	Image Understanding						Video Understanding					Avg. (%)
	TextVQA	ChartQA	MMStar	MMBench	MME	RealWorldQA	MVBench	EgoSchema	VMME	LVB	VMMMU	
Kimi-VL-A3B												
$k = 6^*$	88.70	89.48	49.89	83.16	2207	65.36	61.80	78.18	66.59	63.13	49.33	100.00
67%	87.53	88.40	48.42	82.47	2159	65.88	61.33	78.19	66.07	62.53	47.78	98.79
50%→67%	86.91	88.40	48.69	82.74	2206	65.75	61.95	78.22	66.63	62.53	49.00	**99.38**
83%	84.37	86.08	46.97	82.04	2092	65.49	61.10	76.96	64.70	62.30	45.89	96.84
50%→83%	85.22	86.48	46.72	82.05	2103	64.58	61.27	76.92	65.04	62.38	46.78	**97.09**
Qwen3-VL-MoE												
$k = 8^*$	83.41	85.08	59.67	86.60	2500	66.80	64.67	62.45	54.89	55.42	47.11	100.00
75%	81.66	81.08	58.48	86.17	2422	64.71	64.28	62.10	55.00	55.35	47.55	98.57
63%→75%	82.46	81.92	57.62	86.08	2425	64.71	64.25	61.92	54.67	55.72	47.44	**98.58**

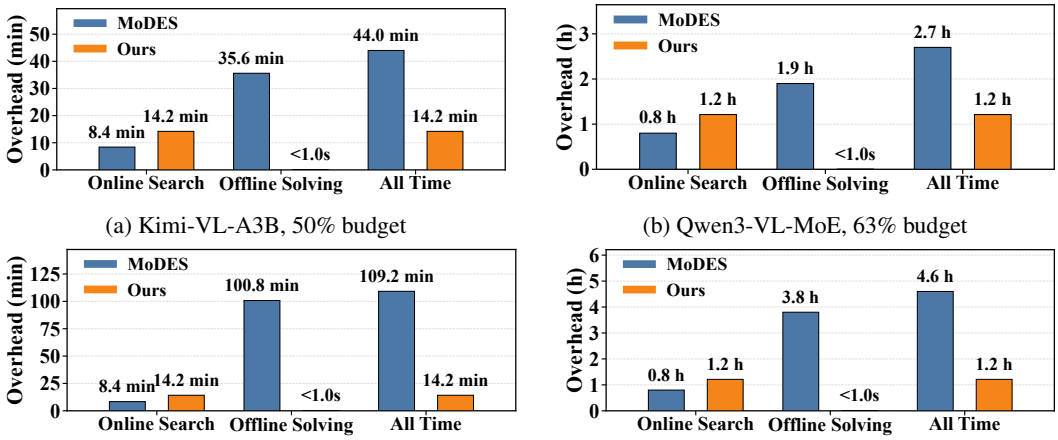
Table 3: Inference efficiency comparison. Here, *TTFT* denotes *Time To First Token*, and *MoE Total* refers to the specific latency of MoE experts. *(T/M)* denotes the speed-up of two metrics. Sequence Length refers to the number of visual tokens tested.

Method	Sequence Length = 512			Sequence Length = 1024		
	TTFT (ms)	MoE Total (ms)	Speedup (T/M)	TTFT (ms)	MoE Total (ms)	Speedup (T/M)
Baseline	195.96	95.27	1.00× / 1.00×	361.62	148.26	1.00× / 1.00×
MoDES	113.55	48.60	1.73× / 1.96×	204.80	100.92	1.77× / 1.47×
Ours	**92.65**	**30.74**	**2.12× / 3.10×**	**169.86**	**62.43**	**2.13× / 2.38×**

skipping ratios, as reported in Tab. 1. The compared methods include uniform fixed-k baselines, LLM-oriented expert skipping methods such as NAEE (10), MC-MoE (11) and DiEP (12), as well asd the multimodal skipping method MoDES (22). As can be seen, the baseline that unformly reduces the number of experts leads to substantial performance degradation. For instance, using only one expert ($k = 1$) reduces the average retention of Kimi-VL-A3B to 75.11%. A similar trend is observed on Qwen3-VL-MoE, where the uniform baseline($K = 2$) achieves only 92.03% retention under the skipping ratio of 75%. These results suggest that expert redundancy is highly layer-dependent, making a layer-wise skipping policy necessary. Another observation is that skipping approaches designed for MoE-LLMs struggle to achieve a good performance-efficiency trade-off in multimodal settings, *e.g.*, only 84.33% on Kimi-VL by NAEE. In contrast, the multimodal method MoDES maintains 96.94%. This gap suggests that MoE-MLLMs require skipping strategies that better account for multimodal redundancy patterns (22). In contrast, KAB-MoE consistently achieves the best performance across different skipping budgets. Notably, under the skipping ratio of 83%, KAB-MoE further surpasses MoDES and achieves better results on several image benchmarks, including TextVQA, ChartQA, MMStar and MMB. Similar advantages are also observed on Qwen3-VL-MoE, where KAB-MoE obtains 100.04% retention under the skipping ratio of 63% and 98.58% retention under the skipping ratio of 75%, outperforming MoDES by 0.82%. Notably, KAB-MoE strictly follows a fixed predefined expert budget for all inputs, whereas dynamic skipping methods may exhibit slight variations in their actual skipping ratios during inference. Overall, these results confirm that KAB-MoE can derive high-quality static expert skipping policies and generalize effectively across different MoE-MLLM backbones and skipping budgets.

Cross-budget policy derivation. We further evaluate the cross-budget generalization ability of KAB-MoE in Tab. 2. These results show that the reward matrix searched under lower skipping ratios remains in effective deriving policies at higher skipping ratios. For Kimi-VL, reusing the reward matrix learned under 50% skipping yields 99.38% average retention for the 67% skipping policy, outperforming the directly searched counterpart, *i.e.*, 98.79% retention. This advantage can be also witnessed on Qwen3-VL-MoE. When reusing the reward matrix learned under the skipping ratio of 63% to derive the 75% policy, KAB-MoE achieves 98.58% average retention. These results indicate that the learned reward matrix is not tightly bound to a single skipping budget, but instead captures reusable layer-action preferences across different skipping ratios. As a result, KAB-MoE can derive effective higher-skipping policies without repeated reward estimation, substantially improving search flexibility and efficiency.

