Computers and Artificial Intelligence; Vol.3, No.6; 2026
E-ISSN: 3006-3728 P-ISSN: 3006-371X
DOI: https://doi.org/10.70267/cai.v3n6.186196
Published by: Zeus Press

From Patches to Codebooks: A Discrete-Continuous Spectrum of Transformer Tokenization Across Modalities

Xinyuan Du*

International School, Beijing University of Posts and Telecommunications, Beijing 102206 China

**Corresponding author: Xinyuan Du.*

Abstract

Since its introduction, the Transformer has expanded from text processing to multimodal fields such as images and speech. During this process, a recurring question is how to convert continuous and heterogeneous raw signals into token representations that a model can process. This tokenization process is often accompanied by varying degrees of information selection or bottlenecks, and has consequently developed into two basic routes: discrete quantization (codebook/vector quantization [VQ]) and continuous representations (patches/latent features). Several existing representative reviews mainly focus on discrete tokenization, while unified comparisons of the continuous and discrete routes remain relatively limited. Taking the discrete-continuous spectrum as its main thread, this paper reviews representative work on continuous serialization, discrete codebook quantization, discrete units in speech, latent bottlenecks, and continuous latent variables. It analyzes the respective trade-offs of discrete and continuous approaches in terms of task suitability, information efficiency, vocabulary compatibility, and other dimensions. On this basis, this paper summarizes several empirical design principles concerning when to use discrete representations and when to use continuous representations, and attempts to provide a unified analytical framework for representation design in multimodal foundation models.

Keywords

transformer, tokenization, discrete representation, continuous representation, vector quantization, multimodality

1. Introduction

1.1 From Text to All Modalities: Cross-Modal Transfer of the Transformer

The Transformer [1], through its self-attention mechanism, profoundly changed the way sequences are modeled. Although it was originally designed for machine translation, it quickly moved beyond the boundaries of natural language processing. In vision, the Vision Transformer (ViT) [2] first showed that a pure Transformer could achieve performance comparable to or even better than that of convolutional networks in image classification. In speech, Wav2Vec 2.0 [3] and Hidden-Unit Bidirectional Encoder Representations from Transformers (HuBERT) [4] extended the Transformer to self-supervised speech representation learning. In multimodal research, Contrastive Language-Image Pre-training (CLIP) [5] and DALL-E [6] achieved

© 2026 The Author(s). Authors retain copyright in their articles.

image-text alignment and text-to-image generation, respectively. The Transformer is therefore widely regarded as a general-purpose foundational architecture for contemporary artificial intelligence.

1.2 Tokenization: An Overlooked Common Mechanism in Cross-Modal Transfer

However, the input to a Transformer has an implicit assumption: it must be a token sequence. Text naturally satisfies this assumption because words or subwords can be directly indexed. Images, however, are pixel grids, and speech consists of continuous waveforms; neither has ready-made tokens. Therefore, whenever the Transformer enters a new modality, researchers must answer the same question: how can continuous and heterogeneous raw signals be converted into token sequences? It should be clarified that the term "tokenization across modalities" in this paper does not refer to designing a unified cross-modal tokenizer. Instead, it refers to the tokenization methods separately adopted for different non-text modalities, such as images and speech, and examines their discrete-continuous differences within a unified framework.

1.3 The Tension Between Discrete and Continuous Representations: An Ongoing Debate

During tokenization, a fundamental divergence has gradually emerged: should continuous signals be discretized into indices in a finite vocabulary, as in text (codebook quantization), or should they be directly retained as continuous feature vectors? Early work showed the success of ViT with continuous patches, whereas the vector-quantized variational autoencoder (VQ-VAE) [7] and DALL-E relied on discrete tokens to support generation. In recent years, this divergence has not converged. Instead, it has become sharper with the rise of large-scale multimodal models. On the one hand, discrete tokens have been widely adopted because they are structurally isomorphic to the vocabulary of language models. On the other hand, methods such as autoregression over continuous latent variables have shown that generation does not necessarily require discretization. This tension between discrete and continuous representations is a key clue for understanding multimodal representation design.

