TuringViT: Making SOTA Vision Transformers Accessible to All

Foundation Model Team, Xpeng Inc.
https://turingvit.github.io

Abstract

Modern VLMs and VLA systems commonly adopt off-the-shelf ViTs such as SigLIP2 as visual encoders, but diverse downstream requirements in latency, temporal modeling, and VLM integration often call for customized SOTA-level ViTs. Training such encoders remains beyond the reach of much of the community, as it requires massive image-text data, while standard softmax attention makes high-resolution or dynamic-resolution pretraining prohibitively costly and often forces low-resolution pretraining followed by post-hoc adaptation. **TuringViT** addresses these challenges with three key designs: **Turing Linear Attention (TLA)** for efficient sequence modeling, **VISTA-Curation** to construct supervision-rich image-video training data, and native dynamic-resolution pretraining that supports flexible inputs from the start and transfers seamlessly to downstream VLMs. As a result, TuringViT outperforms leading open-source ViT baselines with only 10% of the data, achieves stronger downstream VLM performance, and delivers substantially better latency scaling on high-resolution inputs. Our scaling-law analysis further shows that TuringViT continues to improve predictably with curated data scale, far from saturation. Its fast adaptation, hardware-friendly design, and efficient deployment have made it a unified visual foundation across XPeng's AI systems. More broadly, TuringViT provides a reproducible pipeline that dramatically lowers the cost for the community to train, customize, and deploy SOTA-level ViTs, moving toward making such Vision Transformers accessible to all.

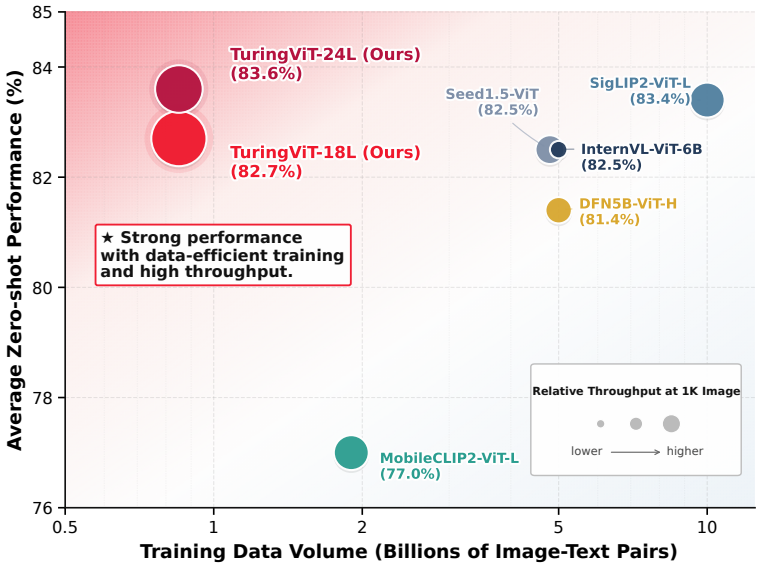


Figure 1: Zero-shot performance, training data scale, and 1K-resolution throughput of vision transformer encoders. TuringViT achieves stronger accuracy–efficiency trade-offs than leading open-source baselines with only 10% of the training data.

Technical Report.

1 Introduction

In modern VLMs [1–3] and VLA [4, 5] systems, the visual encoder has become the primary interface through which language models perform increasingly complex multimodal perception, reasoning, and action. A common practice is to adopt powerful off-the-shelf encoders pretrained on large-scale image-text data [6, 7]. Recent models such as SigLIP2 [8] have shown strong general-purpose performance, but simply reusing an existing encoder is not always sufficient. Different downstream settings may impose distinct requirements on resolution, latency, temporal modeling, domain coverage, and integration with the language model [9–11]. This creates a growing need for SOTA-level visual encoders that can be trained or customized for specific use cases, yet achieving this goal under realistic resource budgets remains challenging.

This difficulty arises from multiple sources. First, the compute cost of visual encoding grows rapidly as visual sequences become longer. High-resolution images, multi-image inputs, and video frames all increase the number of visual tokens, making standard softmax-attention ViTs expensive to train and deploy [12–14]. Second, strong ViTs often rely on massive web-scale image-text data, yet reproducible and actionable recipes for turning such noisy data into high-quality visual-language supervision remain limited [15–18]. Simply scaling data can introduce noise, weak alignment, and redundant supervision, making training less efficient. Third, most pretrained ViTs are still trained with fixed-resolution inputs, whereas downstream VLMs often require flexible resolution handling. Existing solutions typically rely on tiling-based processing or additional continual training for dynamic resolution, which introduces extra cost, creates a mismatch between visual pretraining and downstream VLM usage, and may lead to suboptimal performance [19–21].

Together, these barriers make SOTA visual encoders [8, 22, 23] inaccessible to much of the broader research and application community. In this report, we revisit ViT design from the perspective of accessibility: can SOTA-level visual encoders be trained, adapted, and deployed under controlled and reproducible resource budgets? Our answer is TuringViT, a VLM-native Vision Transformer designed for efficiency across architecture, data, and training.