Efficiency Comparison. We compare the policy expenditure of KAB-MoE and MoDES in Fig. 2. KAB-MoE incurs slightly higher online search cost due to its policy-sampling process, whereas



(a) Kimi-VL-A3B, 50% budget

(b) Qwen3-VL-MoE, 63% budget

(c) Kimi-VL-A3B, 3 budgets

(d) Qwen3-VL-MoE, 2 budgets

Figure 2: Policy overhead comparison between KAB-MoE and MoDES. The top row reports the single-budget overhead breakdown, while the bottom row reports the one for multiple budgets.

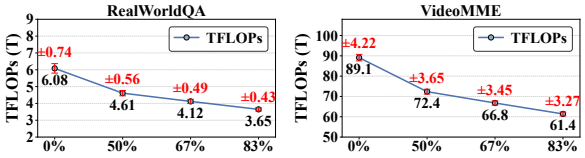


Figure 3: TFLOPs reductions with different skipping ratios for Kimi-VL-A3B.

Table 4: Ablation on the calibration size for Kimi-VL-A3B-50%.

Calib. size	ChartQA	MMBench	VMME	ΔAvg.
Baseline ($k = 6^*$)	89.48	83.16	66.59	0.00
1K	89.00	83.07	66.92	-0.05
3K	**89.36**	82.99	67.08	**+0.13**
5K	89.00	83.16	**67.14**	+0.10
10K	89.16	**83.33**	66.85	+0.08

MoDES repeatedly performs inference on the same calibration samples for layer-importance estimation. Nevertheless, since both methods rely on online calibration-set evaluation, the practical online-time gap remains limited. In contrast, KAB-MoE significantly reduces offline solving cost. MoDES requires an expensive offline stage to derive the final skipping policy, while KAB-MoE only solves a fixed expert skipping policy from the learned reward matrix, reducing the offline solving cost to less than 1 second. As a result, KAB-MoE substantially reduces the total search overhead. For instance, on Kimi-VL-A3B with 50% skipping, the search time decreases from 44 minutes to 14.2 minutes, achieving a $3\times$ reduction. This advantage becomes more significant under cross-budget policy derivation. Since the same reward matrix can be reused across multiple skipping ratios, KAB-MoE requires only a single reward-estimation stage, while MoDES has to repeat its deriving process for each target budget. As can be seen in Fig. 2c and Fig. 2d, deriving three Kimi-VL-A3B skipping ratios from a 50% reward matrix reduces the effective search cost from about 100 minutes to 14 minutes, corresponding to roughly $7\times$ lower overhead. Similarly, for Qwen3-VL-MoE, deriving two skipping ratios from the same 63% reward matrix reduces the total overhead from 4.6 hours to 1.2 hours, achieving about $4\times$ lower search cost.

Fig. 3 further shows that the searched static policies consistently reduce the computation cost compared with the default Kimi-VL-A3B baseline. Under 83% skipping ratio, the average TFLOPs decrease from 6.08T to 3.65T on RealWorldQA and from 89.1T to 61.4T on VideoMME. These results show that KAB-MoE is efficient not only in the policy search stage, but also in reducing the actual computation for both image and video inputs. In Tab. 3, we further report the actual inference acceleration under the skipping ratio of 83%. We measure the metrics of *Time To First Token* (TTFT) and the latency of MoE expert computation (*MoE Total*). As shown in Tab. 3, KAB-MoE consistently achieves stronger acceleration than MoDES across different sequence lengths, especially for MoE expert computation. At the sequence length of 512, KAB-MoE improves the TTFT and MoE Total from $1.73\times$ to $2.12\times$ and $1.96\times$ to $3.10\times$. This indicates that the proposed fixed expert skipping policy can bring stable acceleration benefits by the static expert allocations, which avoids input-dependent threshold decisions during inference and strictly follows the predefined expert budget. Overall, these results demonstrate that KAB-MoE not only preserves performance under high skipping ratios, but also brings practical inference acceleration for MoE-MLLMs.

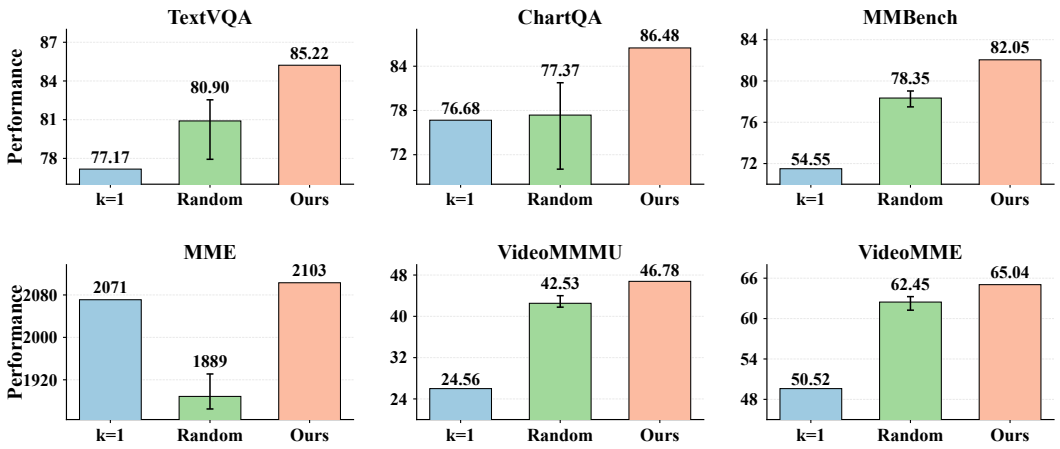


Figure 4: Comparisons among three policy strategies under the skipping ratio of 83% on Kimi-VL-A3B. For random sampling, we average the results of three randomly sampled policies.