1.4 Scope and Contributions of This Paper

This paper adopts a selective review method and uses the discrete-continuous spectrum as its main thread to review representative work on tokenization across modalities. Compared with several existing reviews that mainly focus on discrete tokenization, this paper attempts to place the continuous and discrete sides within the same comparative framework. This paper makes three contributions. First, it constructs an organizational framework based on the discrete-continuous spectrum and distinguishes the relevant mechanisms at three levels: input tokenization, latent-variable tokenization, and discretization of training targets. Second, it compares discrete and continuous approaches at four levels: task suitability, information efficiency, vocabulary compatibility, and training characteristics. Third, it explains the choices between the two types of methods from the perspective of information selection/bottlenecks and summarizes several empirical design principles concerning when to use discrete representations and when to use continuous representations. The scope of this paper is limited to vision (images), speech, and typical multimodal models (image-text), without discussing other modalities such as video.

1.5 Comparison With Existing Reviews

To explain the necessity of this paper, Table 1 compares this paper with several representative reviews in terms of the time span, modalities covered, whether discrete and continuous approaches receive equal attention, and whether unified design principles and theoretical perspectives are provided.

Table 1: Comparison with existing reviews

Review	Time span	Modalities covered	Discrete and continuous	Design principles/theoretical perspective
[8] Review of discrete tokenization	Through 2025	Multimodal	Mainly discrete	Partial
[9] Review of discrete tokenizers	Through 2025	Multimodal	Mainly discrete	Partial
[10] Review of discrete audio tokens	Through 2025	Audio only	Mainly discrete	No

© 2026 The Author(s). Authors retain copyright in their articles.

Review	Time span	Modalities covered	Discrete and continuous	Design principles/theoretical perspective
[11] Review of multimodal Transformers	Through 2023	Multimodal	Brief coverage only	No
This paper	2017-2026	Images, speech, and multimodality	Discrete + continuous	Yes

2. Review Method

2.1 Search Scope and Search Terms

This paper is a selective (narrative) review. It does not seek to exhaust all variants; instead, it organizes the literature around approaches that established a tokenization paradigm and were widely adopted by subsequent work. The main sources were Google Scholar, DBLP, and arXiv. Search terms included "transformer tokenization," "vector quantization image," "discrete visual token," "continuous latent autoregressive," "speech quantization," and "multimodal tokenizer." The initial search cutoff was December 31, 2025. As a selective review, this paper uses whether a work established a widely adopted paradigm as the selection criterion, rather than tracking the latest state of the art (SOTA). To maintain timeliness, a supplementary search was conducted in 2026 during the final writing stage, adding recent work such as MANZANO [12] (International Conference on Learning Representations [ICLR] 2026) and TokenBridge [13] (International Conference on Computer Vision [ICCV] 2025). These works further support rather than change this paper's judgment regarding the discrete-continuous spectrum.

2.2 Inclusion and Exclusion Criteria

The inclusion criteria were as follows: (1) representative work that first introduced a type of tokenization strategy into a modality and had broad influence; and (2) milestone papers that occupy a clear position on the discrete-continuous spectrum. The exclusion criteria were the many variants that only made engineering improvements to an existing tokenization mechanism without changing the token form, as well as purely application-oriented studies unrelated to the topic of this paper.

2.3 Basis for Identifying Milestone and Representative Work

The main criterion for identifying milestone or representative work was whether it established a tokenization paradigm and was widely adopted as a foundation by subsequent work. Google Scholar citation counts were used as supporting evidence. It should be noted that citation counts were used only to help confirm historical influence, not as the core screening criterion, so as to avoid selection bias toward earlier highly cited work and against emerging directions. After screening according to these criteria, this paper included 25 references: 19 core works, including the original Transformer architecture and representative methods for each modality; four comparative reviews; and two extended methods, finite scalar quantization (FSQ) and VQ-VAE-2.

