上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Week 9 Extension: Discrete Tokenizers and Visual AR Models

Tao Huang

John Hopcroft Center, School of Computer Science, Shanghai Jiao Tong University

https://taohuang.info/cs3317

https://oc.sjtu.edu.cn/courses/89538

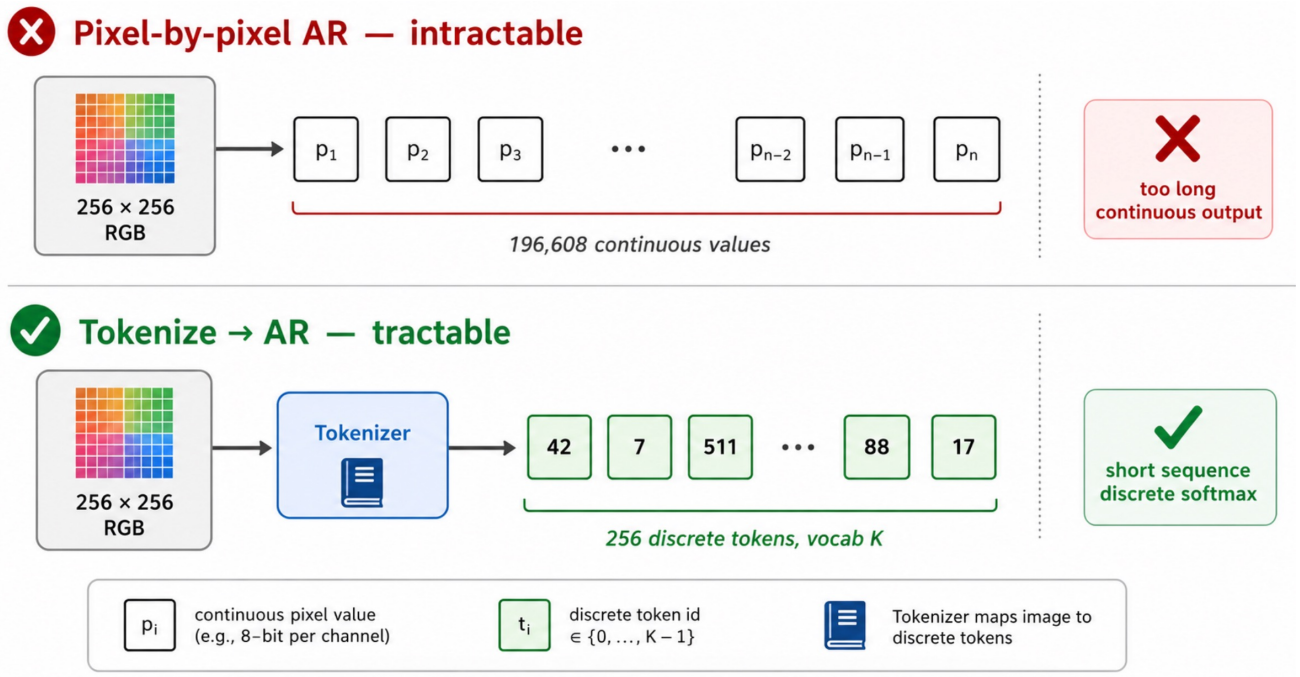
AI tools assisted in generating some figures in these slides. All such content has been reviewed, and the instructor is responsible for its accuracy.

Why Discrete Tokens for Visual Generation?

L23's recipe: factorize p(x) = ∏ p(x_t | x_<t), train a transformer to predict the next token. Worked beautifully for text — but for images?

Obstacle: a 256×256 RGB image is 196,608 continuous values. Pixel-by-pixel AR means a 196k-step sequence over a continuous output — too long, and softmax doesn't apply.

The fix in two acts: (1) tokenize the image into a short discrete sequence, (2) run autoregression on the tokens.