We first address the most fundamental source of cost: the model architecture itself. TuringViT introduces **Turing Linear Attention (TLA)** as its dominant attention operator, replacing most softmax-attention layers with a linear-complexity alternative tailored to high-resolution and dynamic-resolution visual inputs [24–27]. TLA combines efficient global context aggregation with sequence-length-aware normalization and an input-dependent output gate, improving stability under variable-length token sequences while preserving local and high-frequency visual responses. To avoid sacrificing explicit token-to-token interaction, TuringViT further retains a small number of standard multi-head softmax-attention layers, inserted periodically among TLA layers. This hybrid design makes linear attention the main computational path while using full attention only sparsely for

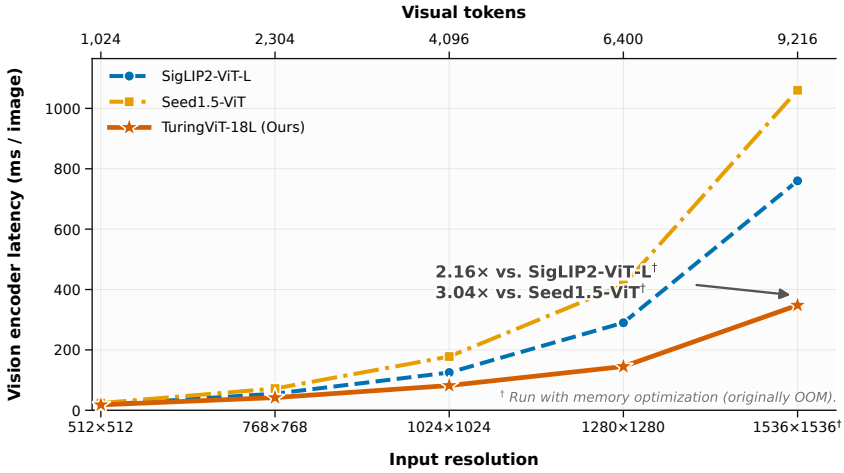


Figure 2: Inference latency scaling with input resolution and visual token count. Latency is measured per image using FP16 TensorRT on an NVIDIA GeForce RTX 3080 Ti with CUDA 11.6; TuringViT shows flatter high-resolution scaling than standard ViT baselines.

global routing and token-level interaction. As shown in Figure 2, TuringViT exhibits a much flatter inference-latency curve than standard softmax-based ViTs as the visual sequence length increases, reducing practical sequence-length-to-latency scaling from quadratic to near-linear. This architectural efficiency is crucial for making high-resolution and dynamic-resolution ViT training feasible under controlled resource budgets.

Complementing the efficient architecture, we build **VISTA-Curation** (**Vi**sion Data Curation via Image-Video **S**coring and **T**emporal **A**ggregation), a supervision-enhancing multimodal data curation pipeline that enables TuringViT to outperform leading open-source ViT baselines using only 10% of the data scale. Rather than relying on brute-force data scaling, VISTA-Curation increases the training value of each sample by converting noisy image-language and video-language data into more grounded, discriminative, and temporally informative supervision. For image-language data, it filters low-quality images, generates multiple visually grounded caption candidates, and selects the best caption through relative image-text scoring and textual informativeness ranking. This process suppresses ambiguous or weakly aligned captions while retaining descriptions that are faithful, specific, and useful for CLIP-style alignment. For video-language data, VISTA-Curation decomposes raw videos into compact clips, samples representative frames, filters clips by semantic and motion consistency, and fuses local frame-level details with global temporal semantics. The resulting video captions provide transformation-aware supervision across changes in viewpoint, object state, motion, and scene context. Together, these curated image and video data shift visual-language pretraining from simply using more data to extracting more reliable supervision from each sample. Our scaling-law analysis further shows predictable gains as the curated data scale increases, indicating that TuringViT remains far from saturated.

TuringViT is trained with dynamic resolution throughout the entire visual pretraining process while preserving high training efficiency. Instead of restricting the model to a fixed input size, we allow images with different resolutions and aspect ratios from the beginning. To make this efficient, we adopt a progressive resolution schedule: during the first 80% of training epochs, we cap the maximum image size at 512, which is already comparable to the maximum resolution supported by many existing ViT encoders; in later stages, we remove this constraint and train with larger images and video data. Benefiting from its linear-complexity architecture, TuringViT maintains efficient training even as visual sequence length increases. This native dynamic-resolution training reduces the need for post-hoc resolution adaptation and leads to consistently stronger downstream VLM performance.

TuringViT delivers strong results across both vision and multimodal evaluations. It achieves an average score of **83.6** on six zero-shot classification benchmarks and **78.9** on two retrieval benchmarks, outperforming SigLIP2 [8] while using only **10%** of the data scale. When integrated into downstream VLMs, TuringViT also brings consistent improvements across multimodal benchmarks. These results show that TuringViT is not merely a faster replacement for standard softmax-based visual encoders, but a stronger, more data-efficient, and more VLM-compatible visual backbone.