3. Background and Perspective

3.1 The Input Assumption of the Transformer and the Need for Serialization

The self-attention mechanism of the Transformer naturally operates on sequences: each position interacts pairwise with all other positions. This requires the input to be organized first as a sequence of vectors. For text, word embeddings directly provide such a sequence. For continuous modalities such as images and speech, the signal must first be serialized, that is, divided into units and then mapped unit by unit to vectors. This step determines what information the downstream model sees and is therefore the basis for understanding all subsequent designs.

3.2 Information Selection and Bottlenecks: Tokenization and Information Compression

Tokenization is often accompanied by information selection to varying degrees. Discrete quantization creates an explicit bottleneck through a finite vocabulary. If the codebook size is K, the upper bound on the

© 2026 The Author(s). Authors retain copyright in their articles.

number of bits used to encode a single discrete index is approximately $\log_2 K$ bits. However, the total bitrate or compression efficiency of the system also depends on the number of tokens, the number of codebook levels, such as in residual vector quantization (RVQ), and the index distribution; it cannot be summarized by log K alone. In contrast, continuous patch projection is not necessarily compression. It may only rearrange or linearly project the signal and may even increase the representational dimensionality. Therefore, the information-bottleneck perspective mainly applies to mechanisms with explicit constraints, such as discrete quantization. For continuous representations, this perspective should be used more cautiously, with greater attention paid to their information-preserving properties.

3.3 Definition of Terms

Discrete token: an integer index selected from a finite vocabulary (codebook), usually mapped back to a vector through an embedding table.

Continuous representation: a token represented directly as a real-valued vector without discretization, such as a ViT patch embedding or a Perceiver [14] latent vector.

Latent bottleneck: a fixed-size set of vectors used as an intermediate representation that reads the input through cross-attention without explicitly assigning a discrete index to every input unit.

Vector quantization (VQ): the operation of mapping a continuous vector to the nearest codeword in a codebook; it is the core means of discretization.

4. The Discrete-Continuous Spectrum: An Organizational Framework

4.1 Two Dimensions of the Spectrum

This paper organizes the strategies along two dimensions. The first is discrete-continuous, which describes whether tokenization includes quantization. The second is discriminative-generative, which describes the nature of the downstream task. These two dimensions are not strictly bound together but instead reflect a common tendency. Discriminative tasks only need to compress the input into a representation that supports discrimination, for which continuous representations are usually sufficient. Generative tasks require sampling the next unit one by one, and a finite discrete vocabulary makes it easier to define the sampling distribution in engineering practice. However, this correspondence is not absolute. For example, the masked autoregressive (MAR) model [15] performs generation with continuous latent variables, whereas Bidirectional Encoder representation from Image Transformers (BEiT) [16] uses discrete tokens for understanding-oriented self-supervised learning; both are counterexamples.

4.2 Overview of Four Routes of Discussion

To avoid conflating tokenization mechanisms with training objectives or representation-learning methods, and because the different works do not lie at the same mechanism level, this paper does not claim to provide a strictly mutually exclusive classification of tokenizers. Instead, it organizes the discussion into four routes: (1) continuous serialization, represented by ViT patch projection; (2) discrete codebook quantization, represented by VQ-VAE, the discrete variational autoencoder (dVAE), and the Transformer-based one-dimensional tokenizer (TiTok) [17]; (3) discrete units/quantization targets in speech, treated as a cross-mechanism modality case and represented by Wav2Vec 2.0, HuBERT, and EnCodec [18]; and (4) latent bottlenecks and continuous latent variables, represented by Perceiver and MAR, where Perceiver is closer to a latent-bottleneck architecture than to a tokenizer in the strict sense. The masked autoencoder (MAE) [19], CLIP, BEiT, DALL-E, and Chameleon [20] are regarded as representative training paradigms or applications built on the routes above and are discussed under their corresponding routes.

4.3 Correspondence Between Three Mechanism Levels and Representative Work

To avoid conceptual misalignment among tokenizers, latent representations, and training targets in horizontal comparisons, this paper distinguishes the relevant mechanisms at three levels and assigns the representative works to their corresponding levels (Table 2).

© 2026 The Author(s). Authors retain copyright in their articles.

