Proceedings of CONF-CIAP 2027 Symposium: Applications of Complex Analysis, Optimization and Integrability to Physical Theories
DOI: 10.54254/2753-8818/2027.TA39198

The Mathematical Foundation and Optimization Methods of the Core Modules of Multimodal Large Models

Zhuoxuan Cai

School of Mathematics and Statistics, Northeastern University at Qinhuangdao, Qinhuangdao, China
13998267813@163.com

Abstract. Multimodal large models integrate language, vision and other input modalities through attention, alignment, connector mapping and optimization modules, and their capabilities depend not only on parameter scale but also on how these components exchange and transform information. This review examines the mathematical foundations of scaled dot-product attention, contrastive learning, positional encoding, low-rank adaptation, quantization and knowledge distillation. Attention scaling is explained through variance control, InfoNCE through mutual-information estimation, and parameter-efficient adaptation through low-rank approximation. Efficient attention computation and activation-aware quantization are also analysed from computational and numerical perspectives. Together, these connections clarify how architecture and computation shape stability, efficiency and reliability, and they identify directions for more interpretable and controllable multimodal systems.

Keywords: Multimodal Large Models, Mathematical Foundations, Attention Mechanism, Cross-Modal Alignment, Efficient Model Adaptation

1. Introduction

Multimodal large models integrate text, images, audio and other signals into a shared computational framework. These inputs have distinct statistical properties and therefore often require different encoders and mapping functions. Unlike text-only systems, multimodal models must match heterogeneous representations while supporting understanding and inference, so system performance depends not simply on model size but on cross-modal semantic alignment, feature fusion and efficient computation. Transformers provided a matrix formulation for modelling long-range dependencies and a shared architecture for visual and language processing [1]. CLIP aligned images and text through large-scale contrastive learning, enabling zero-shot recognition and cross-modal retrieval [2]. BLIP-2 connected frozen image encoders to language models through a small Q-Former and showed that useful visual information could be transferred without retraining either backbone [3]. FlashAttention reduced memory traffic through I/O-aware computation while preserving exact attention results [4]. Knowledge distillation (KD) transferred predictive information from larger models to smaller students [5]. These advances motivated the compressed deployment and efficient training strategies discussed in this review.

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2027.TA39198

Several surveys have described the architectures, datasets, benchmarks and applications of multimodal models [6, 7], and theoretical work has linked InfoNCE-style objectives to mutual-information estimation [8]. However, these lines of research have not accounted, within a single framework, for how attention scaling controls score variance, how contrastive losses produce semantic alignment, or why low-rank updates suffice for a large model. The questions are closely related, because the performance of deep learning is determined by the combination of representation structure, learning objective and optimization algorithm [9]. This review therefore covers the mathematical foundations and optimization principles of the core multimodal modules, with emphasis on representation mapping, attention, alignment and efficient adaptation.

2. Core modules and infrastructure

A characteristic multimodal system combines a modality encoder, a cross-modal connector and a backbone language model. The encoder extracts compact features from images, audio or other inputs. The connector maps these features into representations compatible with language embeddings, using linear projections, multilayer perceptrons, Q-Formers or cross-attention. The language model then conditions inference and generation on visual and textual tokens. In a vision-language framework, image patches first become visual tokens through a visual encoder; after connector mapping, these tokens were supplied as prefix information or soft prompts alongside the text sequence [3]. The stages therefore form an ordered transformation pipeline rather than an undifferentiated model.

The encoder and connector can be formulated as a composition of mappings between visual and textual representations:

$$h_v = f_v(x_v), \quad h_t = f_t(x_t), \quad z_v = g(h_v) \tag{1}$$

Here h_v denotes the visual features produced by the visual encoder, $g(\cdot)$ the cross-modal connector mapping, z_v the visual representation after connector conversion, and h_t the representation after text encoding; x_v and x_t denote the visual and textual inputs, while f_v and f_t are the corresponding encoding mappings.

The connector maps visual features into the representation space of the language model. Semantically compatible image and text representations then allow the backbone to condition generation on the relevant visual information, which involves vector embeddings, projection matrices and efficient conditional prediction. CLIP performed similarity search between image and text embeddings in a dual-encoder model, and BLIP-2 employed learned queries to extract task-relevant visual information from a frozen encoder [2, 3]. Both showed that successful integration requires a representation interface that preserves the differences among heterogeneous signals rather than simply concatenating visual and textual features.