4.3 Ablation Studies

Search Strategy. To validate the effectiveness of our RL-based search strategies, we compare it with two alternatives in Fig. 4. From this figure, our KAB-MoE exhibits great advantages over the uniform baselines. Random sampling directly samples visual expert skipping policies that satisfy the target budget. However, the performance of random sampling is highly dependent on the quality of the sampled policy, since the error ranges in Fig. 4 reflect the large variance of this policy space. In contrast, KAB-MoE uses the reward matrix to model the relative benefits of retaining different numbers of visual experts at different layers, enabling the final solver to derive a stronger fixed policy from this high-variance search space. For example, KAB-MoE clearly outperforms random sampling on all benchmarks. Overall, these results show that the advantage of KAB-MoE lies in its reward-matrix-guided policy search and solving, which effectively derives stronger fixed visual expert skipping policies under high skipping ratios.

Calibration Sample Size. We also ablate the effect of the calibration set size in Tab. 4. From these results, one observation is that KAB-MoE can already achieve near-baseline performance with only a small calibration set. For instance, using 1K calibration samples leads to an average performance change of only -0.05, indicating that a limited number of samples can already provide useful layer-wise reward estimations. When the calibration size increases to 3K, the average performance change further improves to $+0.13$, showing that a moderate number of calibration samples is sufficient for learning stable layer-action preferences. After further increasing the calibration size to 5K and 10K, the average changes become $+0.10$ and $+0.08$, respectively, without bringing consistent additional gains. This case suggests that KAB-MoE does not rely on a large calibration set or excessive sampling steps to obtain effective policies. It also implies that the reward matrix can converge quickly through advantage-guided reward estimation, well confirming our designs.

5 Conclusion

In this paper, we study the problem of visual expert redundancy in MoE-MLLMs and propose a novel and training- free approach for MoE-MLLMs termed KAB-MoE, which adopts the K-armed bandit based expert redundancy estimation. Specifically, KAB-MoE defines the visual expert skipping as a K-armed bandit problem, and formulates the number of retained visual experts per MoE layer as a discrete action. In practice, KAB-MoE conducts efficient one-shot RL estimations on a small calibration set to obtain the layer-action reward matrix, based on which we can quickly solve the optimal skipping policy under a predefined computation budget. The resulting fixed policy can be directly applied to all inputs at inference time, enabling stable resource allocation and practical acceleration. Extensive experiments on two representative MoE-MLLMs and a variety of image and video benchmarks will confirm the effectiveness of KAB-MoE towards the superior performance-efficiency trade-off for MoE-MLLMs.

Acknowledgments and Disclosure of Funding

This work was supported by the New Generation Artificial Intelligence-National Science and Technology Major Project (No. 2025ZD0122703), the National Science Fund for Distinguished Young Scholars (No. 62525605), the National Natural Science Foundation of China (No. U25B2066, No. U22B2051, No.62572407) , Special Science and Technology Program and Natural Science Foundation of Fujian Province (No. 2025H0041, No. 2026J010001) and the Xiamen Municipal Science and Technology Bureau, China (3502ZC-QXT2024009). The authors declare no competing interests.

References

[1] DeepSeek-AI, "Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model," *CoRR*, vol. abs/2405.04434, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2405.04434

[2] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang, G. Dong, H. Wei, H. Lin, J. Tang, J. Wang, J. Yang, J. Tu, J. Zhang, J. Ma, J. Yang, J. Xu, J. Zhou, J. Bai, J. He, J. Lin, K. Dang, K. Lu, K. Chen, K. Yang, M. Li, M. Xue, N. Ni, P. Zhang, P. Wang, R. Peng, R. Men, R. Gao, R. Lin, S. Wang, S. Bai, S. Tan, T. Zhu, T. Li, T. Liu, W. Ge, X. Deng, X. Zhou, X. Ren, X. Zhang, X. Wei, X. Ren, X. Liu, Y. Fan, Y. Yao, Y. Zhang, Y. Wan, Y. Chu, Y. Liu, Z. Cui, Z. Zhang, Z. Guo, and Z. Fan, "Qwen2 technical report," *CoRR*, vol. abs/2407.10671, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2407.10671

[3] A. Yang *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[4] B. Lin, Z. Tang, Y. Ye, J. Huang, J. Zhang, Y. Pang, P. Jin, M. Ning, J. Luo, and L. Yuan, "Moe-llava: Mixture of experts for large vision-language models," *arXiv preprint arXiv:2401.15947*, 2024.

[5] L. Shen, G. Chen, R. Shao, W. Guan, and L. Nie, "Mome: Mixture of multimodal experts for generalist multimodal large language models," in *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak, and C. Zhang, Eds., 2024. [Online]. Available: http://papers.nips.cc/paper_files/paper/2024/hash/4a3a14b9536806a0522930007c5512f7-Abstract-Conference.html

[6] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, W. Ge, Z. Guo, Q. Huang, J. Huang, F. Huang, B. Hui, S. Jiang, Z. Li, M. Li, M. Li, K. Li, Z. Lin, J. Lin, X. Liu, J. Liu, C. Liu, Y. Liu, D. Liu, S. Liu, D. Lu, R. Luo, C. Lv, R. Men, L. Meng, X. Ren, X. Ren, S. Song, Y. Sun, J. Tang, J. Tu, J. Wan, P. Wang, P. Wang, Q. Wang, Y. Wang, T. Xie, Y. Xu, H. Xu, J. Xu, Z. Yang, M. Yang, J. Yang, A. Yang, B. Yu, F. Zhang, H. Zhang, X. Zhang, B. Zheng, H. Zhong, J. Zhou, F. Zhou, J. Zhou, Y. Zhu, and K. Zhu, "Qwen3-vl technical report," 2025. [Online]. Available: https://arxiv.org/abs/2511.21631

[7] W. Wang, Z. Gao, L. Gu, H. Pu, L. Cui, X. Wei, Z. Liu, L. Jing, S. Ye, J. Shao *et al.*, "Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency," *arXiv preprint arXiv:2508.18265*, 2025.

[8] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[9] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. V. Le, G. E. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *5th International Conference on Learning Representations, ICLR*, 2017.