Table 2: Correspondence between three mechanism levels and representative work

Mechanism level	Meaning	Representative work
Input tokenization (serialization)	Directly divides a raw signal and maps it to a sequential representation	ViT (continuous patch serialization)
Latent-variable tokenization/bottleneck	First obtains latent variables through an encoder and then quantizes or compresses them	VQ-VAE/dVAE, EnCodec, and TiTok (codebook quantization); Perceiver (continuous latent bottleneck); MAR (continuous latent variables)
Discretization of training targets	Constructs discrete (or continuous) prediction targets for self-supervised training	Wav2Vec 2.0 (quantized targets); HuBERT (cluster-based pseudo-labels); BEiT (discrete masked prediction); MAE (continuous masked reconstruction)

Note: The same method may involve multiple mechanism levels. The table indicates its primary level. For example, quantization in VQ-VAE and EnCodec acts on the encoder output (latent variables), rather than on the raw input, and is therefore classified as latent-variable tokenization.

5. Continuous Serialization Route

5.1 ViT: Continuous Patch Serialization (A Representative Continuous Tokenizer)

ViT [2] divides an image into fixed-size patches, flattens each patch, maps it through a linear projection to a D-dimensional vector, and then adds a [CLS] token for classification and positional encoding to form a continuous token sequence. The entire process contains no discretization. The central finding of ViT is that, with a sufficiently large amount of data, a pure Transformer can achieve performance comparable to or even better than that of convolutional networks without the inductive biases specific to convolutional networks, such as translation invariance and locality. This makes continuous serialization a representative paradigm for discriminative tasks, in which continuous representations are usually sufficient to retain the required information.

5.2 MAE: A Masked Reconstruction Paradigm on Continuous Tokens

MAE [19] does not introduce a new tokenizer. Instead, it uses ViT-style continuous patch serialization and proposes a masked-autoencoding pretraining paradigm on this basis. It randomly masks most patches, feeds only visible patches into the encoder, and uses a lightweight decoder to reconstruct the pixels of the masked patches. The reconstruction target is continuous pixels rather than discrete tokens. Therefore, MAE can be regarded as a representative training paradigm under the continuous-token route, rather than a new tokenizer.

5.3 CLIP: A Cross-Modal Alignment Objective on Continuous Representations

CLIP [5] likewise does not propose a new tokenizer. It uses an image encoder (ViT or a residual network [ResNet]) and a text encoder to extract continuous representations separately, and aligns them in the same space through contrastive learning. It represents a cross-modal alignment objective under the continuous-token route. It shows that, when the task is alignment rather than generation, continuous representations are more natural because alignment depends on geometric relationships between representations rather than on discrete symbols.

Taken together, the common logic of the continuous route is that, for discriminative, alignment, and reconstruction tasks that do not require discrete sampling, direct representation with real-valued vectors is often sufficient, and the information bottleneck introduced by discretization is not necessary. In addition, self-supervised visual Transformer methods such as self-distillation with no labels (DINO) [21] further show that continuous patch representations are also effective in unsupervised understanding tasks.

6. Discrete Codebook Route

6.1 VQ-VAE and dVAE: The Origin of Discretization

VQ-VAE [7] first introduced vector quantization into an image autoencoder. The continuous features output by the encoder are mapped to discrete codes in a codebook through nearest-neighbor lookup, and the decoder

© 2026 The Author(s). Authors retain copyright in their articles.

then reconstructs the input from the discrete codes. Its direct objectives are reconstruction and compression, and its central contribution is obtaining a discrete latent representation. The generative ability does not come from VQ-VAE itself but from an autoregressive prior subsequently trained on the discrete codes, such as VQ-VAE-2 [22]. Building on this idea, dVAE was used by DALL-E as an image tokenizer, further establishing the role of discrete tokens in generative tasks.

6.2 BEiT: A Masked Prediction Paradigm on Discrete Tokens

BEiT [16] does not design a new image tokenizer. Instead, it uses the discrete visual tokens of dVAE and introduces BERT-style masked prediction on top of them: it randomly masks some tokens and trains the model to predict the masked discrete tokens. This design shows that discrete tokens can serve not only generation but also understanding-oriented self-supervised tasks; the key point is that discretization provides an explicit and predictable target.