Poor alignment can propagate across modules. A language model may depend too strongly on pre-trained linguistic expectations, and noise or missing detail in the encoder output may be compressed and passed on by the connector. At the same time, weak evidence constraints may yield fluent but factually inaccurate answers. An assessment of multimodal models should therefore analyse intermediate representations, the distortion introduced by mappings, and final task performance; this distinction is especially important when claimed outputs are checked against visual evidence.

DOI: 10.54254/2753-8818/2027.TA39198

3. The mathematical basis of the attention mechanism

For query, key and value matrices, scaled dot-product attention is defined as:

$$\text{Attention}\left(Q,K,V\right)=\text{softmax}\left(\frac{QK^{T}}{\sqrt{d_k}}\right)V \tag{2}$$

Here Q denotes the query matrix, K the key matrix, V the value matrix and d_k the dimension of each key vector, while $QK^{\top}$ is the similarity matrix obtained between the query and key matrices after transposition.

The dimension of the key vectors determines the statistical scale of the attention scores. Suppose that the components of a query and a key are approximately independent, each with zero mean and unit variance. Under these assumptions the variance of their unscaled dot product grows linearly with the key dimension, and dividing by the square root of that dimension approximately normalizes the score variance:

$$\text{Var}\left(q\cdot k\right)=d_k,\qquad \text{Var}\left(\frac{q\cdot k}{\sqrt{d_k}}\right)\approx 1 \tag{3}$$

Here d_k denotes the dimension of the key vector: the larger the dimension, the more easily the variance of the dot product between a query vector and a key vector grows, so square-root scaling is necessary to maintain numerical stability. When the amplitude of the dot product is too large, the softmax output approaches a highly concentrated distribution, with gradients concentrated at a few positions, which increases training instability. Dividing by the square root of the key dimension can therefore be regarded as variance normalization that keeps the attention scores within an appropriate numerical range [1]. The design embodies the fundamental idea of maintaining activation-scale stability in deep networks [9].

Multi-head attention projects the inputs into several lower-dimensional subspaces, computes attention separately in each and aggregates the outputs, so that different heads highlight different relationships in the context. A useful representation must balance local visual evidence against global semantic information. Overly homogeneous attention can hide salient details in multimodal tasks, whereas prematurely high attention discards information from the broader context. From an information-theoretic perspective, the entropy of the attention weights describes how widely a head distributes its mass: high entropy corresponds to a spread-out allocation of attention, and low entropy indicates that few tokens receive most of the focus.

For images, attention models the relationships among patches, and cross-attention or token concatenation links visual and textual inputs [3]. Attention therefore acts as an information-routing mechanism that determines which visual evidence is used during generation.

4. Cross-modal alignment and connector mechanism

Cross-modal alignment draws semantically matched image-text embeddings together while separating unrelated pairs. For normalized embeddings, their cosine similarity is:

$$s\left(v_i,t_j\right)=\frac{{v_i}^{T}t_j}{\|v_i\|_2\|t_j\|_2} \tag{4}$$

DOI: 10.54254/2753-8818/2027.TA39198

For a batch of several paired image-text samples, the image-to-text InfoNCE loss is formulated as:

$$L_{i\to t} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(v_i,t_i)/\tau}}{\sum_{j=1}^{N}e^{s(v_i,t_j)/\tau}} \tag{5}$$

Here v_i denotes the embedding of the i -th image sample, t_i the embedding of the i -th text sample and N the number of samples in the current training batch, while τ is the temperature parameter that adjusts the smoothness of the similarity distribution and t_j the embedding of the j -th text sample in the denominator of (5).

The temperature parameter controls the sharpness of the similarity distribution and the scale of the gradients, which supports retrieval [2]; a theoretical lower bound of the objective is given by the mutual information [8]. The behaviour of this objective depends mainly on the number and quality of negative samples, on batch composition and on the temperature. Alignment performance therefore depends not only on dataset size but also on data sampling and on quantitative decisions such as these.

Connectors translate encoded visual features into language-compatible tokens. Multilayer perceptrons offer greater expressiveness at the cost of additional computation and a risk of over-fitting, whereas linear projections use relatively few parameters and are simple to train but cannot capture complex nonlinear queries. A Q-Former learns such queries and uses them to select task-relevant visual information as a task-dependent bottleneck [3]. It does not attempt to reconstruct all the original visual information but to eliminate redundancy without discarding the information that language generation requires.

