ECM: ENHANCING COMPRESSIBILITY OF QUANTIZED VISION ENCODER AND LLM FOR LARGE VISION-LANGUAGE MODELS

Weilan Wang[1,2], *Yu Mao*[2,*], *Dongdong Tang*[1,2], *Nan Guan*[1], *Chun Jason Xue*[2]

[1]City University of Hong Kong, [2]MBZUAI

ABSTRACT

Quantizing the large language model (LLM) in vision-language models (VLMs) is an effective approach to reducing memory size. However, quantizing only the LLM shifts the memory bottleneck to the vision encoder, particularly in lightweight models. This paper proposes ECM, an end-to-end quantization framework for VLMs that compresses both the vision encoder and the LLM. We first study the impact of quantization granularity on model compressibility and accuracy, and find that finer granularity improves compression at the cost of performance, motivating the need for adaptive strategies. ECM incorporates Adaptive Granularity Quantization and Weight Scaling to balance compression and accuracy. We further apply lossless compression to the quantized weights to maximize storage efficiency. Experiments show that ECM achieves 1.34× and 1.25× compression ratios for the vision encoder and LLM, respectively, reducing the memory usage by 80.3% of FP16 VLM, and 51.3% of LLM-quantized VLM on average.

***Index Terms*—** Vision Language Models, Quantization, Data Compression

1. INTRODUCTION

Large Vision-Language Models (VLMs) have achieved significant advancements in recent years, driven by the remarkable capabilities of large language models (LLMs) in multimodal reasoning and contextual understanding [1, 2, 3]. Despite their impressive capabilities, the practical deployment of VLMs on resource-limited devices is challenging due to the substantial model parameter size of both the vision and language models in VLMs.

VLMs integrate a vision encoder, which transforms images into visual tokens, with an LLM that jointly processes visual and textual inputs to generate language outputs. While model compression has proven effective in reducing LLM size, particularly through quantization [4, 5, 6], most prior efforts have focused exclusively on the LLM, as it typically dominates memory usage. However, when the LLM is quantized to low bit-widths, the FP16 vision encoder emerges as a significant memory bottleneck, especially in smaller VLMs ($\leq$3B). As shown in Fig. 1, the vision encoder accounts for 35% of the memory in VILA1.5-3B [7] and 77% in LLaVA-OneVision-0.5B [8], even after LLM quantization. It underscores the critical role of the vision encoder in overall memory consumption. In this work, we compress both the vision encoder and the LLM to enable greater memory savings for VLMs.

Lossless compression presents a promising approach to further reduce the memory size of quantized models [9, 10]. Compressibility, defined as the degree to which data can be reduced in size without information loss, directly impacts memory savings. Higher compressibility enables a greater reduction in model size. While quantized VLMs retain compressibility, the extent varies across different quantization methods, such as AWQ [4], GPTQ [11], MBQ [5], and SmoothQuant [6].

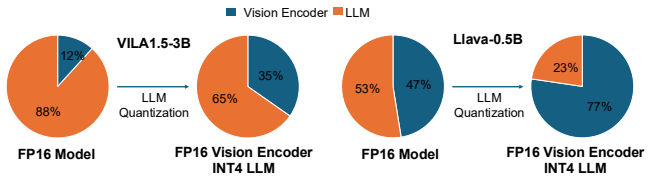

Fig. 1. Model sizes. Quantization on LLM can reduce the model size, but the vision encoder still takes a large memory size.

However, a naive application of quantization via compression often leads to compromised model performance despite reduced memory usage. Consequently, it results in a trade-off between compression ratio and accuracy. How to effectively mitigate the trade-off and find the connection between quantization design and compressibility remains underexplored. We conduct a comprehensive study across a range of quantization strategies. Our analysis reveals one key finding: **quantization granularity strongly affects both compressibility and performance** and larger granularity improves compression but tends to hurt accuracy.

