arXiv:2508.13460v1 [cs.CV] 19 Aug 2025

Revisiting MLLM Token Technology through the Lens of Classical Visual Coding

Jinming Liu[1*], Junyan Lin[1*], Yuntao Wei[1*], Kele Shao[2], Keda Tao[2],
Jianguo Huang[1], Xudong Yang[1], Zhibo Chen[3], Huan Wang[2], and Xin Jin[1†]
[1]Eastern Institute of Technology, Ningbo, China [2]Westlake University [3]USTC

***Abstract*—Classical visual coding and Multimodal Large Language Model (MLLM) token technology share the core objective—maximizing information fidelity while minimizing computational cost. Therefore, this paper reexamines MLLM token technology, including tokenization, token compression, and token reasoning, through the established principles of long-developed visual coding area. From this perspective, we (1) establish a unified formulation bridging token technology and visual coding, enabling a systematic, module-by-module comparative analysis; (2) synthesize bidirectional insights, exploring how visual coding principles can enhance MLLM token techniques' efficiency and robustness, and conversely, how token technology paradigms can inform the design of next-generation semantic visual codecs; (3) prospect for promising future research directions and critical unsolved challenges. In summary, this study presents the first comprehensive and structured technology comparison of MLLM token and visual coding, paving the way for more efficient multimodal models and more powerful visual codecs simultaneously.**

***Index Terms*—Multimodal large language models, Visual token technology, Visual coding**

I. Introduction

In the digital media era, image and video coding [1]–[3] underpin nearly every visual experience—from real-time video calls and streaming movies to the billions of photos stored in the cloud—by making raw pixel data orders of magnitude smaller while preserving perceptual quality. The core objective is to represent visual content with the minimum number of bits that still meet application-specific fidelity constraints, thereby reducing bandwidth, storage, and latency costs across heterogeneous networks and devices. Classical codecs (here including traditional [1], [2] and learning-based neural [3]–[5] codecs), typically compress signals by exploiting signal structure through analysis transforms, quantization, and predictive & entropy coding, while optimizing explicit rate–distortion (R–D) trade-offs under device and network constraints.

With the rise of deep learning, we are entering an AI-centric era in which applications increasingly depend on multimodal understanding. MLLMs [6]–[8] extend language models with visual inputs and already support tasks such as captioning, visual question answering, tool use, and embodied reasoning. However, practical deployment is frequently constrained by computational cost: the time and memory complexity of standard attention grow roughly quadratically with the number of tokens [9], [10], so naively encoding rich visual inputs into many tokens may dominate end-to-end latency and energy.

To address this bottleneck, a growing body of work [11], [12] compresses the visual token stream before or within the MLLM. Techniques include more compact tokenization [13], token compression [14], and sparse token reasoning [11]. Although developed in different communities, these methods share deep commonality with visual coding. Both pursue compact representations under source constraints and can be cast in a unified R–D view—where "rate" is a token/compute budget and "distortion" measures semantic fidelity (task performance) rather than purely pixel/perceptual error. Thus, this paper revisits MLLM token technology from the perspective of classical visual coding and makes the following contributions:

- We present a unified problem formulation that links token technology and visual coding, while comparing each module of them by analogy, as shown in Fig. 1, to highlight similarities and differences in their underlying philosophies and operational methodologies.
- We further analyze bidirectional insights from the comparison above and explore how classical coding tools can improve token efficiency and reasoning quality, and how token systems and semantic objectives can guide next-generation semantic codecs.
- Finally, we outline future directions and figure out unresolved problems spanning metrics, algorithms, and system design, for both MLLMs and visual codecs.

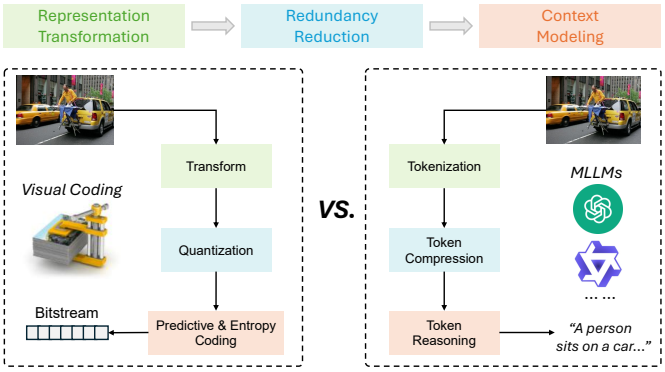


Fig. 1. Module-by-Module comparison of visual coding and MLLMs. This paper revisits them from a unified pipeline of "Representation Transformation → Redundancy Reduction → Context Modeling".