2 Methodology

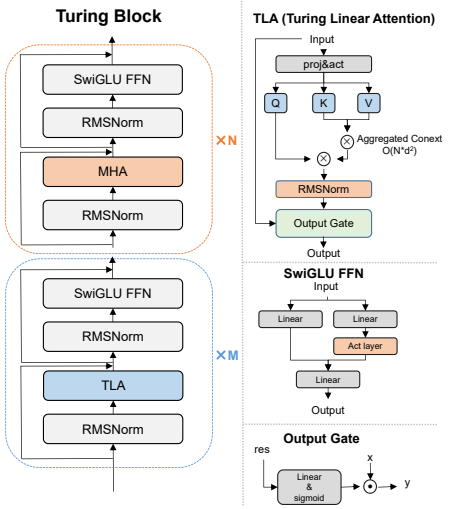
TuringViT consists of three key components: an efficient visual architecture, a large-scale visual supervision pipeline, and a native dynamic-resolution training paradigm.

First, we introduce **Turing Linear Attention (TLA)**, a linear-attention-dominant architecture designed for efficient visual sequence modeling as shown in Figure 3. Second, we develop **VISTA-Curation**, a large-scale image-video curation framework that improves supervision quality and data efficiency. Finally, we adopt **native dynamic-resolution training**, enabling the visual encoder to learn directly from variable-resolution visual inputs throughout pretraining.

Together, these design choices allow TuringViT to achieve strong visual representation quality while remaining efficient and naturally compatible with downstream multimodal systems. The remainder of this section describes these components in detail.

2.1 Linear-Attention-Dominant Native-Resolution Architecture

TuringViT is designed to support high-resolution and dynamic-resolution visual inputs for VLM usage. Instead of resizing all images to a fixed square resolution, TuringViT keeps images at native or dynamically selected resolutions and converts them into variable-length visual token sequences with



TuringViT Configurations

Setting	TuringViT-18L	TuringViT-24L
M	5	5
N	1	1
Num blocks	**3**	**4**
Total layers	**18**	**24**
Patch size	16 × 16	16 × 16
pos embedding	Dinov3 RoPE	Dinov3 RoPE
Head dim	64	64
Num heads	20	20
Embed dim	1280	1280
MLP ratio	4	4

Figure 3: TuringViT block architecture and model configurations. (Left) The backbone is built on repeated Turing Blocks, Turing Linear Attention (TLA) aggregates global context; its sequence-length-aware normalization and input-dependent output gate stabilize variable-token training and preserve high-frequency local details. **(Right)** Model configurations for TuringViT-18L and TuringViT-24L, which scale in blocks and total layers while sharing identical patch size (16×16), 2D RoPE, embed/head dimension, and MLP ratio.

a patch size of 16×16 [21]. This design allows the visual encoder to process images with different aspect ratios and resolutions under a unified architecture.

To provide spatial information for variable token grids, we adopt a Dinov3-style [28] two-dimensional rotary positional embedding [29, 30]. The 2D RoPE is applied to the query and key features according to their spatial coordinates on the patch grid. Compared with fixed absolute positional embeddings, this design naturally supports different image sizes and aspect ratios without requiring position-embedding interpolation.

The main computation module of TuringViT is **Turing Linear Attention (TLA)**. Given an input token sequence $X \in \mathbb{R}^{N \times C}$, TLA first projects the hidden states into query, key, and value features Q, K, V, where $G(X)$ denotes the input-dependent output gate, $\sigma(\cdot)$ is the sigmoid function, N is the input sequence length, and d is the head dimension. We use SiLU as the kernel function $\phi(\cdot)$ and apply it to QKV features. The TLA output is computed as

$$\begin{aligned} G(X) &= \sigma(XW_g), \\ \mathrm{TLA}(X) &= G(X) \odot \frac{\phi(Q)\left(\phi(K)^\top \phi(V)\right)}{N\sqrt{d}}. \end{aligned} \tag{1}$$

The normalization by $N\sqrt{d}$ improves numerical stability under dynamic-resolution inputs with variable token lengths. Since linear attention aggregates global context efficiently, it may also smooth local details and high-frequency visual cues. The input-dependent sigmoid output gate provides an additional element-wise modulation path for preserving input-dependent local and high-frequency responses, thereby complementing the low-cost global mixing of linear attention.