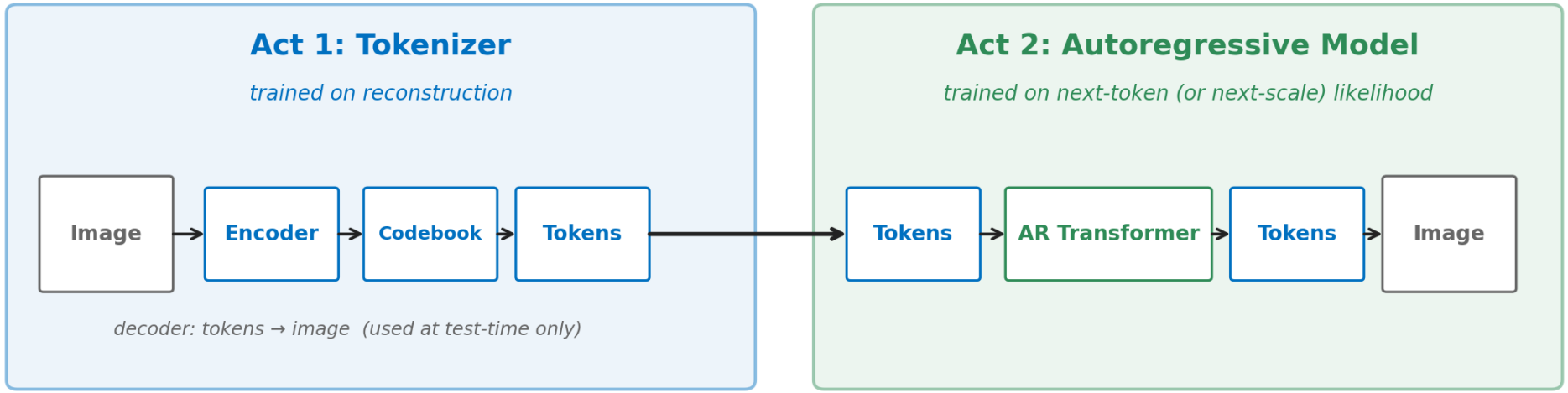
The Two-Act Plan

Tokenizer: image → grid of discrete tokens (and back). Trained on reconstruction.

Autoregressive model: predicts token sequences. Trained on next-token (or next-scale) likelihood.

The two are trained separately — tokenizer first, then the AR model on the frozen token codes.

This decoupling is what makes the recipe scale: the tokenizer is a one-time investment.



Decoupled training: tokenizer is a one-time investment, AR scales like an LLM

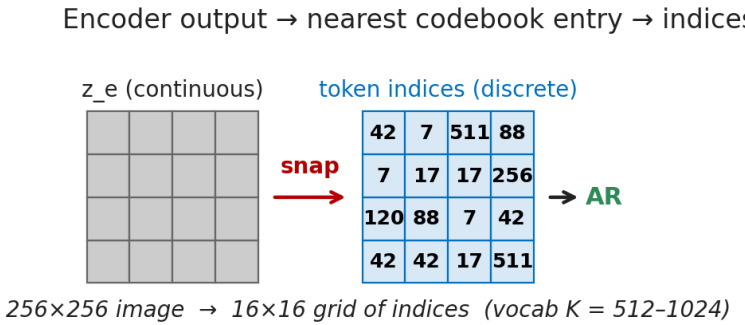
VQ-VAE: The Codebook Idea

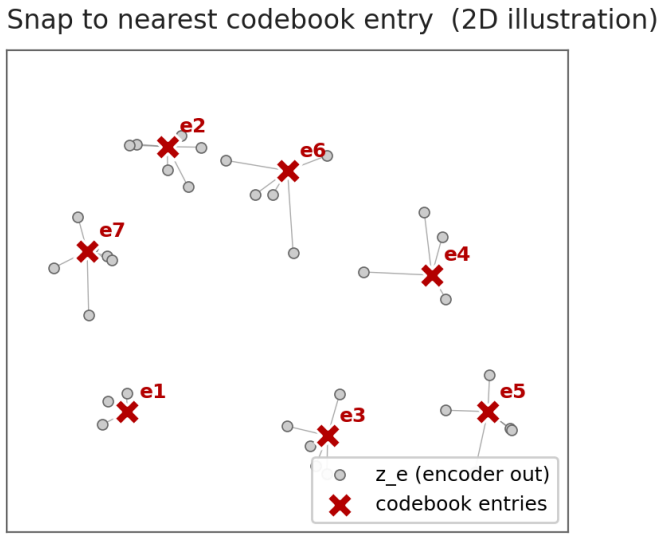
Oord et al. Neural Discrete Representation Learning. NeurIPS 2017

Snap each encoder output vector to its nearest entry in a learned codebook of size K. The image becomes a grid of codebook indices — a discrete token sequence.