6.3 DALL-E: Autoregressive Generation on Discrete Tokens

DALL-E [6] unifies text and images as discrete token sequences. Text is encoded through byte-pair encoding (BPE), images are encoded through dVAE, and the two are concatenated and fed to an autoregressive Transformer for next-token prediction. Discrete tokens play an important role here: they allow images and text to share the same "language," enabling a unified Transformer to perform autoregressive generation over any sequence.

6.4 TiTok: A Compact Discrete Tokenizer

TiTok [17] proposes an extremely compact discrete tokenizer that can represent an entire image with a very small number of latent tokens, such as 32, and achieves competitive reconstruction and generation results under the settings reported in the original paper. TiTok challenges the simple intuition that more tokens necessarily provide more complete information, showing that the quality and structure of discrete tokens are more important than their quantity.

6.5 Chameleon: Early Fusion With Unified Discrete Tokens

Chameleon [20] converts images into discrete tokens through a discrete tokenizer and combines them with text tokens in the same sequence for autoregressive modeling, thereby enabling any-to-any generation. Its central feature is unified discrete tokens rather than continuous latent variables, so this paper places it under the discrete route. It highlights the engineering and scalability advantages arising from the structural isomorphism between discrete tokens and the vocabulary of language models.

The common logic of the discrete route is to compress continuous signals into enumerable indices through a finite codebook, thereby obtaining representations that are structurally isomorphic to a text vocabulary and can be sampled token by token. The costs are additional training difficulties, such as quantization error and codebook collapse.

7. Discrete Units and Quantization Targets in Speech

Speech is a continuous waveform signal and has no natural discrete units similar to words. At the same time, speech research also contains many continuous representation methods, such as wav2vec [23], which directly uses continuous features for contrastive predictive learning; discretization is not required. This section focuses on a specific choice: when a task requires an explicit discrete prediction target, such as self-supervised contrastive learning or codec compression, how do researchers construct discrete units?

7.1 Wav2Vec 2.0: Quantized Targets and Contrastive Learning

Wav2Vec 2.0 [3] first uses a convolutional neural network (CNN) to extract continuous features from raw waveforms, then maps some features to discrete codes through a quantization module (Gumbel softmax), and trains the model using contrastive learning and masked prediction. It should be noted that the quantization module is mainly used to provide discrete targets for contrastive learning, whereas the contextual representations output by the Transformer remain continuous.

© 2026 The Author(s). Authors retain copyright in their articles.

7.2 HuBERT: Cluster-Based Pseudo-Labels

HuBERT [4] adopts another discretization approach. It first applies k-means clustering to speech features, uses the cluster assignments as pseudo-labels, and then trains the model to predict the pseudo-labels at masked positions. Unlike the end-to-end quantization of Wav2Vec 2.0, the discrete targets in HuBERT come from offline clustering, and the clustering results mainly serve as pseudo-labels for self-supervised training rather than as a complete discretization of speech.

7.3 EnCodec: Residual Quantization for Coding and Decoding

From the perspective of compression/transmission, EnCodec [18] uses RVQ to encode audio into multilevel discrete codes, which are then decoded for high-fidelity reconstruction. Through multiple rounds of quantization, RVQ progressively approximates the residual and achieves a balance between compression rate and fidelity, becoming a commonly used discrete solution for audio coding and decoding.

The speech examples show that discretization here is more often a tool serving self-supervised objectives or coding and decoding than a comprehensive discretization of the signal. The contextual representations after quantization are often still continuous.

8. Latent Bottlenecks and Continuous Latent Variables

8.1 Perceiver: A Continuous Latent Bottleneck Set

Perceiver [14] does not assign a discrete index to each input unit. Instead, it maintains a fixed-size set of continuous latent vectors and reads inputs of arbitrary size through asymmetric cross-attention. This continuous latent bottleneck avoids the quadratic complexity of self-attention while maintaining generality across arbitrary modalities, and constitutes a special form at the continuous end of the spectrum.