Built upon our analysis, we propose ECM, an end-to-end compression framework of VLMs that targets both the LLM and the vision encoder. ECM integrates quantization via lossless compression to improve memory efficiency and enable lightweight deployment. To address the trade-off between accuracy and compressibility in quantized models, ECM introduces **Adaptive Granularity Quantization** (AGQ), which automatically selects the optimal quantization granularity for each weight. For the vision encoder, we further improve compressibility by applying a scaling-based normalization and removing the zero-point offset, reducing redundancy in the quantized weights. Experiments on multiple VLMs demonstrate that ECM achieves strong compressibility, with an average of 1.34x and 1.25x compression ratios for the vision encoder and LLM, while preserving accuracy. ECM reduces the memory usage by 80.3% of FP16 VLM, and 51.3% of LLM-quantized VLM on average, validating its effectiveness for efficient model deployment.

2. BACKGROUND AND RELATED WORK

Large Vision-Language Models (VLMs) typically consist of a vision encoder (e.g., CLIP or ViT) and a backbone LLM (e.g., LLaMA or GPT). As illustrated in Fig. 2, the input images are first processed by the vision encoder and then projected into a latent space compat-

*Corresponding author. Email: yu.mao@mbzuai.ac.ae

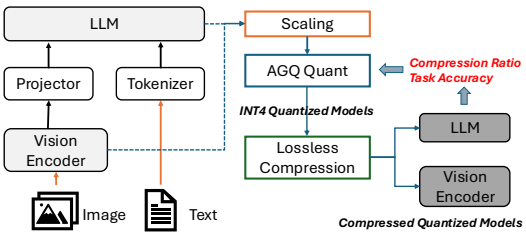

Fig. 2. Overview of VLM Compression via ECM.

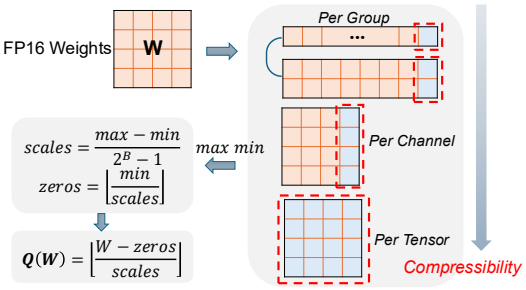

Fig. 3. Weight Quantization Process.

Methods	LLM CR	ACC	VE CR	IMG-ACC
FP16	-	69.89	-	68.96
per-tensor	3.39	00.00	2.25	56.62
per-channel	1.23	66.45	1.24	67.92
per-group	1.09	67.53	1.12	69.66
AWQ	1.06	67.67	-	67.98
SmoothQuant	3.52	00.00	-	56.61
MBQ	1.07	68.13	1.25	68.91

Table 1. The compression performance of INT4 quantized VILA1.5-3B model evaluated on ScienceQA task.

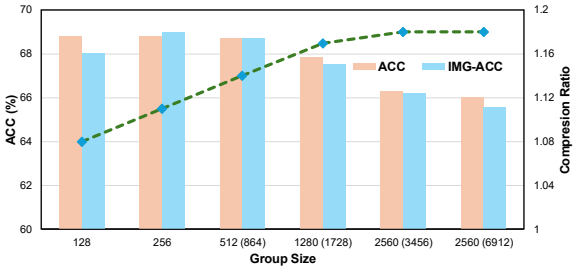

Fig. 4. VILA1.5-3B model performance and compressibility with different group sizes. Different sizes for different dimensions.

ible with the LLM's token embedding space. The projected image tokens, along with text tokens, are fed into the LLM for multimodal reasoning. Previous compression works mainly focus on reducing the LLM size.

Quantization reduces memory size via low-precision formats. GPTQ [11], Smoothquant and AWQ [4] are widely used LLM quantization methods. QSLAW [12], ClusComp [13] and MBQ [5] are designed for VLMs considering the multiple modalities. But all of these methods compress the model with a fixed ratio, and the quantized model can actually be further compressed. Existing methods can be broadly categorized into three types: per-tensor, per-channel, and per-group with variable group sizes as shown in Fig.3. Quantized weights $Q(W)$ are computed using per-group scaling factors $scales$ and zero-points $zeros$, derived from the max/min within each group. For per-channel quantization, $zeros$ and $scales$ are computed for each channel (each row of the tensor), while per-tensor quantization shares the same across the entire tensor. The compressibility of these methods will be evaluated in Section 3.1.