Three losses: reconstruction, codebook (move codes toward encoder outputs), commitment (keep encoder outputs near codes). Argmin is non-differentiable → straight-through estimator. Codebook updated by EMA in practice.

Result: a 256×256 image becomes ~256 tokens from a vocabulary of 512–1024.





VQ-GAN: Perceptual + Adversarial Decoder

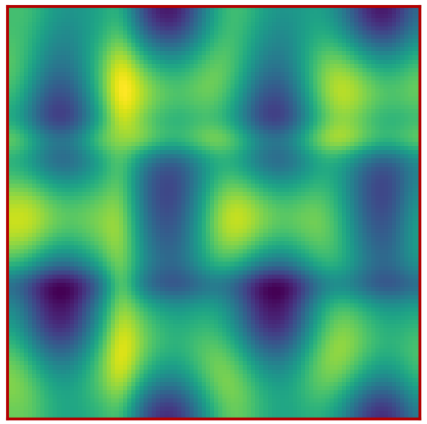
Esser et al. Taming Transformers for High-Resolution Image Synthesis. CVPR 2021

VQ-VAE reconstructions were blurry — same blurriness problem L24 flagged for VAEs.

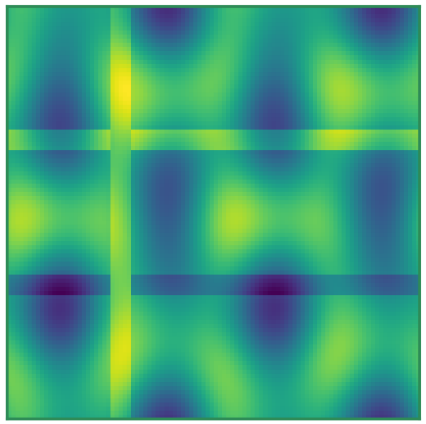
Fix: replace L2 reconstruction loss with perceptual + adversarial loss on the decoder. The decoder is trained as a GAN generator (callback to L25). Same codebook + straight-through machinery as VQ-VAE — only the decoder objective changed.

Indices alone now preserve visual quality. This is what made transformer-based image generation viable.

VQ-VAE (L2 reconstruction)
blurry — same VAE pathology



VQ-GAN (perceptual + adversarial)
sharp — usable for AR generation



Same architecture, different decoder loss

Codebook Collapse → FSQ / LFQ

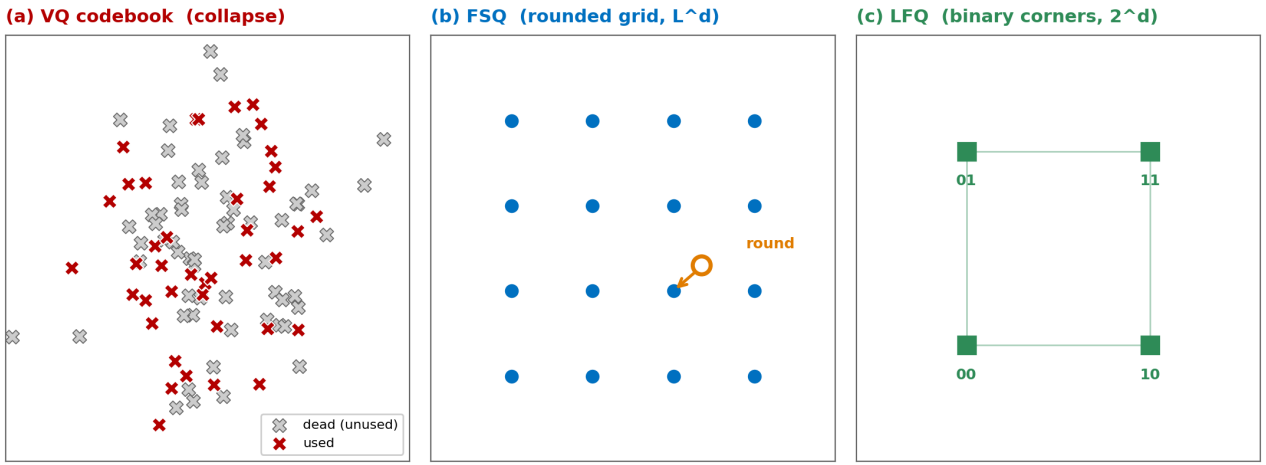
Mentzer et al. FSQ. ICLR 2024 / Yu et al. MAGVIT-v2 (LFQ). ICLR 2024