8.2 Continuous Latent Autoregression (MAR): A Return to the Continuous Side

MAR [15] challenges the preconception that generation must use discretization. It performs token-by-token autoregression in a continuous latent space and uses a diffusion loss for each token. Under the experimental settings and benchmarks reported in the original paper, MAR shows that autoregression over continuous latent variables can achieve competitive generative performance, bringing the debate between discrete and continuous representations back to the foreground.

Latent bottlenecks and continuous latent variables show that discrete and continuous representations are not a binary choice. There are continuous bottlenecks without explicit discretization and continuous autoregressive generation that bypasses vector quantization; together, they form the middle region of the spectrum. This middle region has been supported more directly by recent work. MANZANO [12] uses the same shared visual encoder to output continuous features for understanding and discrete tokens for generation, while TokenBridge [13] bridges the continuous and discrete routes in autoregressive generation by applying post-training quantization to pretrained continuous features.

9. Comprehensive Comparison and Design Principles

9.1 Reorganization of the Comparison Dimensions

The differences between discrete and continuous approaches appear at different levels. Task type, efficiency indicators, and compatibility should not be placed in one homogeneous set of dimensions. This paper groups them into four levels: task suitability (discrimination/generation), information efficiency (compression/fidelity), vocabulary compatibility (alignment/information preservation), and training characteristics (stability). Task suitability depends on the downstream task; information efficiency concerns the rate-distortion properties of the representation; vocabulary compatibility concerns cross-modal composability; and training characteristics concern optimization difficulty and include two opposing aspects. During tokenizer training, discrete VQ often requires a straight-through estimator (STE) and is susceptible to codebook collapse, whereas continuous approaches permit direct backpropagation. During downstream

© 2026 The Author(s). Authors retain copyright in their articles.

autoregressive modeling, discrete representations can use cross-entropy classification, whereas continuous representations often require more complex distribution modeling, such as the diffusion loss used in MAR [15]. To avoid level misalignment, the comparison in the table mainly concerns input tokenization and latent-variable tokenization, corresponding to the three-level distinction in Section 4.3; discretization of training targets is discussed separately. Table 3 provides an overview.

Table 3: Comparison of discrete and continuous tokens at different levels

Comparison level	Discrete tokens	Continuous tokens
Task suitability	Commonly used for autoregressive generation	Commonly used for discrimination/alignment
Information efficiency	A single index uses approximately log2 K bits; the total bitrate depends on the number of tokens, the number of codebook levels, and the distribution, and quantization error is present	No discrete quantization error; whether compression occurs depends on the serialization/projection/bottleneck design
Vocabulary compatibility	Structurally isomorphic to a text vocabulary and easy to model jointly	Preserves geometric information; alignment requires separate handling
Training characteristics	Nearest-neighbor selection is non-differentiable and requires an STE; susceptible to codebook collapse	Usually permits direct backpropagation and has no codebook-collapse problem

Note: The table summarizes empirical tendencies rather than universal conclusions. The "training characteristics" row refers only to the difficulty of training the tokenizer itself; the difficulty of downstream autoregressive modeling is discussed in Section 9.1. TokenBridge [13] attempts to combine the strengths of discrete and continuous approaches by first training continuous features and then training quantization.

9.2 When to Use Discrete Representations and When to Use Continuous Representations: Design Principles

Based on the comparison above, several empirical design principles, rather than universal boundary conditions, can be summarized. When a task centers on autoregressive generation and requires sharing a vocabulary with a language model, discrete tokens tend to be used. When a task centers on discrimination, alignment, or reconstruction fidelity and does not require discrete sampling, continuous representations are commonly adopted and are more natural for such tasks. When both compression and fidelity, as in audio coding and decoding, or general-purpose perception, as in Perceiver, must be considered, intermediate forms such as latent bottlenecks may be more appropriate candidate routes. It should be emphasized that these principles are heuristic and continue to evolve rapidly with model scale and training methods. This paper does not provide a unified quantitative benchmark to support judgments such as "better."