II. Classical Visual Coding and Codecs

Visual coding aims to represent visual data with minimal bits while retaining crucial visual information, enabling faster transmission and lower storage costs. We analyze the codec evolution roadmap as shown in Fig. 2.

A. Traditional Coding Methods

Traditional codecs have developed a series of influential standards. These methods typically included the most impor-

*Equal contribution.
†Corresponding author: Xin Jin (jinxin@eitech.edu.cn).



Fig. 2. Codec evolution roadmap, from traditional codecs [1], [2], [15], [16] towards more general visual coding [3]–[5], [17]–[21].

tant three steps: transformation, quantization, and predictive & entropy coding. For instance, the widely used JPEG [1] standard divides images into 8×8 blocks, applies the Discrete Cosine Transform, and then compresses the coefficients through quantization and entropy coding. PNG [15] uses filtering and deflate algorithms to maintain exact image fidelity. Advances in video coding, such as HEVC [16] and VVC [2], have brought more sophisticated block partitioning, prediction, and transform techniques, improving performance.

B. Learning-based Coding Methods

The resurgence of artificial neural networks has driven rapid progress in learned image compression [17], now surpassing even the traditional standards like VVC [2] in RD performance by utilizing end-to-end optimization. For example, Google [3], [22] proposed hyperprior models, which use side information to estimate better latent code distributions. Some works [4], [5] introduced better transformations, achieving results comparable to VVC [2]. DCVC-series [23] also showed superior performance on video compared to traditional standards, while achieving over 100 FPS for 1080p video on a GPU.

C. Perceptual/Semantic-driven Coding Methods

Recently, image compression is moving beyond pixel-level fidelity to address either perceptual quality for humans or semantic utility for machines [18]. Perceptual codecs aim to optimize images for human perception. These methods relied on GAN-based models [20], [24] or diffusion models [25], [26] to produce visually pleasing results. In parallel, semantic codecs [18], [19], [27] focus on preserving information needed for machine tasks such as classification and detection, allowing better rate-accuracy. Jointly optimizing the compression model and task network [28] has been shown to improve machine task accuracy. Some works [29], [30] enabled scalable bitstreams that can flexibly serve both human and machine needs. Reflecting these trends, industry and standardization bodies have launched initiatives like MPEG Video Coding for Machines (VCM) [21] and JPEG AI [31] to unify perceptual and semantic goals within a single framework.

III. MLLM Token Technology

An MLLM's token working flow has three main stages: the tokenizer first extracts tokens, a post-processing step (e.g., token compression), and the resulting tokens are fed into LLMs for reasoning, completing different downstream tasks.

A. Visual Tokenizers in MLLMs

Visual tokenizers in MLLMs convert image inputs into structured token sequences. They directly influence token efficiency and cross-modal alignment, can be categorized into two classes as follows:

1) Contrastive learning-based visual tokenizers: Contrastive tokenizers dominate mainstream MLLMs by producing semantically rich image representations. CLIP [13] is a foundational model that aligns image and text embeddings through large-scale contrastive pretraining. Building on CLIP, SigLIP [32] replaces the contrastive loss with a sigmoid-based variant and removes the [CLS] token, enabling denser supervision via averaged patch features.

2) Self-Supervised Visual Tokenizers: In contrast, self-supervised tokenizers like DINO [33] require no text supervision. Using a teacher–student framework, DINO promotes view consistency, leading to emergent semantic structure in intermediate tokens. DINOv2 [34] scales this approach with improved alignment and stronger performance on linear classification and segmentation.

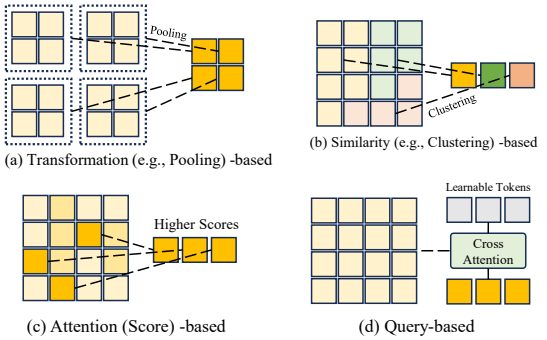


Fig. 3. Comparison of different token compression strategies.