The pain: VQ codebooks of 16k+ entries rarely use more than a few thousand. Tuning EMA, commitment weight, and dead-code resampling became dark art.

FSQ: drop the codebook. Project to dim d, round each component to L levels. The "codebook" is the implicit grid of L^d points.

LFQ (MAGVIT-v2): the binary case — d ≈ 18, L = 2. Vocab of 2^18 ≈ 260k tokens, no learned embeddings, no collapse.

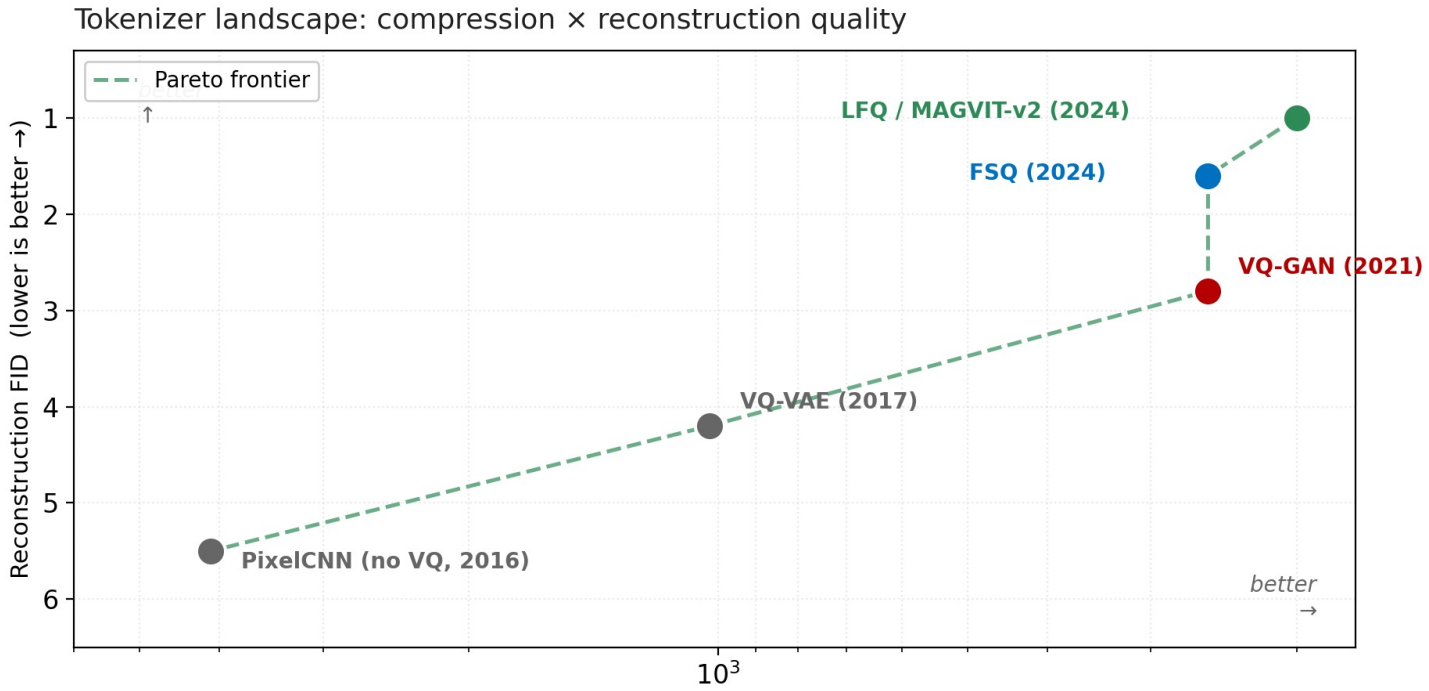
Same downstream AR quality, far simpler training.