Some works also apply lossless compression for LLM. Zipnn [14], NeuZip [15] and Huff-LLM[16] compress FP16 model with limited Compression Ratio. Deep Compression [10], Double Compression [9] and Analysis [17] try to compress INT8 quantized LLMs. But these methods are applied to higher bit precision and do not consider VLMs. In this paper, we focus on INT4 precision-the lowest bitwidth that maintains acceptable model performance, which is more challenging than FP16 and INT8 data compression.

3. DESIGN OF ECM

We first study how quantization methods affect compression, finding that settings and scaling impact compressibility differently in LLMs and vision encoders. ECM is proposed as a new compression framework designed to enhance the compressibility of quantized LLM and vision encoder in VLMs, as shown in Fig. 2. The new quantization method AGQ is designed for better compression ratio and task accuracy.

3.1. Preliminary Study of Model Compressibility

We first quantize VILA1.5-3B with different quantization methods and compress the quantized model using the zstd [18] compression algorithm. The results are shown in Table 1. The compression ratio (CR) is original INT4 size/compressed size and accuracy (ACC) is evaluated on ScienceQA [19]. Since the quantization of the vision encoder only affects the image processing, we only report the image accuracy.

Quantized LLM. Different quantization settings exhibit varying performance in compressibility and accuracy. Per-tensor quantization achieves the highest CR (up to $3.39\times$), but with most weights quantized to zero, the model loses the ability for QA and we set the accuracy as 0. Per-group quantization with a group size of 128 delivers the best model performance while maintaining the lowest CR. Per-channel quantization exhibits a moderate performance decrease. Quantization methods with scaling also modify the compressibility as the scaling operation changes model weight distribution. Smoothquant with per tensor quantization and fixed scaling factor achieves the highest CR, while significantly sacrificing the model accuracy. Conversely, for MBQ and AWQ, which utilize per-group quantization with a dynamic scaling factor, the compressed VLMs have a relatively higher accuracy at the cost of a limited CR. *Achieving high compressibility of INT4 quantized model while maintaining accuracy remains a significant challenge for LLMs.*

Quantized Vision Encoder. Similar to LLM, per-tensor quantization of the VE also achieves the best compression with largest accuracy loss. In contrast, per-group quantization enhances model performance on image input tests, though it results in the lowest compression ratio. Quantization does not lead to a significant performance drop; instead, it even improves performance on the per-group quantized model. This suggests that the vision encoder is not particularly sensitive to quantization, possibly due to the redundancy in vision tokens. Notably, the VEs typically achieves a higher compression ratio compared to LLMs without experiencing substantial accuracy degradation.

Quantization Granularity Analysis. Based on above results, we

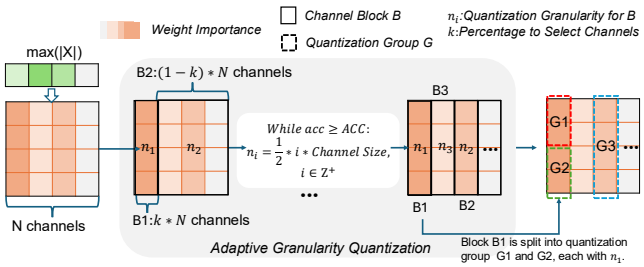

Fig. 5. Adaptive Granularity Quantization. Each weight matrix is divided into several channel blocks, and within each block per-group quantization is applied with granularity n_i.

observe that quantization granularity have a greater impact on model compressibility compared to scaling operations. We apply a more fine-grained settings to VILA1.5-3B as shown in Fig. 4. As the quantization group size increases, the compression ratio rises at the cost of greater performance loss, which can be attributed to quantization granularity. It determines the max and min used to calculate scales and zero-points, which in turn influence the quantized model weights and overall performance.