[10] X. Lu, Q. Liu, Y. Xu, A. Zhou, S. Huang, B. Zhang, J. Yan, and H. Li, "Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024, pp. 6159–6172.

[11] W. Huang, Y. Liao, J. Liu, R. He, H. Tan, S. Zhang, H. Li, S. Liu, and X. Qi, "Mixture compressor for mixture-of-experts llms gains more," *arXiv preprint arXiv:2410.06270*, 2024.

[12] S. Bai, H. Li, J. Zhang, Z. Hong, and S. Guo, "Diep: Adaptive mixture-of-experts compression through differentiable expert pruning," *arXiv preprint arXiv:2509.16105*, 2025.

[13] Y. Zhou, T. Ren, C. Zhu, X. Sun, J. Liu, X. Ding, M. Xu, and R. Ji, "Trar: Routing the attention spans in transformer for visual question answering," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2074–2084.

[14] B. Tong, B. Lai, Y. Zhou, G. Luo, Y. Shen, K. Li, X. Sun, and R. Ji, "Flashsloth: Lightning multimodal large language models via embedded visual compression," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 14 570–14 581.

[15] L. Chen, H. Zhao, T. Liu, S. Bai, J. Lin, C. Zhou, and B. Chang, "An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models," in *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXI*, ser. Lecture Notes in Computer Science, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., vol. 15139. Springer, 2024, pp. 19–35. [Online]. Available: https://doi.org/10.1007/978-3-031-73004-7_2

[16] D. Lu, Y. Sun, Z. Zhang, L. Huang, J. Zeng, M. Shu, and H. Cao, "Internvl-x: Advancing and accelerating internvl series with efficient visual token compression," *CoRR*, vol. abs/2503.21307, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2503.21307

[17] Q. Wu, Y. Zhou, W. Ye, X. Sun, and R. Ji, "Not all attention is needed: Parameter and computation efficient tuning for multi-modal large language models via effective attention skipping," *International Journal of Computer Vision*, vol. 134, no. 3, p. 128, 2026.

[18] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang, G. Dong, H. Wei, H. Lin, J. Tang, J. Wang, J. Yang, J. Tu, J. Zhang, J. Ma, J. Yang, J. Xu, J. Zhou, J. Bai, J. He, J. Lin, K. Dang, K. Lu, K. Chen, K. Yang, M. Li, M. Xue, N. Ni, P. Zhang, P. Wang, R. Peng, R. Men, R. Gao, R. Lin, S. Wang, S. Bai, S. Tan, T. Zhu, T. Li, T. Liu, W. Ge, X. Deng, X. Zhou, X. Ren, X. Zhang, X. Wei, X. Ren, X. Liu, Y. Fan, Y. Yao, Y. Zhang, Y. Wan, Y. Chu, Y. Liu, Z. Cui, Z. Zhang, Z. Guo, and Z. Fan, "Qwen2 technical report," 2024. [Online]. Available: https://arxiv.org/abs/2407.10671

[19] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu *et al.*, "Llava-onevision: Easy visual task transfer," *arXiv preprint arXiv:2408.03326*, 2024.

[20] X. Ye, Y. Gan, Y. Ge, X.-P. Zhang, and Y. Tang, "Atp-llava: Adaptive token pruning for large vision language models," *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24 972–24 982, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:274436316

[21] Q. Wu, W. Lin, Y. Zhou, W. Ye, Z. Zen, X. Sun, and R. Ji, "Accelerating multimodal large language models via dynamic visual-token exit and the empirical findings," *arXiv preprint arXiv:2411.19628*, 2024.

[22] Y. Huang, Z. Wang, Z. Yuan, Y. Ding, R. Gong, J. Guo, X. Liu, and J. Zhang, "Modes: Accelerating mixture-of-experts multimodal large language models via dynamic expert skipping," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2026.

[23] Z. Wen, Y. Gao, W. Li, C. He, and L. Zhang, "Token pruning in multimodal large language models: Are we solving the right problem?" in *Annual Meeting of the Association for Computational Linguistics*, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276407981

[24] S. Rajbhandari, C. Li, Z. Yao, M. Zhang, R. Y. Aminabadi, A. A. Awan, J. Rasley, and Y. He, "Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation AI scale," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 2022, pp. 18 332–18 346. [Online]. Available: https://proceedings.mlr.press/v162/rajbhandari22a.html

[25] C. Hwang, W. Cui, Y. Xiong, Z. Yang, Z. Liu, H. Hu, Z. Wang, R. Salas, J. Jose, P. Ram, J. Chau, P. Cheng, F. Yang, M. Yang, and Y. Xiong, "Tutel: Adaptive mixture-of-experts at scale," in *Proceedings of the Sixth Conference on Machine Learning and Systems, MLSys 2023, Miami, FL, USA, June 4-8, 2023*, D. Song, M. Carbin, and T. Chen, Eds. mlsys.org, 2023. [Online]. Available: https://proceedings.mlsys.org/paper_files/paper/2023/hash/5616d34cf8ff73942cfd5aa922842556-Abstract-mlsys2023.html

[26] K. Team *et al.*, "Kimi-vl technical report," *arXiv preprint arXiv:2504.07491*, 2025.

[27] J. Xin, R. Tang, J. Lee, Y. Yu, and J. J. Lin, "Deebert: Dynamic early exiting for accelerating bert inference," in *Annual Meeting of the Association for Computational Linguistics*, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:216552850

[28] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[29] H. Cai, L. Zhu, and S. Han, "Proxylessnas: Direct neural architecture search on target task and hardware," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=HylVB3AqYm

[30] Y. Zhou, R. Ji, X. Sun, G. Luo, X. Hong, J. Su, X. Ding, and L. Shao, "K-armed bandit based multi-modal network architecture search for visual question answering," in *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*, C. W. Chen, R. Cucchiara, X. Hua, G. Qi, E. Ricci, Z. Zhang, and R. Zimmermann, Eds. ACM, 2020, pp. 1245–1254. [Online]. Available: https://doi.org/10.1145/3394171.3413998

[31] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," 2022. [Online]. Available: https://arxiv.org/abs/2203.02155

[32] Z. Wu, X. Chen, Z. Pan, X. Liu, W. Liu, D. Dai, H. Gao, Y. Ma, C. Wu, B. Wang, Z. Xie, Y. Wu, K. Hu, J. Wang, Y. Sun, Y. Li, Y. Piao, K. Guan, A. Liu, X. Xie, Y. You, K. Dong, X. Yu, H. Zhang, L. Zhao, Y. Wang, and C. Ruan, "Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding," *CoRR*, vol. abs/2412.10302, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2412.10302

[33] Y. Zhou, R. Ji, X. Sun, J. Su, D. Meng, Y. Gao, and C. Shen, "Plenty is plague: Fine-grained learning for visual question answering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 2, pp. 697–709, 2019.