Tokenizer Landscape

Three axes: compression ratio (tokens per image), reconstruction quality (rFID), downstream AR perplexity. They trade off — more compression → worse reconstruction → harder AR task.

Modern tokenizers (LFQ, FSQ) sit on a better Pareto frontier than VQ-VAE — the AR model on top is what we look at next.



Modern tokenizers (LFQ, FSQ) achieve ~3× more compression at lower FID than VQ-VAE.

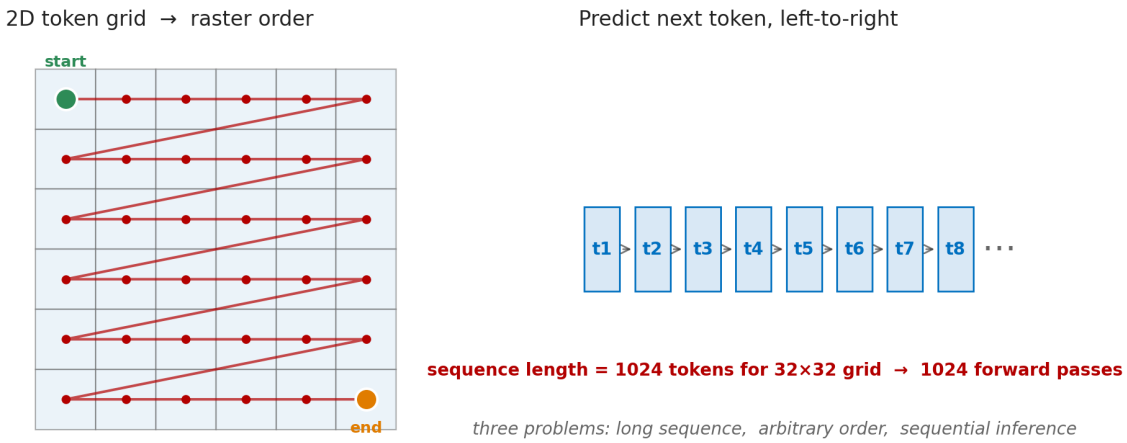
Raster-Scan AR: The Baseline

Esser et al. (2021) "Taming" / Parmar et al. Image Transformer. ICML 2018

Simplest recipe: flatten the 2D token grid into a 1D sequence (raster order, left-to-right, top-to-bottom). Train a standard decoder-only transformer — exactly L23's recipe applied to image tokens.

Three problems become visible:

- Sequence length: 1024 tokens for 32 × 32 grid → quadratic attention cost
- Arbitrary order: no natural left-to-right in images
- Inference is sequential: 1024 forward passes per image