This effect stems from their impact on weight distributions, as prior studies have shown that greater distribution unevenness enables higher compression ratios [20, 21]. *Larger quantization granularity improves compressibility by increasing distribution unevenness, but degrades model performance.* How to decide these factors is challenging to balance model compressibility and performance.

3.2. Adaptive Granularity Quantization

All of the previous quantization methods use a fixed granularity for one weight, limiting the balance between compressibility and accuracy. To overcome this limitation, we propose Adaptive Granularity Quantization (AGQ), a novel method that dynamically tailors the quantization granularity for each weight based on empirical evaluations.

The basic idea of AGQ is that the quantization granularity of one weight can be configured at different levels for different channels based on the channel's importance. Inspired by the finding [4] that channel importance correlates with activation distribution, AGQ protects salient weights, which are critical for maintaining accuracy, by grouping channels based on their maximum activation values ($max(|X|)$). This strategy ensures high-saliency channels are quantized with high precision, thereby preserving model performance more effectively than uniform quantization with fixed granularity. For each model, we evaluate multiple quantization granularities, ranging from fine granularity (group size smaller than one channel) to coarse granularity (grouping several channels). When benchmarking model performance, we progressively increase the granularity levels until the accuracy loss is non-negligible. Based on these results, we determine the optimal granularity for each weight.

As demonstrated in Fig. 5, the weight is divided into several channel blocks $B_1, B_2, ...B_i$. Each block is assigned a quantization granularity n_i , forming B/n_i quantization groups for the block. Firstly, the N channels are ranked based on activation $max(|X|)$. The top k percentage of the total channels is assigned with n_1 and the rest of the channels are quantized with granularity n_2. k ranging from 1% to 10% are tested to enhance the model performance. The granularity n progressively increases (e.g., one channel, two channels, ...) as long as the accuracy loss introduced by quantization error remains acceptable ($acc >= ACC$). If the accuracy loss is not negligible, we stop at the largest granularity that provides the best compressibility under the given accuracy constraint. With this adaptive granularity quantization method, AGQ effectively balances model compressibility and accuracy performance.

3.3. Customized Optimization for Vision Encoder

The vision encoder is more robust to compression than the LLM as the visual token redundancy. Since LLM quantization error causes most performance loss, we can compress the vision encoder more aggressively. In ECM, we apply two further steps to it: 1)Disable the zero-point; 2) Scale its weight. While asymmetric quantization with zero-point is widely adopted to minimize accuracy loss, our experiments reveal that symmetric quantization maintains the accuracy performance of vision encoders while enabling higher compressibility, as shown in Table 4. By utilizing an image calibration dataset, the activation values of the vision encoder can be obtained. Multiplying the weights of the vision encoder by a scaling factor derived from these activation values can further enhance the model's compressibility. The scaling factor is set to a value between 0 and 1.0 with a 0.1 step, within 1% accuracy loss.

Model	Method	AVG ACC	LLM CR	VE CR	vs. INT4*
LLaVA -Onevision -0.5B	FP16	52.76	-	-	-
	AWQ+C	49.72	1.10	-	2.2%
	MBQ+C	50.12	1.10	1.22	62.8%
	ECM	49.17	1.23	1.32	66.2%
InternVL2 -2B	FP16	62.77	-	-	-
	AWQ+C	62.11	1.09	-	5.4%
	MBQ+C	62.50	1.09	1.19	33.2%
	ECM	61.37	1.22	1.33	40.3%
Qwen2 -VL-2B	FP16	58.75	-	-	-
	AWQ+C	57.65	1.07	-	2.3%
	MBQ+C	57.89	1.07	1.21	53.3%
	ECM	57.20	1.30	1.34	60.5%
VILA1.5 -3B	FP16	55.59	-	-	-
	AWQ+C	54.87	1.06	-	4.0%
	MBQ+C	54.68	1.07	1.25	27.5%
	ECM	54.19	1.26	1.37	38.1%

Table 2. Performance evaluation and lossless compression gain on INT4 quantized models. *Memory Reduction compared with INT4 LLM quantized VLM.