B. Token Compression Strategies in MLLMs

High-resolution images often contain many redundant or uninformative tokens. To address this, researchers have proposed various token compression strategies for image modalities [35], which can be broadly categorized into four types: transformation-based, similarity-based, attention-based, and query-based methods as shown in Fig. 3.

1) Transformation-based Compression: These methods [7], [8], [36], [37] compress image tokens by applying spatial transformations such as pixel unshuffle, pooling, or convolution, effectively reducing token count while preserving local structure. These lightweight operations are typically applied at the model frontend, requiring minimal computation.

Representatively, Qwen2 [36] and InternVL [8] series adopt pixel unshuffle to convert high-resolution images into lower-resolution but higher-channel representations, followed by an MLP for feature dimension alignment. LLaVA-OneVision [7] uses average pooling or bilinear interpolation to learn multi-granular visual representations for adaptive reasoning. Unlike those approaches that implement non-parametric operations, MobileVLM [37] employs depthwise convolution to achieve over 75% token reduction while retaining spatial locality.

2) Similarity-based Compression: These methods [10], [38] reduce the token count by merging redundant or semantically similar visual tokens into representative ones, making them particularly effective for highly redundant image inputs. A foundational work is ToMe [10], which accelerates ViT inference by inserting token clustering modules into each

Transformer layer and fusing tokens based on bidirectional cosine similarity. AuroraCap [38] adopts this strategy and demonstrates its effectiveness in MLLMs by progressively merging tokens after each visual layer.

3) Attention-based Compression: Attention-based strategies [11], [9], [12], [39], [40] leverage the inherent sparsity in attention distributions to dynamically prune less important tokens, thereby improving inference efficiency. These methods can be grouped into two categories based on the pruning stage.

Tokenizer Stage: pruning is guided by attention scores computed within the vision encoder, typically with respect to the [CLS] token. These scores are used to evaluate token importance. For example, PruMerge [9] selects high-attention tokens as cluster centers and aggregates the rest via KNN-based merging. VisPruner [12] retains salient tokens based on attention and further ensures token diversity through iterative semantic deduplication, preserving both relevance and variety.

LLM Stage: pruning is guided by attention scores within the LLM. This enables more semantically aware token selection, crucial for downstream understanding tasks. FastV [11] observes that visual tokens receive minimal attention beyond the early layers and proposes to prune half of the visual tokens immediately following the second layer, achieving a significant speed-up with negligible performance loss. PyramidDrop [39] extends this idea by applying layer-wise adaptive pruning based on attention distributions, striking a better balance between efficiency and accuracy.

4) Query-based Compression: In many multimodal tasks, such as visual question answering, a large portion of visual information may be irrelevant to the task objective. Query-based compression methods [14], [41] aim to reduce this redundancy by using learnable query tokens or external prompts to extract compact visual representations that are most useful for downstream language processing. The BLIP series offers a representative solution. BLIP-2 [41] uses fixed learnable query tokens to distill visual features into a fixed-length representation aligned with the language model, though it does not condition on task instructions.

MiniGPT-4 [14] follows similar designs, where visual features are distilled via learnable queries and projected into compact embeddings, enabling modular and efficient multimodal reasoning.

IV. Bridging Visual Coding and MLLM Tokens

A. The Unified Objective: A Rate-Distortion Perspective

The optimization challenge in both fields is to solve the rate-distortion trade-off, formally expressed as minimizing the cost function $J = R + \lambda D$ [3], [4]. While the formula is shared, its components are defined by the end-user. For **classical visual coding** [3], [4], the goal is to serve the human eye: **Rate** (R) is the *transmission cost* measured in bits, and **Distortion** (D) is the *fidelity error* (e.g., PSNR, MS-SSIM). Conversely, for **MLLM token technology** [7], the goal is to serve the machine model: **Rate** (R) becomes the *computational cost* dictated by the number of tokens, and **Distortion** (D) is the *semantic error* measured in task performance degradation. The unified problem, therefore, is to find an optimal representation by balancing cost and information loss, tailored either for human perception or machine cognition.