Parallel Decoding: MaskGIT

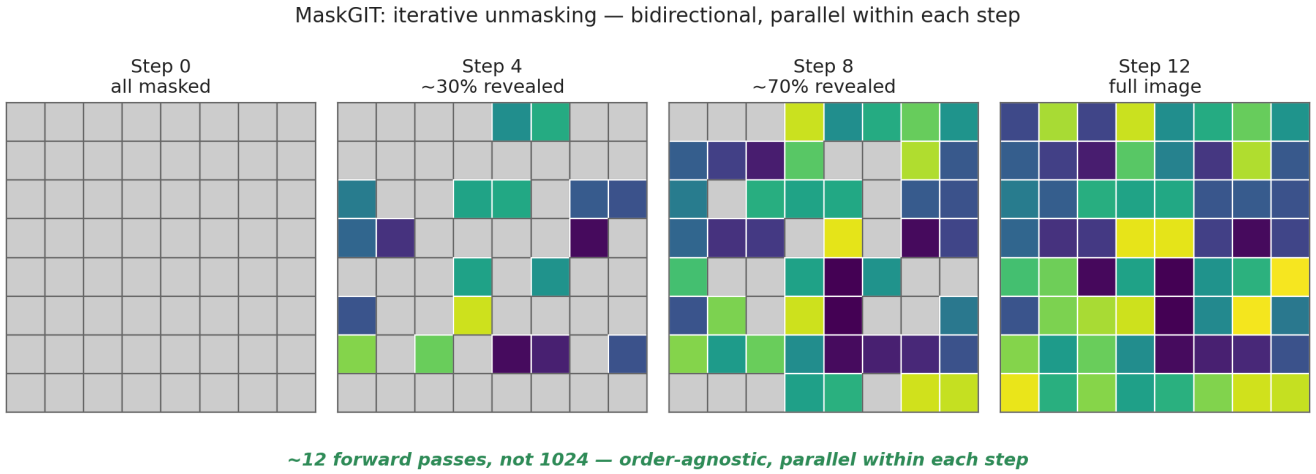
Chang et al. MaskGIT: Masked Generative Image Transformer. CVPR 2022

Reframe: image generation as iterative unmasking, not left-to-right prediction.

Training: random subset of tokens masked; bidirectional transformer predicts all masked tokens at once. (BERT objective applied to image tokens.)

Inference: start with all tokens masked. Each step, predict all positions, keep the most confident k%, mask the rest, repeat.

8–16 steps instead of 1024. Order-agnostic, parallel within each step. Cost: not strictly autoregressive — sacrifices exact likelihood for speed.



VAR: Next-Scale Prediction

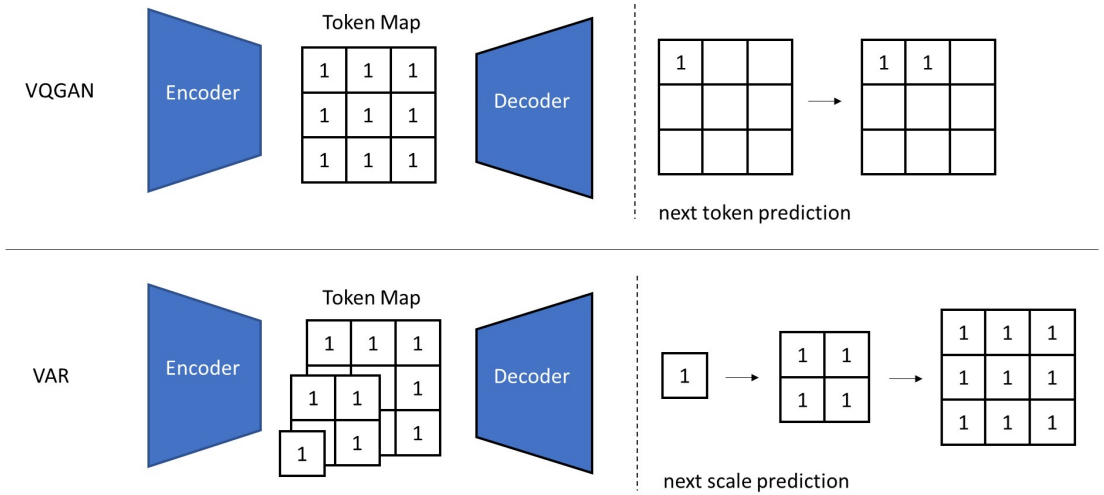
Tian et al. Visual Autoregressive Modeling. NeurIPS 2024 (Best Paper)

Key reframe: autoregress over scales, not over tokens.