[34] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 34 892–34 916. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf

[35] G. Luo, Y. Zhou, X. Sun, Y. Wu, Y. Gao, and R. Ji, "Towards language-guided visual recognition via dynamic convolutions," *International Journal of Computer Vision*, vol. 132, no. 1, pp. 1–19, 2024.

[36] G. Luo, Y. Zhou, X. Sun, Y. Wang, L. Cao, Y. Wu, F. Huang, and R. Ji, "Towards lightweight transformer via group-wise transformation for vision-and-language tasks," *IEEE Transactions on Image Processing*, vol. 31, pp. 3386–3398, 2022.

[37] G. Luo, Y. Zhou, M. Huang, T. Ren, X. Sun, and R. Ji, "Moil: Momentum imitation learning for efficient vision-language adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 7, pp. 5192–5204, 2024.

[38] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge *et al.*, "Qwen3-VL Technical Report," *CoRR*, vol. abs/2511.21631, 2025. [Online]. Available: https://doi.org/10.48550/ARXIV.2511.21631

[39] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach, "Towards VQA Models That Can Read," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 8317–8326. [Online]. Available: http://openaccess.thecvf.com/content_CVPR_2019/html/Singh_Towards_VQA_Models_That_Can_Read_CVPR_2019_paper.html

[40] A. Masry, D. X. Long, J. Q. Tan, S. R. Joty, and E. Hoque, "ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning," in *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, ser. Findings of ACL, vol. ACL 2022. Association for Computational Linguistics, 2022, pp. 2263–2279. [Online]. Available: https://doi.org/10.18653/V1/2022.FINDINGS-ACL.177

[41] L. Chen, J. Li, X. Dong, P. Zhang, Y. Zang, Z. Chen, H. Duan, J. Wang, Y. Qiao, D. Lin, and F. Zhao, "Are We on the Right Way for Evaluating Large Vision-Language Models?" in *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. [Online]. Available: http://papers.nips.cc/paper_files/paper/2024/hash/2f8ee6a3d766b426d2618e555b5aeb39-Abstract-Conference.html

[42] Y. Liu, H. Duan, Y. Zhang, B. Li, S. Zhang, W. Zhao, Y. Yuan, J. Wang, C. He, Z. Liu, K. Chen, and D. Lin, "MMBench: Is Your Multi-modal Model an All-Around Player?" in *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part VI*, ser. Lecture Notes in Computer Science, vol. 15064. Springer, 2025, pp. 216–233. [Online]. Available: https://doi.org/10.1007/978-3-031-72658-3_13

[43] C. Fu, P. Chen, Y. Shen, Y. Qin, M. Zhang, X. Lin, J. Yang, X. Zheng, K. Li, X. Sun, Y. Wu, R. Ji, C. Shan, and R. He, "MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models," in *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. [Online]. Available: http://papers.nips.cc/paper_files/paper/2025/hash/d79a27cf2772fe00be7f341efc0eb517-Abstract-Datasets_and_Benchmarks_Track.html

[44] xAI, "Grok-1.5 Vision Preview," Official release of the RealWorldQA benchmark, April 2024. [Online]. Available: https://x.ai/news/grok-1.5v

[45] K. Li, Y. Wang, Y. He, Y. Li, Y. Wang, Y. Liu, Z. Wang, J. Xu, G. Chen, P. Lou, L. Wang, and Y. Qiao, "MVBench: A Comprehensive Multi-modal Video Understanding Benchmark," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*. IEEE, 2024, pp. 22 195–22 206. [Online]. Available: https://doi.org/10.1109/CVPR52733.2024.02095

[46] K. Mangalam, R. Akshulakov, and J. Malik, "EgoSchema: A Diagnostic Benchmark for Very Long-form Video Language Understanding," in *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. [Online]. Available: http://papers.nips.cc/paper_files/paper/2023/hash/90ce332aff156b910b002ce4e6880dec-Abstract-Datasets_and_Benchmarks.html

[47] C. Fu, Y. Dai, Y. Luo, L. Li, S. Ren, R. Zhang, Z. Wang, C. Zhou, Y. Shen, M. Zhang, P. Chen, Y. Li, S. Lin, S. Zhao, K. Li, T. Xu, X. Zheng, E. Chen, C. Shan, R. He, and X. Sun, "Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*. Computer Vision Foundation / IEEE, 2025, pp. 24 108–24 118. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2025/html/Fu_Video-MME_The_First-Ever_Comprehensive_Evaluation_Benchmark_of_Multi-modal_LLMs_in_CVPR_2025_paper.html

[48] H. Wu, D. Li, B. Chen, and J. Li, "LongVideoBench: A Benchmark for Long-context Interleaved Video-Language Understanding," in *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. [Online]. Available: http://papers.nips.cc/paper_files/paper/2024/hash/329ad516cf7a6ac306f29882e9c77558-Abstract-Datasets_and_Benchmarks_Track.html

[49] K. Hu, P. Wu, F. Pu, W. Xiao, X. Yue, B. Li, Y. Zhang, and Z. Liu, "Video-MMMU: Evaluating Knowledge Acquisition from Multidisciplinary Professional Videos," in *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*. Association for Computational Linguistics, 2026, pp. 27 798–27 828. [Online]. Available: https://doi.org/10.18653/V1/2026.ACL-LONG.1281

[50] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved baselines with visual instruction tuning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 26 296–26 306.