4. EVALUATIONS

4.1. Evaluation Settings

Models. ECM is evaluated on several vision-language models, including LLaVA-Onevision [8], InternVL2 [22], Qwen2-VL [23] and VILA [7] families. We select both small-size and large-size models from each family to illustrate the performance of ECM.

Datasets. For the scaling operation, we use the COCO [1] calibration dataset containing both images and text. We evaluate the performance of the quantized model using LMMs-Eval [24] across multiple vision-language benchmarks. For text recognition and comprehension, we use OCRBench [25] and TextVQA [26]. Regarding visual perception, VizWiz [27] and SEED-Bench [28] are used for evaluation. For visual reasoning, we use ScienceQA [19] and MMMU [29].

Model	LLM CR	VE CR	ScienceQA*
LlaVA-Onevision-7B	1.29	1.32	84.08/85.40
InternVL2-8B	1.26	1.33	96.20/96.20
Qwen2-VL-7B	1.30	1.34	82.57/85.10
VILA1.5-8B	1.28	1.37	70.67/71.15

Table 3. Performance evaluation of large vision-language models on ScienceQA(*ECM/FP16). ECM achieves better compressibility for larger models.

Baselines. We compare ECM with AWQ [4] and MBQ [5], two state-of-the-art methods for VLM quantization. Both AWQ and MBQ employ INT4 per-group quantization with a group size of 128 and we apply further compression for them (+C). For the quantized model compressibility, we present the Compression Ratio (CR), which is calculated by quantized model size/compressed model size.

4.2. Compression Results

As shown in Table 2, ECM improves the compression ratio of the quantized Vision Encoder and LLM, thereby lowering VLM memory usage while maintaining comparable performance. ECM achieves an average compression ratio of 1.25 for LLM. Compared with AWQ [4] and MBQ [5], ECM achieves a 1.16× average improvement and up to 1.21× in CR. AWQ [4] and MBQ [5] have similar CR because both of them apply per-group quantization with a group size of 128. In terms of vision encoder, ECM achieves an average compression ratio of 1.34, achieving significant memory savings for VLMs, where the vision encoder constitutes a substantial portion of the total model size. ECM reduces memory usage by up to 80.9% on FP16 models with minimal accuracy loss, and by 51.3% on LLM-quantized VLMs, where the vision encoder remains FP16 and the LLM is quantized to INT4.

4.3. Analysis of Efficiency and Accuracy

Performance on Large Models. To explore the ECM efficiency on large VLMs, we also evaluate the 7B or 8B models on ScienceQA tasks. As shown in Table 3, ECM achieves an average CR of 1.34 for the vision encoder and an average CR of 1.28 for LLM compared with INT4 quantized model. ECM compresses the model InternVL2-8B [22] without accuracy loss. Other compressed models also maintain comparable model performance with the FP16 model. It's noticeable that large VLMs have higher compressibility (higher compression ratio with less performance loss) compared with small VLMs. ECM is efficient for large VLMs and substantially reduces memory overhead.

The Effect of ECM Methods. The quantization settings affect the compressibility and performance of the model. We evaluated VILA1.5-3B to present the impact of each setting and the results are shown in Table 4. For the vision encoder, AGQ can improve the CR by 0.12. Zero-point disabling can increase the CR from 1.24 to 1.32 without impacting the model performance. Scaling the weight can further improve the CR to 1.37 with negligible accuracy loss. For LLM, AGQ significantly improves the CR from 1.09 to 1.26. These results indicate that ECM offers a viable solution for efficient model compression with minimal performance trade-offs.

Performance Trade-off. As AGQ provides different quantization options for VLMs, the compression and accuracy performance vary according to different settings. We show the performance trade-off in Fig. 6. Existing fixed-granularity methods: per-tensor (purple square) and per-channel quantization (green rhombus) achieve good CR but severe accuracy degradation; per-group methods (orange triangles) preserve accuracy well with limited compression. ECM achieves a flexible trade-off (the curve) and performs better than fixed quantization. We select the optimal point (red star) that delivers better CR while maintaining accuracy.