Although TLA is used as the dominant attention operator, TuringViT still retains a small number of standard multi-head softmax attention layers. Specifically, we organize the network with a repeated Turing Block, where each block contains five TLA layers followed by one vanilla MHA layer:

$$\mathcal{B}_{\mathrm{Turing}} = [\mathrm{TLA}, \mathrm{TLA}, \mathrm{TLA}, \mathrm{TLA}, \mathrm{TLA}, \mathrm{MHA}]. \tag{2}$$

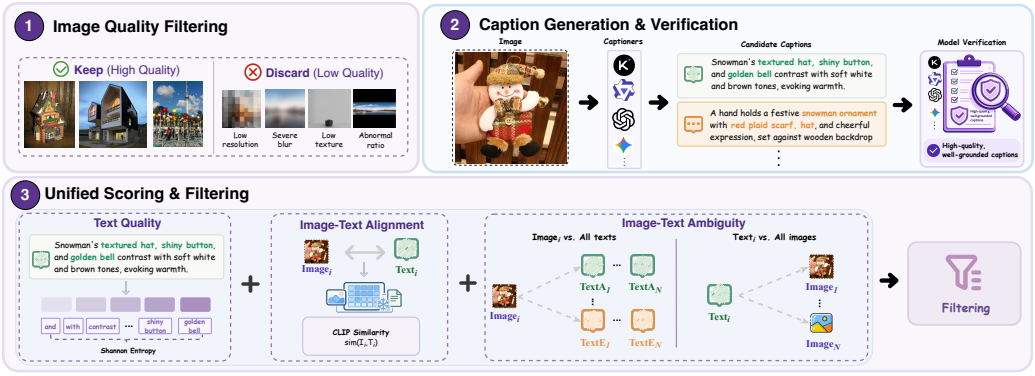


Figure 4: Overview of the image-language curation engine. The pipeline filters low-quality images, generates and verifies diverse candidate captions, and ranks all verified candidates in a shared comparison pool. The unified scoring function combines relative visual-language discriminativeness with textual informativeness, selecting captions that are visually grounded, semantically rich, and discriminative for CLIP-style pre-training.

In this design, vanilla MHA is sparsely inserted at the layer level rather than used in every Transformer layer. The TLA layers provide efficient global context aggregation for long visual sequences, while the periodically inserted MHA layers maintain explicit token-to-token interaction. Therefore, TuringViT is not a strictly linear-attention-only model; its practical efficiency comes from making linear attention the dominant path while using full softmax attention only periodically.

All layers follow a pre-normalization Transformer layout with RMSNorm [31]. Each attention layer is paired with a SwiGLU feed-forward network [32], following the block primitives commonly used in modern large language models. This gives the visual encoder a simple and VLM-friendly structure while keeping the computation efficient for high-resolution inputs.

We instantiate two model variants. **TuringViT-18L** contains three Turing Blocks, resulting in 18 total layers, including 15 TLA layers and 3 MHA layers. This variant is designed as an efficient and deployment-friendly backbone with strong throughput under dynamic-resolution inputs. **TuringViT-24L** contains four Turing Blocks, resulting in 24 total layers, including 20 TLA layers and 4 MHA layers. This larger variant further improves representation capacity while keeping the computation dominated by the linear-attention path.

2.2 Vision Data Curation via Image-Video Scoring and Temporal Aggregation

With an efficient vision encoder established, we further build **VISTA-Curation**, a data curation pipeline that selects visually grounded, semantically rich, and discriminative image-video captions for CLIP-style pre-training.

2.2.1 Image-Language Data Curation

As shown in Figure 4, image-language data curation aims to convert noisy web image-text pairs into high-quality supervision with stronger visual grounding and higher semantic density. Instead of directly using raw alt-text, we first filter low-quality images, then generate and verify multiple candidate captions to obtain descriptions that are more visually faithful, informative, and suitable for CLIP-style alignment.

Stage 1: Image Quality Filtering. We first collect large-scale image sources from public datasets such as DataComp-1B [16] and LAION-2B [15], and perform rule-based filtering to remove low-quality visual samples before caption generation. Images with extremely low resolution, severe blur, low texture, or abnormal ratios are discarded. This step prevents the captioning models from producing unreliable descriptions for images that are visually ambiguous or lack sufficient detail.

Stage 2: Caption Generation and Verification. We use carefully designed prompts to generate visually grounded and informative captions while suppressing hallucinations. For each retained image, multiple multimodal models and prompt variants produce diverse candidate captions, which

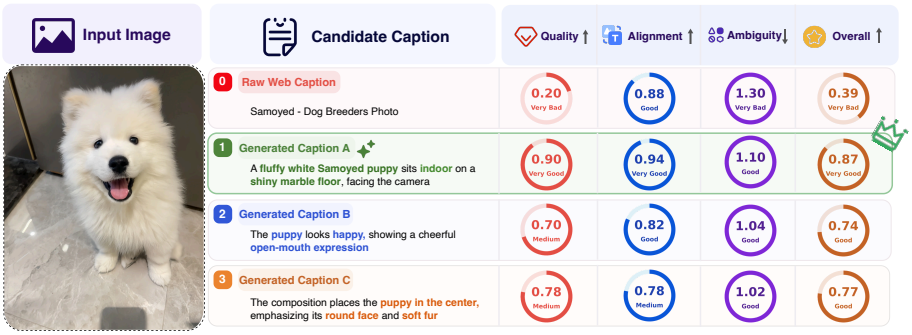


Figure 5: Qualitative example of multi-candidate recaptioning and scoring. The selected caption is more detailed, less ambiguous, and better aligned with the image than the original web caption.

are cross-verified for grounding and factual consistency. This converts weak web annotations into richer, semantically broader supervision.