Cross-modal alignment can also be viewed geometrically as mapping the image and text data manifolds into a shared representation space. Cosine similarity and dot products offer tractable objectives, but they do not necessarily represent complex local geometry: matched samples should remain close to each other while modality-specific traits remain distinct. Riemannian metrics, optimal transport or neighbourhood-preserving maps could provide rigorous tools for describing alignment quality and generalization error, although these theoretical explanations remain an open research direction.

5. Modal coding, positional representation, and backbone language model

The tokens are converted into embeddings by the text encoder, and two-dimensional image patches are projected into feature vectors by the visual encoder. Because Transformers have no inherent notion of sequence order, the order must be supplied to both the input and the output through explicit positional representations [1]. Sinusoidal encoding uses trigonometric functions at many frequencies and can represent both absolute and relative positions.

Alternatively, the Fourier feature map can be regarded as a fixed function, while learned encoding adapts the position vectors to the data. For visual inputs, positional design must also consider two-dimensional relations, image scale and local neighbourhoods.

The language backbone handles inference and generation. LLaVA-1.5 used visual instruction tuning for image-based dialogue [10], MiniGPT-4 combined visual inputs with a large language model [11], and open backbones such as Llama 2 supplied language priors through pre-training and fine-tuning [12].

Language backbones receive connector-produced visual tokens rather than raw image pixels, so their reliability rests on both encoder quality and the ability of the connector to preserve relevant

DOI: 10.54254/2753-8818/2027.TA39198

evidence. Pre-trained language priors help models form clear answers and apply broad reasoning patterns to the visual setting, yet when an encoder misses information or the connector loses detail, those priors can dominate and introduce descriptions unsupported by the image. Such hallucinations highlight the need for bounds that link the output distribution to the observed evidence, especially for questions that require precise visual grounding.

6. Optimization methods corresponding to core modules

Increasing model size raises training and inference costs, so low-rank adaptation freezes the pre-trained weights and parameterizes the update through two smaller matrices:

$$W' = W + \Delta W = W + BA, \qquad \operatorname{rank}(BA) \le r \tag{6}$$

Here W denotes the weight matrix of the original model, ΔW the parameter update, B and A two trainable matrices of the low-rank decomposition, and r the chosen rank, which is much smaller than the dimensions of the original weight matrix; W' is the updated weight matrix.

When task adaptation lies mostly in a low-dimensional subspace, low-rank updates can approximate beneficial parameter changes with far fewer trainable parameters [13].

Standard attention materializes large similarity and softmax matrices, so memory requirements grow quadratically with sequence length. FlashAttention used blockwise tiling and recomputation to reduce transfers to high-bandwidth memory while preserving exact attention results [4], and FlashAttention-2 further improved parallelism and work partitioning for better use of GPU resources [14]. These approaches did not change the mathematical attention function; instead, they reorganized the same matrix operations around the memory hierarchy, showing how algorithmic efficiency can be improved through systems-aware numerical computation.

Quantization reduces model storage and inference cost by mapping floating-point weights to lower-precision integers. Affine quantization and dequantization are formulated as:

$$\mathrm{q} = \operatorname{clip}\left(\operatorname{round}\left(\frac{w}{s}\right) + z, q_{min}, q_{max}\right), \ \hat{w} = s\left(q - z\right) \tag{7}$$

Here w denotes the original floating-point weight, q the quantized integer weight, s the scaling factor and z the zero point that completes the mapping between floating-point and integer values; $\widehat{w}$ is the dequantized weight, and q_{min} and q_{max} are the lower and upper integer bounds.

The scale, zero point and integer bounds determine the precision and range of the quantization, and the reconstruction error of a weight yields a bounded discrepancy in the multiplication output:

$$\varepsilon = \widehat{w} - w, \quad \hat{y} - y = x\varepsilon, \quad \left|\hat{y} - y\right| \le |x| \ |\varepsilon| \tag{8}$$

Here x denotes the input activation, and y and $\hat{y}$ denote the outputs computed with w and $\widehat{w}$, respectively. For a fixed weight quantization error, the output error therefore grows linearly with the absolute activation value. For vector and matrix operations, the error propagation can be stated explicitly as:

$$\Delta y = x\Delta W, \quad \|\Delta y\|_2 \le \|x\|_2 \|\Delta W\|_2 \tag{9}$$

Here ΔW denotes the weight-error matrix and Δy the resulting output error, so channels with larger activation norms amplify the associated weight errors and markedly affect the output. AWQ

DOI: 10.54254/2753-8818/2027.TA39198

identified salient weights from activation distributions and applied per-channel scaling to protect them during low-bit quantization, preserving model accuracy while lowering memory and bandwidth requirements [15]. The process shows, from a numerical-analysis perspective, why quantization must be activation-aware.