Model	Components			CR	ScienceQA
	AGQ	Zero-point	Scaling		
VE	✗	✗	✗	1.12	70.22
	✓	✗	✗	1.24	69.39
	✓	✓	✗	1.32	69.70
	✓	✓	✓	1.37	69.58
LLM	✗	-	-	1.09	67.53
	✓	-	-	1.26	66.39

Table 4. The effect of ECM Methods. INT4 Quantized VILA1.5-3B Compression Performance with different settings is evaluated on the ScienceQA image task.

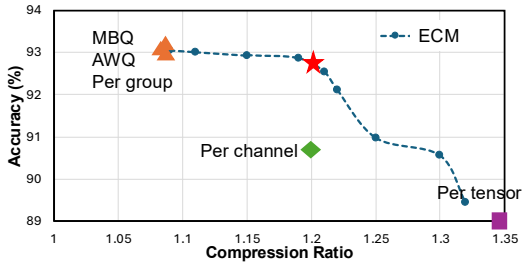

Fig. 6. Performance of InternVL2-2B on ScienceQA. ECM achieves a better trade-off between accuracy and compression ratio .

Model	GPU	Inference Latency (ms)		
		FP	Compressed	Decompress Speed(GB/s)
VILA-1.5-3B	RTX4090	5.26	5.32	121.32
	RTX2080	34.42	29.47	78.41
VILA-1.5-8B	RTX4090	16.12	16.93	177.42
	RTX2080	OOM	49.24	81.23
		Memory Size (GB)		
		FP	Compressed	Saving
VILA-1.5-3B	RTX4090	7.2	1.4	**80.6%**
VILA-1.5-8B	RTX4090	17.4	3.2	**81.6%**

Table 5. Inference Latency and Memory Size for VILA models.

Speedup and Memory Saving. ECM is designed to compress VLMs, reducing GPU memory usage for deployment on low-resource devices while preserving accuracy and inference speed. The inference latency and decompression speed are evaluated on two limited-resource GPUs: Nvidia RTX4090 and RTX2080. With high decompression speed, the compressed models have comparable inference latency with FP models, while delivering over 5× memory savings as shown in Table 5.

5. CONCLUSION

We propose that ECM compress the quantized models for both the vision encoder and LLM of VLMs. We analyze the compressibility of various quantization methods and discover that quantization granularity significantly affects both compressibility and performance. A new quantization method AGQ is designed to achieve optimal compressibility without compromising model performance. The evaluation results demonstrate the effectiveness of ECM.

6. REFERENCES

[1] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick, "Microsoft coco captions: Data collection and evaluation server," *arXiv preprint arXiv:1504.00325*, 2015.

[2] Stanislaw Antol, Aishwarya Agrawal, Lu, et al., "Vqa: Visual question answering," in *Proceedings of the international conference on computer vision*, 2015, pp. 2425–2433.

[3] Jean-Baptiste Alayrac, Jeff Donahue, Luc, et al., "Flamingo: a visual language model for few-shot learning," *Advances in neural information processing systems*, vol. 35, pp. 23716–23736, 2022.

[4] Ji Lin, Jiaming Tang, Tang, et al., "Awq: Activation-aware weight quantization for on-device llm compression and acceleration," *Proceedings of Machine Learning and Systems*, vol. 6, pp. 87–100, 2024.

[5] Shiyao Li, Yingchun Hu, Ning, et al., "Mbq: Modality-balanced quantization for large vision-language models," *arXiv preprint arXiv:2412.19509*, 2024.

[6] Guangxuan Xiao, Ji Lin, Seznec, et al., "Smoothquant: Accurate and efficient post-training quantization for large language models," in *International Conference on Machine Learning*. PMLR, 2023, pp. 38087–38099.

[7] Zhijian Liu, Ligeng Zhu, Shi, et al., "Nvila: Efficient frontier visual language models," *arXiv preprint arXiv:2412.04468*, 2024.

[8] Bo Li, Yuanhan Zhang, Guo, et al., "Llava-onevision: Easy visual task transfer," *arXiv preprint arXiv:2408.03326*, 2024.