[51] Y. Zhang, J. Wu, W. Li, B. Li, Z. Ma, Z. Liu, and C. Li, "Video instruction tuning with synthetic data," *arXiv preprint arXiv:2410.02713*, 2024. [Online]. Available: https://arxiv.org/abs/2410.02713

[52] Y. Jiang, Q. Wu, W. Lin, W. Yu, and Y. Zhou, "What kind of visual tokens do we need? training-free visual token pruning for multi-modal large language models from the perspective of graph," in *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, T. Walsh, J. Shah, and Z. Kolter, Eds. AAAI Press, 2025, pp. 4075–4083. [Online]. Available: https://doi.org/10.1609/aaai.v39i4.32427

[53] M. Mathew, D. Karatzas, and C. V. Jawahar, "DocVQA: A dataset for VQA on document images," *arXiv preprint arXiv:2007.00398*, 2020. [Online]. Available: https://arxiv.org/abs/2007.00398

[54] Y. Liu, Z. Li, M. Huang, B. Yang, W. Yu, C. Li, X. Yin, C. lin Liu, L. Jin, and X. Bai, "OCRBench: On the hidden mystery of OCR in large multimodal models," *arXiv preprint arXiv:2305.07895*, 2023. [Online]. Available: https://arxiv.org/abs/2305.07895

[55] P. Wu and S. Xie, "V*: Guided visual search as a core mechanism in multimodal LLMs," *arXiv preprint arXiv:2312.14135*, 2023. [Online]. Available: https://arxiv.org/abs/2312.14135

A Limitations

KAB-MoE derives a fixed visual expert skipping policy from a calibration set and directly applies it to all inputs during inference. This design improves deployment stability and avoids input-dependent search overhead, but it may be less adaptive to highly atypical inputs whose visual-token redundancy patterns differ from those in the calibration set. In addition, although our experiments cover two representative MoE-MLLM backbones and diverse image/video benchmarks, the learned policies need to be re-estimated when applied to substantially different model architectures or routing designs. We leave more adaptive policy transfer across broader MoE-MLLM families as future work.

B Impact Statements

This paper proposes KAB-MoE, a K-armed bandit based expert redundancy estimation framework for fixed visual expert skipping in MoE-MLLMs. We have not identified any direct negative societal impact caused by the proposed algorithm itself. Instead, we believe that KAB-MoE can bring a positive impact by improving the efficiency and deployability of large multimodal models. By deriving a static visual expert skipping policy under a predefined computation budget, KAB-MoE reduces redundant visual expert computation while preserving strong multimodal performance. This can lower the computational cost of MoE-MLLM inference, thereby helping reduce GPU energy consumption and the associated carbon footprint. Nevertheless, like other efficiency methods for large models, KAB-MoE does not by itself address potential misuse of the underlying multimodal models. Its broader impact, therefore, depends on the responsible deployment and governance of the base models to which it is applied.

C Ablation for Advantage-guided Sampling

We conduct an ablation study on Kimi-VL-A3B under a skipping ratio of 50% to examine the effect of advantage-guided sampling, with results reported in Tab. 5. The main observation is that uniform policy sampling provides a reasonable baseline, but it treats different layer-action choices equally and therefore cannot actively favor more promising expert allocations. Specifically, the uniform sampling baseline retains 99.29% average performance, while introducing advantage-guided sampling further improves the retention to 99.65%. This improvement indicates that the learned layer-action rewards can effectively guide the search toward better expert-number choices. Overall, these results confirm that advantage-guided sampling improves the quality of the derived fixed visual expert skipping policy.

Table 5: Ablation on advantage-guided sampling for KAB-MoE under 50% expert skipping on Kimi-VL-A3B. We compare uniform policy sampling with our advantage-guided sampling. Avg. denotes the average performance retention over the selected benchmarks.

Method	ChartQA	MMBench	VMME	VMMMU	Avg.(%)
Default ($k = 6^*$)	89.48	83.16	66.59	49.33	100.00
Baseline	89.28	82.13	65.67	49.33	99.29
+ Advantage Sampling	89.36	82.99	67.08	48.44	**99.65**

D Ablation for Monotonicity Constraint

We then conduct an ablation study on Kimi-VL-A3B under a skipping ratio of 83% to examine where the monotonicity constraint should be applied, with results reported in Tab. **??**. The first observation is that imposing monotonicity during the search stage is not beneficial. Specifically, when monotonicity is applied to sampled policies during reward estimation, the average retention is only 87.15%. In contrast, removing this search-stage constraint improves the average retention to 97.19%. This large gap suggests that enforcing monotonicity too early overly restricts the sampled policy space, preventing KAB-MoE from sufficiently evaluating diverse layer-action combinations. As a result, the learned reward matrix may fail to fully capture the relative benefits of retaining different numbers of visual experts across layers.

Table 6: Ablation on monotonicity constraints for Kimi-VL-A3B under 83% expert skipping. Search Mono indicates applying the monotonicity constraint during policy sampling for reward estimation, while Solving Mono indicates applying it during final policy solving. Avg. denotes the average performance retention over the selected benchmarks.

Setting	Search Mono	Solving Mono	TextVQA	ChartQA	MMBench	VideoMME	Avg. (%)
$k = 6^*$	–	–	88.70	89.48	83.16	66.59	100.00
Setting 1	✗	✗	84.9	86.68	81.79	65.15	97.19
Setting 2	✓	✗	77.50	64.52	77.41	63.96	87.15
Setting 3	✗	✓	85.22	86.48	82.05	65.04	97.27

In contrast, applying monotonicity only in the final policy solving stage leads to a more reasonable fixed policy. After estimating the reward matrix from an unconstrained and diverse policy space, the final monotonic solver achieves 97.27% average retention. This indicates that the monotonicity constraint can serve as a useful prior at the solving stage, regularizing the final expert allocation without interfering with reward estimation. Overall, these results show that introducing monotonicity only during final policy solving helps derive a stronger visual expert skipping policy.