B. A Unified Pipeline: A Modular-based Analogy

To solve their respective rate-distortion problems, both systems independently converged on functionally analogous pipelines as shown in Fig. 1 and Table I. We can analyze this pipeline by its core stages.

1) Stage 1: Representation Transformation to a Compact Representation: The initial pipeline stage in both representation transforms the pixels into a compact representation, but the nature of this compactness is fundamentally different: Visual coding pursues **information compactness** [3], [5]. It employs handcrafted transforms, like the DCT, or learned transforms to de-correlate pixels as compact representations, optimized for bit-saving compression algorithms while aiming to preserve fidelity. Conversely, MLLM tokenization aims for **semantic compactness** [13], [32]. It uses learned semantic encoders to extract visual meaning into tokens, resulting in a representation that is semantically dense and optimized for the cognitive efficiency of a downstream reasoning model.

2) Stage 2: Redundancy Reduction: Following stage 1, both pipelines apply a post-processing stage to control representation cost by discarding less important content. In visual coding, quantization is often guided by hand-crafted rules [1] or distribution statistics [42]—either dropping low-salience details outright or assigning them fewer bits. Likewise, MLLMs prune tokens produced by the vision encoder using hand-set rules or importance/similarity measures [9], [10], allocating more tokens to important information and fewer to the rest. Although the mechanisms differ, both are fundamentally rate-control strategies that filter the representation according to a guiding principle: the former optimizes for human-perceived fidelity, the latter for machine-perceived relevance.

3) Stage 3: Context Modeling: Finally, both systems leverage context to enhance efficiency, but their predictive mechanisms serve different ends. Visual coding employs explicit predictive coding in inter/intra-prediction [2]. It uses previously encoded data (spatial or temporal neighbors) to predict the current information unit and then encodes only the prediction residual. Also, some entropy models [17] in visual coding use context models to help distribution estimation for better bits estimation. They are strategies purely aimed at maximizing compression efficiency by exploiting statistical redundancy. In parallel, the MLLM architecture inherently performs prediction through its auto-regressive nature [6]. The model leverages the entire preceding sequence of tokens (both text and visual) as context to predict the next token. This predictive process is not for compressing a residual but for ensuring semantic coherence in its generated output. Thus, while both use context to inform the next step, one predicts to reduce bits, the other predicts to construct meaning.

C. Bidirectional Synergy and Future Co-evolution

1) How Visual Coding Principles Can Refine MLLMs:

TABLE I
A COMPARATIVE OVERVIEW OF VISUAL CODING AND MLLM TOKEN TECHNOLOGY.

Aspect	Classical Visual Coding	MLLM Token Technology
Primary Goal	Efficient data storage and transmission.	Efficient machine computation and reasoning.
Core Optimization (Rate-Distortion)	**Rate:** Transmission cost (bits). **Distortion:** Fidelity error for *human observers* (e.g., PSNR).	**Rate:** Computational cost (token count). **Distortion:** Semantic error for *machine observers* (task performance).
Core Pipeline	**1. Transformation:** Transforms for *energy compaction*. **2. Redundancy Reduction:** Quantization guided by *information distribution*. **3. Context Modeling:** Predictive and entropy coding to *reduce bits*.	**1. Transformation:** Vision encoders for *semantic extraction*. **2. Redundancy Reduction:** Token pruning/merging/VQ guided by *semantic relevance*. **3. Context Modeling:** Auto-regressive process to *construct meaning*.
Bidirectional Insights	**Influences MLLMs by:** Providing principles for token reduction (RDO, VQ); Variable-length tokenization; preprocessing method.	**Influences Codecs by:** Semantic prior for rate control; enabling generative compression; and offering a path to universal compressors.

a) Transform-based Pre-processing: Inspired by classical codecs, MLLM tokenization can be improved by first transforming images into a more compressible domain. This includes using some transforms to operate in the frequency domain [43], [44], or adopting principles from video coding to separately tokenize "keyframe" information and more redundant "motion-like" information, leading to a more efficient initial representation [42].

b) Principled Redundancy Reduction: Some compression algorithms provide direct inspiration for token reduction. The concept of Run-Length Coding, for instance, can be adapted to merge sequences of similar tokens [45]. More broadly, "information-preserving guided selection" strategies, which prune or merge tokens based on their contribution to overall information, are a direct application of rate-control philosophies to the semantic domain [46].