Knowledge distillation trains a student to approximate the softened output distribution of a teacher, transferring class relationships and other learned information [5].

Alignment and reasoning abilities can in principle be transferred by distillation, but evidence preservation may weaken in multimodal systems, so the design must balance efficiency against representation consistency and sound inference.

7. Research shortcomings and future directions

Three theoretical gaps remain. First, contrastive learning has limited generalization guarantees under noisy image-text pairs, rare concepts and open-world inputs, even where retrieval improves empirically. Second, multimodal hallucination and safety lack an integrated mathematical treatment. Third, cross-modal representations remain difficult to interpret: attention weights reveal associations but do not establish which visual features causally determine an answer. Errors may originate in feature extraction, in connector compression or in the strong prior expectations of a language model.

Distinguishing among these sources is essential both for developing more robust systems and for understanding why a generated response departs from its evidence.

Future work should construct geometric models of aligned multimodal representations, potentially combining multiple distances, semantic hierarchies and optimal transport, and should establish generalization-error bounds for alignment objectives and downstream fine-tuning while paying attention to how loss functions and optimization adjust decision boundaries. Verifiable generation methods should help models distinguish observed evidence from language priors and articulate uncertainty explicitly. Through such directions the field could move from empirical enhancements to more predictable and controllable behaviour without sacrificing efficiency and flexibility across tasks.

8. Conclusion

This review linked the core modules of multimodal models to their mathematical foundations and optimization methods. Encoders mapped heterogeneous signals into vector representations, connectors bridged modality-specific spaces, and language backbones generated outputs conditionally. The square-root factor in scaled dot-product attention limited score variance and helped stabilize the softmax. Low-rank adaptation concentrated fine-tuning in a smaller parameter subspace, and InfoNCE contrastive targets strengthened semantic alignment. FlashAttention improved efficiency by changing the computation and memory-access pattern, while activation-aware quantization addressed numerical discrepancy during low-bit compression. These systems show how mathematical architecture and engineering design jointly influence model behaviour.

More solid and interpretable models in practical applications would be supported by stronger theoretical guarantees and by evidence-consistency methods. Improving multimodal systems further requires clearer mathematical accounts of representation, error propagation and computational efficiency alongside architectural innovation.

DOI: 10.54254/2753-8818/2027.TA39198

References

[1] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS), 30*.

[2] Radford, A., Kim, J. W., Hallacy, C., et al. (2021). Learning transferable visual models from natural language supervision. *Proceedings of the International Conference on Machine Learning (ICML)*, 8748–8763.

[3] Li, J., Li, D., Savarese, S., et al. (2023). BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *Proceedings of the International Conference on Machine Learning (ICML)*, 19730–19742.

[4] Dao, T., Fu, D. Y., Ermon, S., et al. (2022). FlashAttention: Fast and memory-efficient exact attention with IO-awareness. *Advances in Neural Information Processing Systems (NeurIPS), 35*, 16344–16359.

[5] Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv:1503.02531.

[6] Yin, S., Fu, C., Zhao, S., et al. (2024). A survey on multimodal large language models. *National Science Review, 11*, nwae403.

[7] Wu, J., Gan, W., Chen, Z., et al. (2023). Multimodal large language models: A survey. arXiv:2311.13165

[8] van den Oord, A., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv:1807.03748

[9] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

[10] Liu, H., Li, C., Li, Y., et al. (2024). Improved baselines with visual instruction tuning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 26296–26306.

[11] Zhu, D., Chen, J., Shen, X., et al. (2023). MiniGPT-4: Enhancing vision-language understanding with advanced large language models. arXiv:2304.10592

[12] Touvron, H., Martin, L., Stone, K., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv:2307.09288

[13] Hu, E. J., Shen, Y., Wallis, P., et al. (2022). LoRA: Low-rank adaptation of large language models. *Proc Int. Conf Learn Represent. (ICLR)*

[14] Dao, T. (2023). FlashAttention-2: Faster attention with better parallelism and work partitioning. arXiv:2307.08691

[15] Lin, J., Tang, J., Tang, H., et al. (2024). AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. *Proc. Machine Learning and Systems (MLSys), 6*, 87-100