Stage 3: Unified Scoring and Filtering. Given an image I_i, the caption generation and verification stages produce a filtered candidate set $\mathcal{C}_{I_i}$ from different captioners and prompt variants. Instead of scoring each candidate independently or only comparing captions within the same image, we collect all candidates that pass the preceding filtering stages into a shared comparison pool $\mathcal{P}$. This allows each image-caption candidate (I_i, c) to be evaluated relative to both its intra-image alternatives and candidates from other images. Such relative scoring strengthens both image-side and text-side filtering: a high-quality caption should not only align well with its paired image, but also be sufficiently discriminative, avoiding ambiguous descriptions that are interchangeable across different images or captions.

Following the relative scoring principle of s-CLIPLoss [33], we evaluate each image-caption candidate in a shared comparison space, without directly using the absolute image-text similarity as the only criterion. Given an image I_i and its filtered candidate caption set $\mathcal{C}_{I_i}$, we score each candidate caption $c \in \mathcal{C}_{I_i}$ by a relative visual-language quality function:

$$s_{\mathrm{vl}}(I_i, c) = -\ell_{\mathrm{sCLIP}}(I_i, c; \mathcal{P}), \tag{3}$$

where $\ell_{\mathrm{sCLIP}}(\cdot)$ denotes the relative image-text loss computed over a sampled comparison pool $\mathcal{P}$. Intuitively, this score considers not only the alignment between the current image and its paired caption, but also how well the caption can be distinguished from other captions and how selectively the image matches its paired text compared with other images.

To further emphasize textual informativeness, we compute an additional text quality score $s_{\mathrm{text}}(c)$ using caption-level statistics such as Shannon entropy. The final ranking score is defined as

$$S(I_i, c) = s_{\mathrm{vl}}(I_i, c) + \alpha \cdot \mathrm{Norm}\big(s_{\mathrm{text}}(c)\big), \tag{4}$$

where α balances relative visual-language discriminativeness and textual informativeness.

Different from scoring only a single generated caption, we apply this ranking process to all captions that pass the preceding filtering stages. These candidates include object-centric, attribute-focused, scene-level, relation-aware, and fine-grained captions. By comparing all filtered candidates within the same relative scoring space, our pipeline jointly strengthens both image-side and text-side filtering: visually misaligned captions are suppressed, while generic, redundant, ambiguous, or weakly discriminative captions are also penalized. For each image I_i, we select the caption with the highest final score:

$$c_i^* = \arg\max_{c \in \mathcal{C}_{I_i}} S(I_i, c). \tag{5}$$

This strategy encourages the selected captions to be not only visually grounded, but also semantically specific and discriminative. As shown in Figure 5, the raw web caption can obtain a reasonable alignment score because it is category-related, but its limited textual detail and high ambiguity lead to a lower overall ranking. In contrast, the selected generated caption achieves the best overall score by providing richer, visually grounded details while remaining well aligned with the image and less interchangeable with other samples.

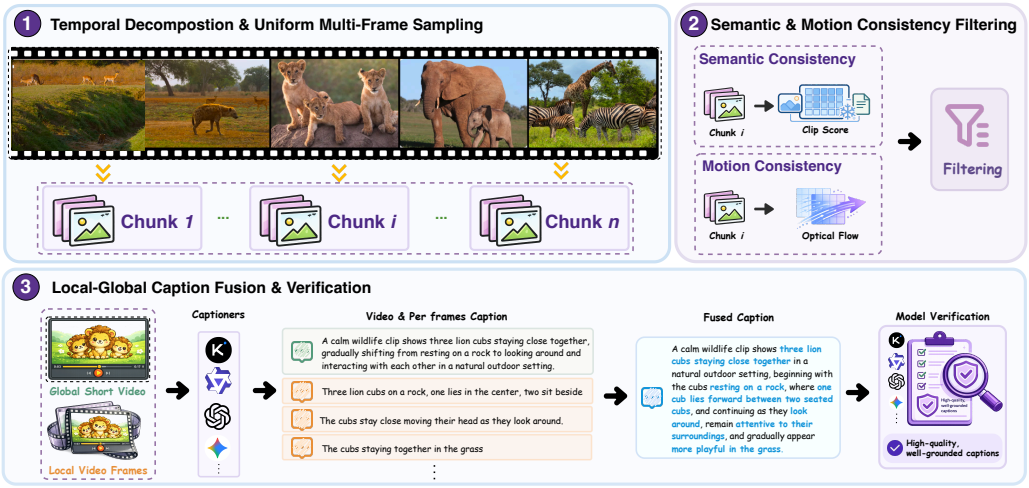


Figure 6: Overview of the proposed video-language data curation pipeline. Raw videos are decomposed into temporally contiguous short videos, from which uniformly sampled frames are organized as frame chunks. Each chunk is filtered using semantic consistency from pairwise CLIP frame similarities and motion consistency from optical-flow statistics. For the retained short videos, local frame-level captions and global video-level summaries are fused and verified by a VLM to remove low-quality or insufficiently grounded captions, yielding concise and temporally grounded video-language pairs for CLIP-style training.