c) Theoretical Foundations: Drawing from the ultimate theoretical principle of compression—Kolmogorov Complexity and its computable proxy, the Minimum Description Length (MDL)—inspires a shift towards variable-length tokenization [47]. This allows models to represent simple images with fewer tokens and complex ones with more, achieving a principled trade-off between token count and semantic fidelity, akin to a quality parameter in visual coding.

d) Vector Quantization: Vector Quantization (VQ), a cornerstone of signal processing, has been successfully imported into MLLMs [48]. By mapping continuous embeddings to a discrete, learned codebook, VQ not only reduces the representation cost but also significantly improves the quality and stability of generative tasks.

2) How MLLM Intelligence Can Refresh Codecs:

a) Semantic Prior: An MLLM can act as a high-level "brain" for a codec [19], [49]. By first providing a semantic analysis of the scene, it can generate a priority map that guides the codec to dynamically allocate more bits to critical regions (e.g., faces, text) while aggressively compressing unimportant backgrounds. This optimizes for downstream task performance rather than uniform perceptual quality.

b) Token Coding: A paradigm shift is emerging where the target of compression is no longer pixels, but the semantic tokens themselves. Compressing tokens [50], [51] directly yields superior performance for machine-vision tasks.

c) Generative Compression: An MLLM can distill an image into a compact text description, which is then transmitted and used by a generative model to reconstruct the image, enabling ultra-low bitrates.

d) Universal Predictive Compressor: At a fundamental level, the powerful next-token prediction capability of large models makes them extraordinarily effective general-purpose compressors. Large models have already demonstrated the ability to compress general data streams [52], [53], surpassing specialized codecs like PNG. This positions MLLMs not merely as a tool for compression, but as the potential core engine for future universal data compression standards.

3) System Perspective: The convergence of MLLM technology and visual coding naturally leads to a practical system architecture: an edge–cloud co-design. The core challenge in deploying large-scale MLLMs for real-world tasks is the bottleneck between the data source (edge) and the processing center (cloud), driven by two constraints: limited network bandwidth and the high computational cost of cloud-based MLLMs. Several studies [54], [55] have explored MLLM-oriented visual coding that reduces transmission bandwidth while preserving MLLM performance. Some works place the tokenizer at the edge [51], encode and transmit tokens directly without image reconstruction at the decoder. Nonetheless, jointly optimizing the bitrate–complexity–performance trade-off remains an open problem.

V. FUTURE DIRECTIONS AND OPEN PROBLEMS

The convergence of visual coding and MLLM token technology opens a rich frontier for research. On the codec side, MLLM tokens can serve as semantic priors to enable object-/region-level adaptive bit allocation, language-/task-driven temporal rate control, scalable semantic layering (transmitting essentials first, details later), generative completion with ultra-low bitrates, and edge-side lightweight semantic summaries. Conversely, importing coding principles into token systems suggests variable-length and hierarchical semantic tokenization (base summaries with enhancement details), better video-stream representations, spatiotemporal prediction with redundancy suppression, and information-theoretic foundations for token semantic entropy. At the system level, edge–cloud co-design for semantic rate control and efficient MLLM inference, together with hardware–software co-design of accelerators under energy/latency constraints, constitute pressing directions for more general, scalable, reliable semantic communication.

REFERENCES

[1] G. K. Wallace, "The jpeg still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[2] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the versatile video coding (vvc) standard and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 3736–3764, 2021.

[3] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," *preprint arXiv:1802.01436*, 2018.

[4] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7939–7948.

[5] J. Liu, H. Sun, and J. Katto, "Learned image compression with mixed transformer-cnn architectures," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 14 388–14 397.

[6] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang *et al.*, "Qwen2. 5-vl technical report," *preprint arXiv:2502.13923*, 2025.

[7] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, and C. Li, "Llava-onevision: Easy visual task transfer," 2025.

[8] Z. Chen, W. Wang, H. Tian, S. Ye, Z. Gao, E. Cui, W. Tong, K. Hu, J. Luo, Z. Ma *et al.*, "How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites," *Sci. China Inf. Sci.*, vol. 67, no. 12, p. 220101, 2024.

[9] Y. Shang, M. Cai, B. Xu, Y. J. Lee, and Y. Yan, "Llava-prumerge: Adaptive token reduction for efficient large multimodal models," 2025.