E Detailed Algorithm

For completeness, we present the full reward matrix estimation procedure of KAB-MoE in Alg. 1. The algorithm follows the advantage-guided sampling strategy described in Sec. 3.2. Specifically, for each calibration example, KAB-MoE samples a group of budget-satisfying visual expert skipping policies, evaluates their KL-based discrepancy from the original model, and converts the relative discrepancy within the group into advantage rewards. These rewards are then assigned to the corresponding layer-action pairs to update the reward matrix and used for fixed policy derivation.

Algorithm 1 KAB-MoE Reward Matrix Estimation

Require: MoE-MLLM G, calibration set $\mathcal{D}_{cal}$, default expert number K, number of MoE layers L, target global skipping budget b, group size m, temperature τ, average visual/text token numbers $\bar{N}_v, \bar{N}_t$

Ensure: Reward matrix $\mathbf{R} \in \mathbb{R}^{L\times(K+1)}$

Initialize reward matrix $\mathbf{R} \leftarrow \mathbf{0}$ and count matrix $\mathbf{C} \leftarrow \mathbf{0}$
Define action space $\mathcal{A} = \{0, 1, \ldots, K\}$
function GLOBALSKIP(P)
    return $\dfrac{\bar{N}_v \sum_{\ell=1}^{L}(K - a_\ell)}{(\bar{N}_v + \bar{N}_t)LK}$
end function
for all $E \in \mathcal{D}_{cal}$ **do**
    Sample m non-duplicated visual policies $\{P_i\}_{i=1}^{m}$ according to $\pi_\ell(a) = \mathrm{Softmax}(\mathbf{R}_{\ell,a}/\tau)$
    Keep only policies satisfying GLOBALSKIP(P_i) $\geq b$
    for all P_i **do**
        Run $G(E \mid P_i)$ by applying P_i to visual experts and keeping text experts unchanged
        Compute policy discrepancy $\mathcal{L}(E, P_i) = \mathrm{KL}(G(E \mid P_0^*) \| G(E \mid P_i))$
    end for
    Compute group-average discrepancy $\bar{\mathcal{L}}(E) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}(E, P_i)$
    for all $P_i = \{a_1^i, \ldots, a_L^i\}$ **do**
        Compute advantage reward $A(E, P_i) = \bar{\mathcal{L}}(E) - \mathcal{L}(E, P_i)$
        for $\ell = 1$ to L **do**
            $\mathbf{C}_{\ell,a_\ell^i} \leftarrow \mathbf{C}_{\ell,a_\ell^i} + 1$
            $\mathbf{R}_{\ell,a_\ell^i} \leftarrow \mathbf{R}_{\ell,a_\ell^i} + \dfrac{A(E,P_i) - \mathbf{R}_{\ell,a_\ell^i}}{\mathbf{C}_{\ell,a_\ell^i}}$
        end for
    end for
end for
return $\mathbf{R}$

F Case Study

We provide qualitative examples to further illustrate the effectiveness of KAB-MoE under high expert skipping rates. As shown in Fig. 5, KAB-MoE produces predictions that are highly consistent with those of the default Top-K routing, even when a large portion of visual experts are skipped. The generated answers remain semantically equivalent, with only minor wording variations in some cases. These results suggest that visual experts, similar to visual tokens, contain substantial redundancy in MoE-MLLMs. Therefore, even after skipping most visual expert activations, KAB-MoE can still preserve the key visual semantics required for multimodal question answering, supporting the effectiveness of visual expert skipping for efficient inference.

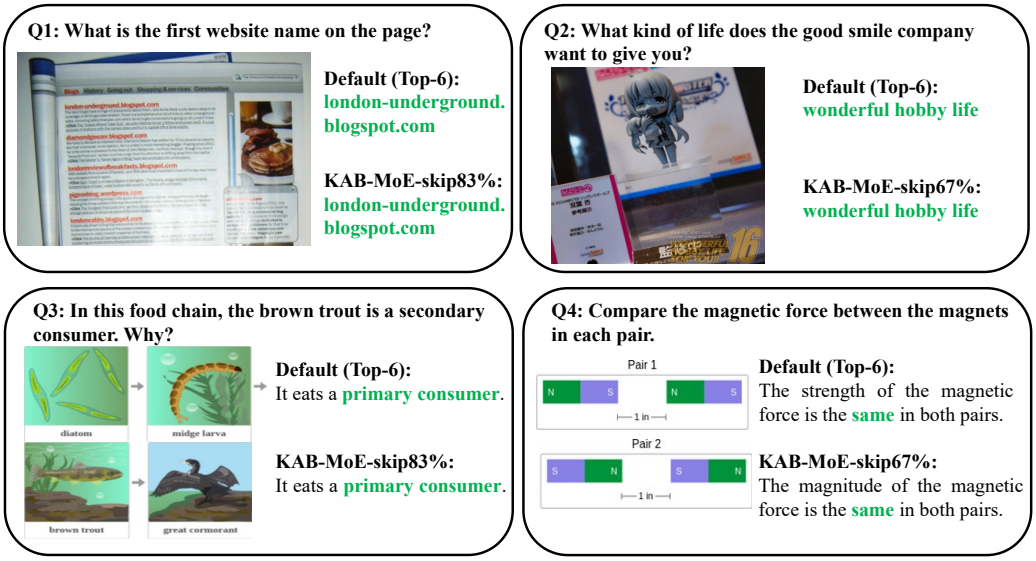


Figure 5: Qualitative case study comparing the default Top-K routing and KAB-MoE under high expert skipping rates. KAB-MoE produces predictions that are highly consistent with the full model, even with substantial visual expert skipping, indicating that a large portion of visual expert computation is redundant.

G Additional Experiments

G.1 Comparison with the Global Baseline

We compare KAB-MoE with direct global search, which searches for complete layer-wise skipping policies using the KL divergence from the full model's final output as the evaluation objective (Tab. 7). At the skipping ratio of 50% and 67%, KAB-MoE achieves an average performance retention of 99.67% and 98.75%, compared to 99.44% and 98.30% for global search. Global search requires approximately six times the search time of KAB-MoE. These results suggest that accumulated layer-action rewards provide an effective and efficient approximation for policy search, with cross-layer interactions reflected in the end-to-end output evaluation.