9.3 An Explanation From the Perspective of Information Selection

From the perspective of information selection, the difference between discrete and continuous approaches lies in the strength of the constraint. Discretization creates a hard constraint through a finite vocabulary, limiting the upper bound on the number of bits used to encode a single index to approximately log2 K bits, although the overall bitrate also depends on the number of tokens, the number of codebook levels, and the distribution. Continuous representations impose soft constraints, or almost no explicit constraints, allowing information to flow with real-valued precision. Both seek a balance between retaining sufficient information to complete the task and constraining representational complexity, but they choose different forms of constraint. This perspective helps place the two seemingly opposing methods on the same spectrum.

10. Open Questions and Outlook

10.1 Unified Tokenizers and Arbitrary Modalities

How to design a truly unified tokenizer that can cover text, images, speech, and even more modalities remains an open question. Existing solutions are mostly optimized for a single modality, and unified cross-modal tokenization still involves tension between information compression and semantic alignment. Early work such as Unified-IO [24] attempted to process tasks across multiple modalities using a unified input-output space, but it has not fully solved the problem of unifying tokenization across heterogeneous modalities.

© 2026 The Author(s). Authors retain copyright in their articles.

10.2 Dynamic and Task-Adaptive Quantization

Discretization is usually static: once a vocabulary is trained, it is fixed. Related improvements can be divided into two directions. The first is codebook-free or simplified quantization. For example, FSQ [25] replaces a learnable codebook with deterministic finite scalar rounding, thereby avoiding codebook collapse. The second is dynamic or task-adaptive quantization and codebook allocation, which allows discretization to adjust to the task or content. Both directions point toward improving the efficiency and flexibility of tokenization.

10.3 Theorizing Tokenization and Developing Testable Criteria

Current discussions of when to use discrete representations and when to use continuous representations mostly remain empirical. Formalizing tokenization with theoretical tools such as the information bottleneck and rate-distortion theory, and providing criteria that can be tested by subsequent work, is a long-term goal for moving the field from empirical findings toward theory.

11. Conclusion

Taking the discrete-continuous spectrum as its main thread, this paper reviews representative strategies for Transformer tokenization across modalities and compares discrete and continuous approaches in terms of task suitability, information efficiency, vocabulary compatibility, and training characteristics. This paper makes three main contributions. First, it constructs an organizational framework based on the discrete-continuous spectrum and distinguishes the relevant mechanisms at three levels: input tokenization, latent-variable tokenization, and discretization of training targets. Second, it systematically compares discrete and continuous approaches at the four levels above. Third, it provides a unified explanation of the choices between the two from the perspective of information selection and summarizes several empirical design principles concerning when to use discrete representations and when to use continuous representations. Overall, discrete and continuous representations are not opposites in which one is superior to the other; they are different choices on the same spectrum for different tasks and constraints. It should be noted that this paper is a selective review and that the relevant design principles are heuristic judgments whose generality remains to be tested quantitatively and systematically. In addition, this paper does not discuss modalities such as video and, as a selective review, does not exhaust all new developments since 2026. Future work can continue in directions such as unified tokenizers, dynamic quantization, and the theorization of tokenization.

References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[2] Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[3] Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in neural information processing systems, 33, 12449-12460.

[4] Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., & Mohamed, A. (2021). HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 3451-3460. https://doi.org/10.1109/TASLP.2021.3122291

[5] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021, July). Learning transferable visual models from natural language supervision. In International conference on machine learning (pp. 8748-8763). PmLR.

[6] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., ... & Sutskever, I. (2021, July). Zero-shot text-to-image generation. In International conference on machine learning (pp. 8821-8831). Pmlr.

[7] Van Den Oord, A., & Vinyals, O. (2017). Neural discrete representation learning. Advances in neural information processing systems, 30.

© 2026 The Author(s). Authors retain copyright in their articles.