[10] D. Bolya, C.-Y. Fu, X. Dai, P. Zhang, C. Feichtenhofer, and J. Hoffman, "Token merging: Your vit but faster," 2023.

[11] 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," 2024.

[12] Q. Zhang, A. Cheng, M. Lu, R. Zhang, Z. Zhuo, J. Cao, S. Guo, Q. She, and S. Zhang, "Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms," 2025.

[13] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[14] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," 2024.

[15] P. N. Graphics, "Specification information technology-computer graphics and image processing-portable network graphics (png): Functional specification," *ISO/IEC*, vol. 15948, 2003.

[16] B. Bross, "High efficiency video coding (hevc) text specification draft 10 (for fdis & last call)," in *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 12th Meeting, Geneva,(Jan. 2013)*, 2013.

[17] D. He, Z. Yang, W. Peng, R. Ma, H. Qin, and Y. Wang, "Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5718–5727.

[18] Y.-H. Chen, Y.-C. Weng, C.-H. Kao, C. Chien, W.-C. Chiu, and W.-H. Peng, "Transtic: Transferring transformer-based image compression from human perception to machine perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 23 297–23 307.

[19] C. Li, G. Lu, D. Feng, H. Wu, Z. Zhang, X. Liu, G. Zhai, W. Lin, and W. Zhang, "Misc: Ultra-low bitrate image semantic compression driven by large multimodal model," *IEEE Transactions on Image Processing*, 2024.

[20] F. Mentzer, G. D. Toderici, M. Tschannen, and E. Agustsson, "High-fidelity generative image compression," *Advances in neural information processing systems*, vol. 33, pp. 11 913–11 924, 2020.

[21] Y. Zhang, C. Rosewarne, S. Liu, and C. Hollmann, "Call for evidence for video coding for machines," *ISO/IEC JTC 1/SC 29/WG*, vol. 2, 2022.

[22] D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," *Advances in neural information processing systems*, vol. 31, 2018.

[23] Z. Jia, B. Li, J. Li, W. Xie, L. Qi, H. Li, and Y. Lu, "Towards practical real-time neural video compression," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 12 543–12 552.

[24] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. V. Gool, "Generative adversarial networks for extreme learned image compression," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 221–231.

[25] R. Mao, X. Feng, C. Gao, L. Li, D. Liu, and X. Sun, "Perceptual image compression with conditional diffusion transformers," in *2024 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2024, pp. 1–5.

[26] T. Xu, J. Li, B. Li, Y. Wang, Y.-Q. Zhang, and Y. Lu, "Picd: Versatile perceptual image compression with diffusion rendering," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 28 436–28 445.

[27] J. Liu, H. Sun, and J. Katto, "Semantic segmentation in learned compressed domain," in *2022 Picture Coding Symposium (PCS)*. IEEE, 2022, pp. 181–185.

[28] H. Li and X. Zhang, "Human–machine collaborative image compression method based on implicit neural representations," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 14, no. 2, pp. 198–208, 2024.

[29] J. Liu, R. Feng, Y. Qi, Q. Chen, Z. Chen, W. Zeng, and X. Jin, "Rate-distortion-cognition controllable versatile neural image compression," in *European Conference on Computer Vision*. Springer, 2024, pp. 329–348.

[30] X. Jin, R. Feng, S. Sun, R. Feng, T. He, and Z. Chen, "Semantical video coding: Instill static-dynamic clues into structured bitstream for ai tasks," *Journal of Visual Communication and Image Representation*, vol. 93, p. 103816, 2023.

[31] J. Ascenso and E. Upenik, "White paper on jpeg ai scope and framework," *ISO/IEC JTC*, vol. 1, 2021.

[32] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.

[33] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.

[34] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *Transactions on Machine Learning Research Journal*, 2024.

[35] K. Shao, K. Tao, K. Zhang, S. Feng, M. Cai, Y. Shang, H. You, C. Qin, Y. Sui, and H. Wang, "When tokens talk too much: A survey of multimodal long-context token compression across images, videos, and audios," *preprint arXiv:2507.20198*, 2025.

[36] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge *et al.*, "Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution," *preprint arXiv:2409.12191*, 2024.