Tokenizer produces a hierarchy: 1×1 → 2×2 → 4×4 → … → 16×16 (multi-scale VQ-VAE). AR model predicts the next finer scale, conditioned on all coarser scales. Within each scale, all tokens predicted in parallel.

Why it works: matches the natural coarse-to-fine structure of images. No arbitrary raster order. ~10 steps total.

Headline: first AR image model to beat diffusion (DiT) on ImageNet 256×256.

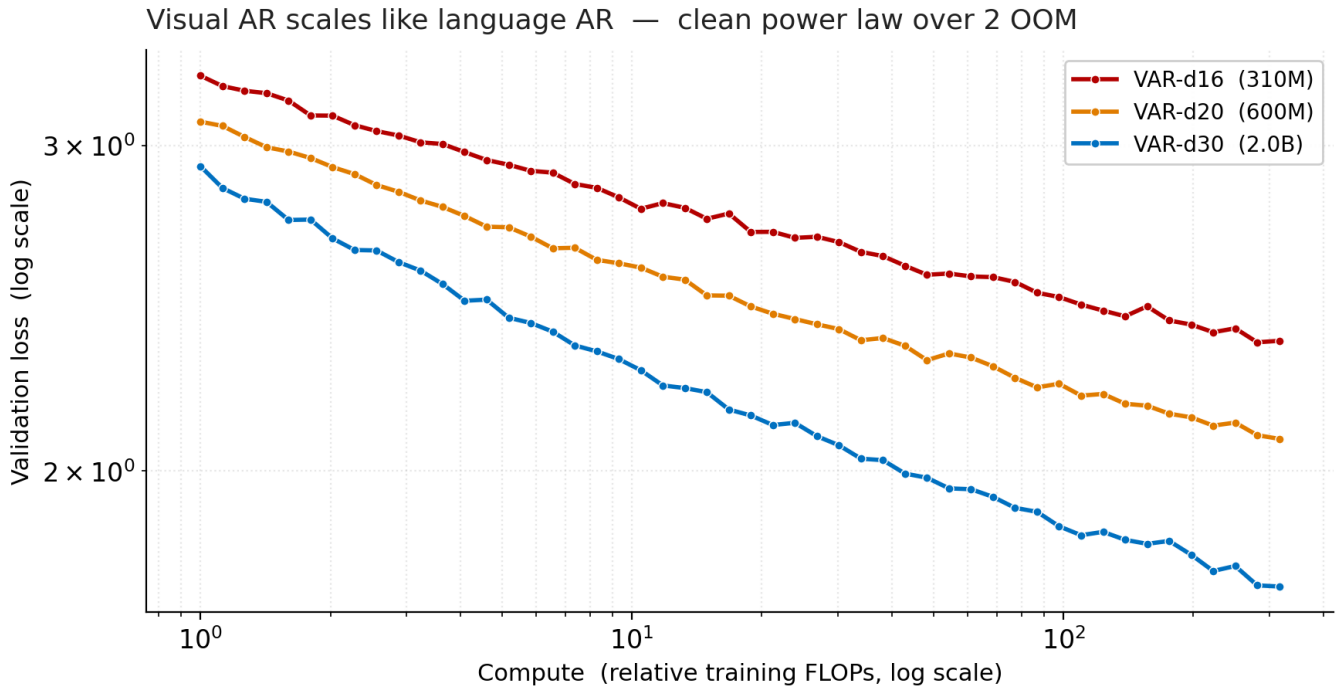


Scaling Laws for Visual AR

Tian et al. (2024) — second contribution of the VAR paper

VAR's other claim: visual AR exhibits clean power-law scaling in compute and parameters, like LLMs. Test loss vs compute follows a straight line on log-log over two orders of magnitude.

Not clearly true for prior visual generators (GANs are erratic; diffusion scaling laws emerged only recently). Implication: visual generation may be entering its "scale is all you need" era.