[9] Weilan Wang, Yu Mao, Tang, et al., "When compression meets model compression: Memory-efficient double compression for large language models," in *The 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics (ACL), 2024, pp. 16973–16983.

[10] Song Han, Huizi Mao, and William J Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[11] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh, "Gptq: Accurate post-training quantization for generative pre-trained transformers," *arXiv preprint arXiv:2210.17323*, 2022.

[12] Jingjing Xie, Yuxin Zhang, Mingbao Lin, Liujuan Cao, and Rongrong Ji, "Advancing multimodal large language models with quantization-aware scale learning for efficient adaptation," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 10582–10591.

[13] Baohao Liao, Christian Herold, Seyyed Hadi Hashemi, Stefan Vasilev, Shahram Khadivi, and Christof Monz, "Cluscomp: A simple paradigm for model compression and efficient finetuning," *arXiv preprint arXiv:2503.13089*, 2025.

[14] Moshik Hershcovitch, Andrew Wood, Leshem Choshen, Guy Girmonsky, Roy Leibovitz, Ilias Ennmouri, Michal Malka, Peter Chin, Swaminathan Sundararaman, and Danny Harnik, "Zipnn: Lossless compression for ai models," *arXiv preprint arXiv:2411.05239*, 2024.

[15] Yongchang Hao, Yanshuai Cao, and Lili Mou, "Neuzip: Memory-efficient training and inference with dynamic compression of neural networks," *arXiv preprint arXiv:2410.20650*, 2024.

[16] Patrick Yubeaton, Tareq Mahmoud, Shehab Naga, Pooria Taheri, Tianhua Xia, Arun George, Yasmein Khalil, Sai Qian Zhang, Siddharth Joshi, Chinmay Hegde, et al., "Huff-llm: End-to-end lossless compression for efficient llm inference," *arXiv preprint arXiv:2502.00922*, 2025.

[17] Yu Mao, Weilan Wang, Hongchao Du, Nan Guan, and Chun Jason Xue, "On the compressibility of quantized large language models," *arXiv preprint arXiv:2403.01384*, 2024.

[18] Facebook, "Zstandard," 2016, `https://github.com/facebook/zstd`.

[19] Pan Lu, Swaroop Mishra, Xia, et al., "Learn to explain: Multimodal reasoning via thought chains for science question answering," *Advances in Neural Information Processing Systems*, vol. 35, pp. 2507–2521, 2022.

[20] Khalid Sayood, *Introduction to data compression*, Morgan Kaufmann, 2017.

[21] Guy E Blelloch et al., "Introduction to data compression," *Computer Science Department, Carnegie Mellon University*, vol. 54, 2001.

[22] Zhe Chen, Jiannan Wu, Wang, et al., "Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks," in *Proceedings of the CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 24185–24198.

[23] Jinze Bai, Shuai Bai, Yang, et al., "Qwen-vl: A frontier large vision-language model with versatile abilities," *arXiv preprint arXiv:2308.12966*, 2023.

[24] Kaichen Zhang, Bo Li, Zhang, et al., "Lmms-eval: Reality check on the evaluation of large multimodal models," *arXiv preprint arXiv:2407.12772*, 2024.

[25] Yuliang Liu, Zhang Li, Biao Yang, Li, et al., "On the hidden mystery of ocr in large multimodal models," *arXiv preprint arXiv:2305.07895*, 2023.

[26] Amanpreet Singh, Vivek Natarajan, Shah, et al., "Towards vqa models that can read," in *Proceedings of the CVF conference on computer vision and pattern recognition*, 2019, pp. 8317–8326.

[27] Danna Gurari, Qing Li, Stangl, et al., "Vizwiz grand challenge: Answering visual questions from blind people," in *Proceedings of the conference on computer vision and pattern recognition*, 2018, pp. 3608–3617.

[28] Bohao Li, Rui Wang, Wang, et al., "Seed-bench: Benchmarking multimodal llms with generative comprehension," *arXiv preprint arXiv:2307.16125*, 2023.

[29] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al., "Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi," in *Proceedings of the CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9556–9567.