[8] Li, J., Fu, Y., Liu, J., Cao, L., Ji, W., Yang, M., ... & Yang, M. H. (2026). Discrete tokenization for multimodal llms: A comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[9] Jia, J., Gao, J., Xue, B., Wang, J., Cai, Q., Chen, Q., Zhao, X., Jiang, P., & Gai, K. (2025). From principles to applications: A comprehensive survey of discrete tokenizers in generation, comprehension, recommendation, and information retrieval. arXiv. https://doi.org/10.48550/arXiv.2502.12448

[10] Mousavi, P., Maimon, G., Moumen, A., Petermann, D., Shi, J., Wu, H., Yang, H., Kuznetsova, A., Ploujnikov, A., Marxer, R., Ramabhadran, B., Elizalde, B., Lugosch, L., Li, J., Subakan, C., Woodland, P., Kim, M., Lee, H.-Y., Watanabe, S., ... Ravanelli, M. (2025). Discrete audio tokens: More than a survey! arXiv. https://doi.org/10.48550/arXiv.2506.10274

[11] Xu, P., Zhu, X., & Clifton, D. A. (2023). Multimodal learning with transformers: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(10), 12113-12132. https://doi.org/10.1109/TPAMI.2023.3275156

[12] Li, Y., Qian, R., Pan, B., Zhang, H., Huang, H., Zhang, B., ... & Chen, Z. (2026, April). Manzano: A simple and scalable unified multimodal model with a hybrid vision tokenizer. In International Conference on Learning Representations (Vol. 2026, pp. 32102-32118).

[13] Wang, Y., Lin, Z., Teng, Y., Zhu, Y., Ren, S., Feng, J., & Liu, X. (2025, October). Bridging continuous and discrete tokens for autoregressive visual generation. In 2025 IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 18596-18605). IEEE.

[14] Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., & Carreira, J. (2021, July). Perceiver: General perception with iterative attention. In International conference on machine learning (pp. 4651-4664). PMLR.

[15] Li, T., Tian, Y., Li, H., Deng, M., & He, K. (2024). Autoregressive image generation without vector quantization. In Advances in Neural Information Processing Systems (Vol. 37). https://doi.org/10.52202/079017-1797

[16] Bao, H., Dong, L., Piao, S., & Wei, F. (2021). Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254.

[17] Yu, Q., Weber, M., Deng, X., Shen, X., Cremers, D., & Chen, L. C. (2024). An image is worth 32 tokens for reconstruction and generation. Advances in Neural Information Processing Systems, 37, 128940-128966.

[18] Défossez, A., Copet, J., Synnaeve, G., & Adi, Y. (2022). High fidelity neural audio compression. arXiv preprint arXiv:2210.13438.

[19] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R. (2022, June). Masked autoencoders are scalable vision learners. In 2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR) (pp. 15979-15988). IEEE.

[20] Chameleon Team. (2024). Chameleon: Mixed-modal early-fusion foundation models. arXiv. https://doi.org/10.48550/arXiv.2405.09818

[21] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., & Joulin, A. (2021, October). Emerging properties in self-supervised vision transformers. In 2021 IEEE/CVF international conference on computer vision (ICCV) (pp. 9630-9640). IEEE.

[22] Razavi, A., Van den Oord, A., & Vinyals, O. (2019). Generating diverse high-fidelity images with vq-vae-2. Advances in neural information processing systems, 32.

[23] Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). wav2vec: Unsupervised pre-training for speech recognition. arXiv preprint arXiv:1904.05862.

[24] Lu, J., Clark, C., Zellers, R., Mottaghi, R., & Kembhavi, A. (2022). Unified-io: A unified model for vision, language, and multi-modal tasks. arXiv preprint arXiv:2206.08916.

© 2026 The Author(s). Authors retain copyright in their articles.

[25] Mentzer, F., Minnen, D., Agustsson, E., & Tschannen, M. (2024, May). Finite scalar quantization: Vq-vae made simple. In International Conference on Learning Representations (Vol. 2024, pp. 51772-51783).

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgment

This paper is an output of the science project.

Copyrights

Copyright for this article is retained by the author (s), with first publication rights granted to the journal. This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/4.0/).

© 2026 The Author(s). Authors retain copyright in their articles.