2.2.2 Video-Language Data Curation

Beyond static image-text pairs, videos provide transformation-aware supervision by observing the same semantic entities across time, capturing objects, scenes, and actions under changing viewpoints, scales, poses, occlusions, lighting conditions, and temporal states. Such continuous observations help ViT learn more invariant, object-centric, and context-aware representations beyond isolated snapshots. However, raw videos are highly redundant, weakly annotated, and expensive to use directly, while simply sampling more frames often brings limited additional supervision due to adjacent-frame redundancy. As illustrated in Figure 6, we therefore curate videos into compact video-language pairs through temporal decomposition, uniform frame sampling, semantic-motion filtering, and local-global caption fusion, transforming long and noisy videos into concise, temporally grounded, and information-rich supervision for CLIP-style video training.

Stage 1: Temporal Decomposition and Uniform Frame Sampling. As shown in Figure 6, each raw video is first decomposed into temporally contiguous short videos based on visual-content changes with the assistance of multimodal models. For each retained short video, we uniformly sample multiple frames at fixed temporal intervals and organize them as frame chunks, following the same protocol used in CLIP-style video training.

Stage 2: Semantic and Motion Consistency Filtering. For each chunk, we evaluate the sampled frames from two complementary aspects: semantic consistency and motion consistency. The former is measured by encoding sampled frames with a CLIP image encoder and computing pairwise frame-to-frame similarities, while the latter is measured by computing optical-flow statistics between adjacent sampled frames. Short videos with excessively large motion or near-zero motion are filtered out, and only those with coherent semantics and informative temporal changes are retained.

Stage 3: Local-Global Caption Fusion and Verification. For each retained short video, we generate local frame-level captions and a global video-level summary. The local captions describe fine-grained visual details from the sampled frames, while the global summary captures the overall semantics and temporal dynamics of the short video. We then fuse them into a single caption and apply a VLM-based verification step to remove low-quality captions, especially those with insufficient grounding in the video content. The final retained captions are used as reliable video-language pairs for CLIP-style training.

Table 1: Setup for the progressive multi-stage ViT pre-training. The H800 GPU-hours* indicates total training cost, measured as the number of H800 GPUs multiplied by wall-clock training hours. All results in this table correspond to the TuringViT-24L pre-training setup.

Configuration	Stage 1	Stage 2	Stage 3	Stage 4
Data type	Unlabeled images	Image–text pairs	Image–text pairs	Image/Video–text pairs
Samples	55M	850M	850M	2M
Resolution	256×256	Range-constrained	Native	Native
Objective	MIM distillation	SigLIP + SuperClass	SigLIP + SuperClass	SigLIP + SuperClass
Batch size	16,384	65,536	8,192	1,024
Peak LR	3.0×10^{-3}	5.0×10^{-4}	5.0×10^{-6}	1.0×10^{-6}
Final LR	3.0×10^{-6}	6.0×10^{-6}	6.0×10^{-8}	2.0×10^{-8}
H800 GPU-hours*	9984	30720	12288	460

2.3 TuringViT Pre-training Stage

We pre-train TuringViT with a four-stage recipe designed for VLM-native visual encoding. The goal is not to introduce complex objectives, but to progressively expose the visual encoder to the same types of inputs used by downstream VLMs: dynamic-resolution images, high-resolution visual content, and mixed image-video data. Both TuringViT-18L and TuringViT-24L follow this recipe. The detailed training configuration is summarized in Table 1.

Stage 1: MIM-based visual initialization. We first pre-train the randomly initialized TuringViT backbone with masked image modeling (MIM) before large-scale image–text alignment [34]. Following the EVA-style feature reconstruction paradigm, we use EVA02-CLIP-E [35] as the teacher model. For each image, 40% of the image patches are randomly masked, and the student visual encoder is trained to reconstruct the teacher CLIP features at the masked locations. By reconstructing masked patch features, MIM supplies dense local supervision that encourages the linear-attention-dominant backbone to preserve fine-grained geometry and high-frequency visual cues, counteracting linear attention's bias toward global context aggregation.

Stage 2: Range-constrained dynamic-resolution image-text pre-training. After visual initialization, we train TuringViT with large-scale image-text contrastive learning. Different from fixed-resolution CLIP pre-training, we preserve the original aspect ratio and resize each image only when necessary, clamping its longer side to the $[256, 512]$ range. This range-constrained dynamic-resolution strategy exposes the model to variable-length visual token sequences from the beginning of contrastive pre-training, while keeping the training cost controlled. Even under this range-constrained dynamic-resolution setting, this stage already involves substantially longer visual token sequences than conventional low-resolution pre-training. Because TuringViT uses linear attention as the dominant computation path, it can efficiently train on these variable-length sequences while maintaining high throughput. We initialize the visual encoder from Stage 1 and the text encoder from the EVA02-CLIP-E [35] text encoder. For each image-text pair, the visual patch features are aggregated by attention pooling into a single 1024-dimensional image embedding. We align image and text embeddings by jointly optimizing the SigLIP loss [36] and the SuperClass loss [37].