[37] X. Chu, L. Qiao, X. Lin, S. Xu, Y. Yang, Y. Hu, F. Wei, X. Zhang, B. Zhang, X. Wei *et al.*, "Mobilevlm: A fast, strong and open vision language assistant for mobile devices," *preprint arXiv:2312.16886*, 2023.

[38] W. Chai, E. Song, Y. Du, C. Meng, V. Madhavan, O. Bar-Tal, J.-N. Hwang, S. Xie, and C. D. Manning, "Auroracap: Efficient, performant video detailed captioning and a new benchmark," 2025.

[39] L. Xing, Q. Huang, X. Dong, J. Lu, P. Zhang, Y. Zang, Y. Cao, C. He, J. Wang, F. Wu *et al.*, "Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction," 2025.

[40] K. Shao, K. Tao, C. Qin, H. You, Y. Sui, and H. Wang, "Holitom: Holistic token merging for fast video large language models," *preprint arXiv:2505.21334*, 2025.

[41] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.

[42] U. Gadot, A. Shocher, S. Mannor, G. Chechik, and A. Hallak, "Rl-rc-dot: A block-level rl agent for task-aware video compression," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 12 533–12 542.

[43] H. Feng, Q. Liu, H. Liu, J. Tang, W. Zhou, H. Li, and C. Huang, "Docpedia: Unleashing the power of large multimodal model in the frequency domain for versatile document understanding," *Science China Information Sciences*, vol. 67, no. 12, p. 220106, 2024.

[44] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine, "Fast: Efficient action tokenization for vision-language-action models," *preprint arXiv:2501.09747*, 2025.

[45] R. Choudhury, G. Zhu, S. Liu, K. Niinuma, K. Kitani, and L. Jeni, "Don't look twice: Faster video transformers with run-length tokenization," *Advances in Neural Information Processing Systems*, vol. 37, pp. 28 127–28 149, 2024.

[46] X. Tan, P. Ye, C. Tu, J. Cao, Y. Yang, L. Zhang, D. Zhou, and T. Chen, "Tokencarve: Information-preserving visual token compression in multimodal large language models," *preprint arXiv:2503.10501*, 2025.

[47] S. Duggal, S. Byun, W. T. Freeman, A. Torralba, and P. Isola, "Single-pass adaptive image tokenization for minimum program search," *preprint arXiv:2507.07995*, 2025.

[48] Z. Geng, Y. Wang, Y. Ma, C. Li, Y. Rao, S. Gu, Z. Zhong, Q. Lu, H. Hu, X. Zhang *et al.*, "X-omni: Reinforcement learning makes discrete autoregressive image generative models great again," *arXiv preprint arXiv:2507.22058*, 2025.

[49] J. Liu, Y. Wei, J. Lin, S. Zhao, H. Sun, Z. Chen, W. Zeng, and X. Jin, "Tell codec what worth compressing: Semantically disentangled image coding for machine with lmms," in *2024 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2024, pp. 1–5.

[50] C. Gao, Y. Ma, Q. Chen, Y. Xu, D. Liu, and W. Lin, "Feature coding in the era of large models: Dataset, test conditions, and benchmark," *preprint arXiv:2412.04307*, 2024.

[51] L. Qiao, M. B. Mashhadi, Z. Gao, R. Tafazolli, M. Bennis, and D. T. Niyato, "Token communications: A large model-driven framework for cross-modal context-aware semantic communications," *IEEE Wireless Communications Magazine*, 2025.

[52] Z. Li, C. Huang, X. Wang, H. Hu, C. Wyeth, D. Bu, Q. Yu, W. Gao, X. Liu, and M. Li, "Lossless data compression by large models," *Nature Machine Intelligence*, pp. 1–6, 2025.

[53] G. Deletang, A. Ruoss, P.-A. Duquenne, E. Catt, T. Genewein, C. Mattern, J. Grau-Moya, L. K. Wenliang, M. Aitchison, L. Orseau *et al.*, "Language modeling is compression," in *The Twelfth International Conference on Learning Representations*.

[54] C.-H. Kao, C. Chien, Y.-J. Tseng, Y.-H. Chen, A. Gnutti, S.-Y. Lo, W.-H. Peng, and R. Leonardi, "Bridging compressed image latents and multimodal large language models," in *The Thirteenth International Conference on Learning Representations*.

[55] B. Li, S. Wang, S. Wang, and Y. Ye, "High efficiency image compression for large visual-language models," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.