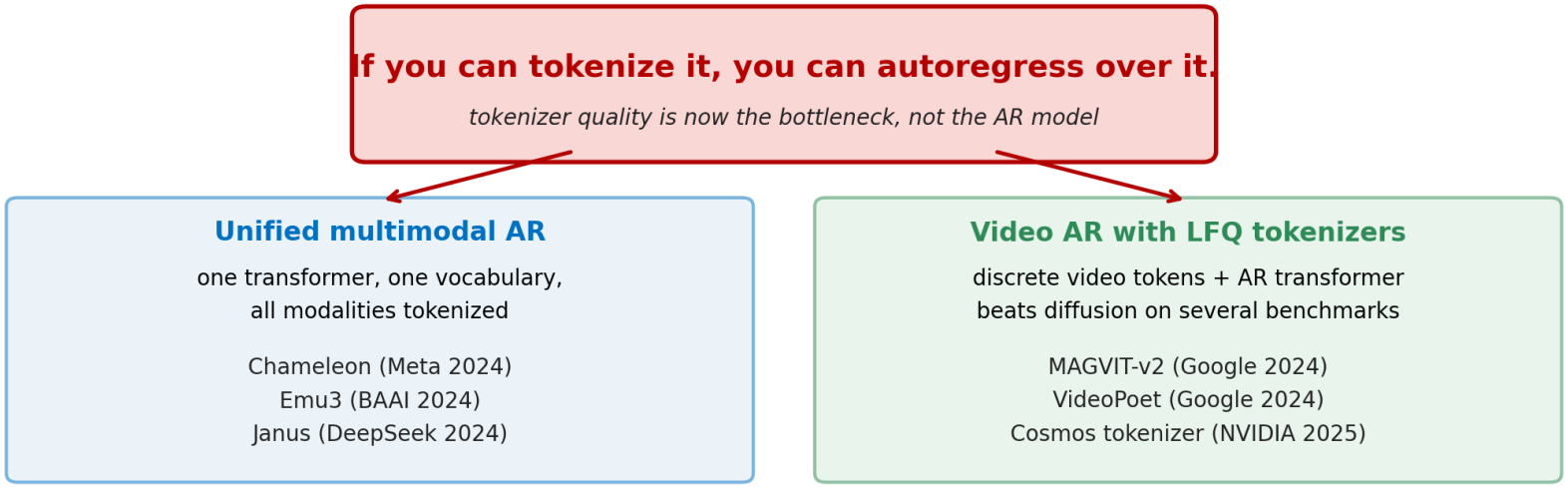
VAR's second contribution: visual AR exhibits clean power-law scaling — this is the result that made it NeurIPS Best Paper.

Where the Field Went Next

Unified multimodal AR — one transformer, one vocabulary, all modalities tokenized: Chameleon (Meta 2024), Emu3 (BAAI 2024), Janus (DeepSeek 2024). Bridges to L28.

Video AR with LFQ tokenizers — MAGVIT-v2 (2024), VideoPoet (2024). Discrete video tokens + AR transformer beats diffusion on several video benchmarks.

Thesis: if you can tokenize it, you can autoregress over it.



References

Tokenizers

[1] Oord et al. Neural Discrete Representation Learning (VQ-VAE). NeurIPS 2017

[2] Esser et al. Taming Transformers for High-Resolution Image Synthesis (VQ-GAN). CVPR 2021

[3] Mentzer et al. Finite Scalar Quantization: VQ-VAE Made Simple (FSQ). ICLR 2024

[4] Yu et al. Language Model Beats Diffusion — Tokenizer is Key (MAGVIT-v2 / LFQ). ICLR 2024

Visual Autoregressive Models

[5] Parmar et al. Image Transformer. ICML 2018

[6] Chang et al. MaskGIT: Masked Generative Image Transformer. CVPR 2022

[7] Tian et al. Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction (VAR). NeurIPS 2024 (Best Paper)

Multimodal & Video

[8] Chameleon Team. Chameleon: Mixed-Modal Early-Fusion Foundation Models. arXiv 2024

[9] Wang et al. Emu3: Next-Token Prediction is All You Need. arXiv 2024

[10] Kondratyuk et al. VideoPoet: A Large Language Model for Zero-Shot Video Generation. ICML 2024