Table 7: Comparison with direct global search.

Method	TextVQA	ChartQA	MMStar	VMME	VMMMU	Avg. (%)
baseline	88.70	89.48	49.89	66.59	49.33	100.00
Global search (50%)	87.58	89.12	49.90	67.18	48.33	99.44
Ours (50%)	87.67	89.36	50.25	67.08	48.44	99.67
Global search (67%)	86.06	88.20	48.65	66.41	48.67	98.30
Ours (67%)	86.91	88.40	48.69	66.63	49.00	98.75

G.2 Robustness to Calibration Image Augmentation

We augment the calibration images and repeat policy search at the same 50% skipping ratio. The resulting policy reallocates only five expert slots across layers, corresponding to approximately 6.9% of the total budget, while retaining the overall allocation trend. As shown in Tab. 8, its average retention is 99.82%, compared with 99.99% for the original policy on the five benchmarks. The experiment indicates that the tested image perturbations have a limited effect on the allocation and downstream performance.

Table 8: Calibration augmentation (50% skipping).

Calibration	TextVQA	ChartQA	MMStar	MMB	VMME	Avg. (%)
baseline	88.70	89.48	49.89	83.16	66.59	100.00
Augmented	88.09	89.24	49.86	82.99	66.81	99.82
Original	87.67	89.36	50.25	82.99	67.08	99.99

G.3 Calibration with Mixed Image and Video Data

We recalibrate KAB-MoE at 50% skipping using a mixture of LLaVA-665K and LLaVA-Video data (50, 51). The mixed-calibration policy achieves 99.76% average retention, close to the 99.63% obtained with GQA in this comparison (Tab. 9). This comparison suggests that competitive performance can also be obtained from calibration data spanning both images and videos. Calibration remains an offline procedure in these experiments.

Table 9: Calibration data (50% skipping).

Calibration set	TextVQA	ChartQA	MMStar	VMME	VMMMU	Avg. (%)
baseline	88.70	89.48	49.89	66.59	49.33	100.00
Image + video	88.27	89.12	49.75	67.19	48.88	99.76
GQA	87.67	89.36	50.25	67.08	48.33	99.63

G.4 Performance at a Lower Skipping Ratio

We additionally evaluate Qwen3-VL-MoE at a 30% visual expert skipping ratio. On the five benchmarks in Tab. 10, the average retention is 100.67%, compared with 100.52% at 63% skipping. Thus, skipping visual experts can slightly exceed the default Top-8 performance on this subset. Reducing redundant or interfering expert contributions is one possible explanation; these results alone do not establish a regularization mechanism.

Table 10: Skipping ratios on Qwen3-VL-MoE.

Method	TextVQA	ChartQA	MMStar	VMME	VMMMU	Avg. (%)
baseline	83.41	85.08	59.67	54.89	47.11	100.00
Ours (63%)	82.81	84.36	58.09	54.89	50.33	100.52
Ours (30%)	83.22	85.08	59.44	55.26	48.67	100.67

G.5 Compatibility with Visual Token Pruning

Expert skipping and visual token pruning reduce computation along different dimensions. We combine KAB-MoE with G-Prune (52) as a compatibility test. As shown in Tab. 11, G-Prune alone retains 97.40% average performance, while its combination with KAB-MoE at 50% expert skipping retains 96.96%. The additional decrease is 0.44 percentage points. This experiment evaluates the combination of the two methods rather than joint optimization of their policies.

Table 11: G-Prune compatibility (50% skipping).

Method	TextVQA	ChartQA	MMStar	VMME	VMMMU	Avg. (%)
baseline	88.70	89.48	49.89	66.59	49.33	100.00
G-Prune	87.60	83.72	48.50	66.41	48.22	97.40
Ours (50%)	87.67	89.36	50.25	67.08	48.44	99.67
Ours (50%) + G-Prune	86.79	83.88	47.91	66.67	47.89	96.96

G.6 OCR and Fine-grained Visual Understanding

To examine the preservation of visual details, we evaluate DocVQA (53), OCRBench (54), and VSTAR (55). As shown in Tab. 12, KAB-MoE retains 100.13%, 99.59%, and 98.03% average performance at 50%, 67%, and 83% skipping, respectively. These results support the preservation of document reading, OCR, and fine-grained visual understanding on the tested benchmarks.

Table 12: OCR and fine-grained understanding.

Method	DocVQA	OCRBench	VSTAR	Avg. (%)
baseline	98.75	83.60	62.83	100.00
Ours (50%)	98.67	83.30	63.35	100.13
Ours (67%)	98.60	82.70	62.83	99.59
Ours (83%)	97.29	79.90	62.83	98.03

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have clearly stated our contributions and scope of the paper in Abstract and Introduction (Sec. 1).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: We have discussed the limitations of our work in Section A.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

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: This paper does not provide any theoretical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

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 (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the implementation in Method (Sec. 3), Experiment Setup and Implementation Details (Sec. 4.1), and Detailed Algorithm (Appendix E). Therefore, we believe that readers can replicate the proposed KAB-MoE by following our instructions.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the 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: [Yes]

   Justification: The code is released in the supplementary materials. The experiments are based on publicly available benchmarks and models, and the implementation details(e.g., Sec. 4.1) clearly specify the key settings needed to reproduce the reported results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

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) necessary to understand the results?

   Answer: [Yes]

   Justification: See Implementation Details (Sec. 4.1) and the Detailed Algorithm (Appendix E).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

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: Due to the limited computational resources, we conducted each experiment only once.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

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: [Yes]

   Justification: We present the time consumption of our method in Experimental Results (Sec. 4.2).

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and make sure that our research conform with it in every respect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed the social impacts in Impact Statements (Appendix B).

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

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., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: We are convinced that the paper does not pose any risk.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

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: [Yes]

Justification: The paper properly credits the creators or original owners of the assets used, such as code, data, and models. We also have correctly cited the relevant literature.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: This paper does not introduce any new assets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

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? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: Our proposed method does not involve LLM.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.