Stage 3: Native-resolution image-text refinement. We further refine TuringViT with image-text contrastive training at native resolution. Unlike the range-constrained setting in Stage 2, this stage preserves the original aspect ratio and resolution of each image whenever possible, making visual preprocessing consistent with downstream VLM usage. The objective remains unchanged, allowing the model to strengthen high-resolution image-text alignment and learn finer visual details while benefiting from the efficiency of its linear-attention-dominant backbone.

Stage 4: Mixed Image/Video-Text Alignment. Finally, we adapt TuringViT with mixed image-text and curated video-text training, where video-text pairs are produced by the pipeline in Section 2.2.2. This compact video stage extends the encoder from static image representation to transformation-aware image-video representation, improving transferability for both image and video tasks. For a video with uniformly sampled T frames, we encode each frame with the image encoder and aggregate

frame features by attention pooling followed by temporal mean pooling:

$$v^{\text{vid}} = \text{Norm}\left(\frac{1}{T}\sum_{t=1}^{T}\text{AttnPool}(F_t)\right).$$

3 Experiments

We evaluate two TuringViT variants, TuringViT-18L and TuringViT-24L, both trained under the training paradigm described in Section 2.3 with dynamic-resolution image inputs. Our main models are trained on approximately 0.85B image-text pairs. We compare TuringViT with representative vision-language encoders at similar or larger scales, including MobileCLIP2-L [38], SigLIP2-L [8], and Seed1.5-ViT [39]. We conduct a comprehensive evaluation covering image zero-shot classification and robustness, image-text retrieval, frozen dense prediction transfer, ablation studies on data strategy, video training, and data scaling, as well as downstream VLM performance under a controlled vision-language training pipeline. These experiments are designed to examine not only the standalone visual representation quality of TuringViT, but also its transferability to dense perception, video-enhanced training, and multimodal reasoning scenarios.

3.1 Evaluating TuringViT in Zero-shot Classification and Retrieval

For zero-shot evaluation, we assess TuringViT on both image-level recognition and cross-modal retrieval benchmarks. For zero-shot classification, we evaluate models on ImageNet-1K [40], ImageNet-v2 [41], ObjectNet [42], ImageNet-A [43], ImageNet-R [44], and ImageNet-Sketch [45], and use the average score across these six datasets as the main robustness metric. We further evaluate zero-shot image-text retrieval on MS-COCO [46] and Flickr30K [47], covering both standard recognition robustness and image-text alignment.

Compared with SigLIP2-L [8], TuringViT-24L achieves a higher average zero-shot classification score while using only about 10% of the pretraining data. The gains are particularly clear on ImageNet-1K, ImageNet-v2, and ImageNet-A, indicating that dynamic-resolution training and the proposed data-centric pretraining recipe lead to strong zero-shot recognition and robustness under challenging visual distribution shifts. We also observe that TuringViT-24L still lags behind SigLIP2-L [8] on ObjectNet, ImageNet-R, and ImageNet-Sketch. These benchmarks emphasize object viewpoint changes, rendition-style distribution shifts, and sketch-like visual abstraction, which are closely tied to the coverage of difficult and out-of-distribution visual samples in pretraining data. Since TuringViT-

Table 2: Zero-shot performance across classification and retrieval benchmarks. All values are percentages and higher is better. Best results are in bold.

Model	MobileCLIP2-L	SigLIP2-L	Seed1.5-ViT	TuringViT-18L	TuringViT-24L
Model configuration					
Resolution	224	384	dyn.	dyn.	dyn.
Pretraining data	1.9B	10B	4.8B	0.85B	0.85B
Zero-shot classification					
Avg	77.0	83.4	82.5	82.7	**83.6**
ImageNet-1K	81.9	83.1	83.6	83.1	**83.9**
ImageNet-v2	74.7	77.4	77.6	77.6	**78.0**
ObjectNet	75.3	**84.4**	79.2	79.7	81.1
ImageNet-A	69.0	84.3	85.5	87.9	**89.7**
ImageNet-R	91.7	**95.7**	95.2	95.0	95.3
ImageNet-Sketch	69.8	**75.5**	74.1	72.8	73.6
Zero-shot retrieval					
Avg	72.5	76.7	–	78.9	**79.4**
COCO T→I	51.6	55.3	–	58.6	**59.4**
COCO I→T	69.0	71.4	–	75.9	**76.0**
Flickr30K T→I	77.2	85.0	–	85.7	**86.0**
Flickr30K I→T	92.0	95.2	–	95.3	**96.3**

24L is trained with a much smaller data budget, the amount of such hard visual data is relatively limited. Nevertheless, TuringViT-24L remains competitive on these benchmarks compared with other baselines, and we expect that further scaling the curated pretraining corpus with more viewpoint-diverse, rendition-style, and sketch-like samples can further improve performance. On retrieval benchmarks, our model demonstrates a clear advantage, which we attribute to the construction of accurate and highly aligned image-text pairs in our data pipeline.

3.2 Evaluating TuringViT as a Frozen Dense Visual Encoder

We further evaluate the transferability of TuringViT to dense prediction tasks, as summarized in Table 3. To this end, we freeze the pretrained visual encoder and use only the final-layer patch features for downstream dense evaluations. Each model is evaluated at its training resolution; for TuringViT, we use 1024 as the representative resolution, corresponding to the median resolution observed during dynamic-resolution training. We consider three representative dense tasks: monocular depth estimation on NYUv2 [48], semantic segmentation on ADE20K [49], and zero-shot object tracking on DAVIS-2017 [50].

Depth estimation. For depth estimation, we evaluate frozen TuringViT features on. We attach a lightweight DPT-style [51] depth head on top of the frozen visual features. Since NYUv2 is relatively small and large pretrained encoders are prone to overfitting under dense probing, we train the depth head for 20 epochs and report the best validation result.

Semantic segmentation. For semantic segmentation, we evaluate frozen TuringViT features on ADE20K. We train a lightweight segmentation head on top of the frozen patch features. The head follows a UPerNet-style [52] design to aggregate multi-scale contextual information and produce per-pixel semantic predictions.

Zero-shot tracking. For object tracking, we adopt a training-free zero-shot label propagation protocol on DAVIS-2017 [50]. Instead of training a task-specific head, we use the first-frame object mask as the reference and extract normalized patch features from each video frame with the frozen TuringViT encoder. For each patch in the current frame, we search for its nearest neighbors among the patch features from the most recent 7 frames and propagate the corresponding object labels to the current frame. We report the standard DAVIS $\mathcal{J}\&\mathcal{F}$ score as the evaluation metric.

Table 3: Evaluation of frozen dense visual representations. We freeze each visual encoder and evaluate its transferability on monocular depth estimation, semantic segmentation, and zero-shot video object tracking. ↓ indicates lower is better, while ↑ indicates higher is better. Best results are in bold.

Model	Input Res.	NYUv2 Depth RMSE ↓	ADE20K Seg. mIoU ↑	DAVIS-2017 Track. $\mathcal{J}\&\mathcal{F}$ ↑
MobileCLIP2-L	224	53.3	37.9	25.6
SigLIP2-L	384	51.6	42.3	32.4
TuringViT-18L	1024	53.7	43.2	47.5
TuringViT-24L	**1024**	**50.7**	**45.0**	**48.5**

3.3 Ablation Study

We conduct extensive ablation studies to verify the effectiveness of the key components in TuringViT-18L. Specifically, we analyze three aspects of our training recipe: the contribution of each data strategy component, including recaptioning, dynamic-resolution training, data augmentation, and data filtering; the effect of introducing video data and image replay in later-stage training; and the scaling behavior with respect to the amount of image-text pretraining data.

Data strategy: We conduct a cumulative ablation study to quantify the contribution of each component in the proposed robust pretraining recipe. All variants use TuringViT-18L and are trained for 10 epochs, while progressively enabling recaptioning, dynamic-resolution training, data augmentation, and data filtering. The results are summarized in Table 4. Starting from the 25M baseline, replacing noisy web captions with recaptioned annotations improves the average robustness score from 49.42 to 60.68, giving an absolute gain of +11.26%. This gain is broad across benchmarks and is especially large on ObjectNet (+17.09%), indicating that higher-quality text supervision helps the model learn

Table 4: Ablation study of the robust pretraining recipe for TuringViT-18L. The average score is computed over ImageNet-1K, ObjectNet, ImageNet-v2, ImageNet-A, ImageNet-R, and ImageNet-Sketch.

	Strategy component				Scale	Zero-shot robustness						
Variant	**Recap.**	**Dyn.**	**Aug.**	**Filt.**	**Data**	**Avg**	**IN-1K**	**ObjNet**	**IN-v2**	**IN-A**	**IN-R**	**IN-S**
Base	×	×	×	×	25M	49.4	56.6	36.6	49.2	45.6	66.8	41.8
+ Recaption	✓	×	×	×	25M	60.7	64.8	53.7	57.5	57.2	78.3	52.6
+ Dynamic res.	✓	✓	×	×	25M	73.5	77.3	62.3	69.9	75.0	90.8	65.7
+ Augmentation	✓	✓	✓	×	25M	74.4	78.7	63.1	71.1	75.4	91.2	66.8
Full data strategy	✓	✓	✓	✓	20M	**76.8**	**80.1**	**68.5**	**73.2**	**78.5**	**91.8**	**68.6**

Table 5: Effect of video data and image replay in later-stage training. All models use the same image training recipe; higher is better. Best results are in bold.

Training strategy	**Image zero-shot classification**						**Video classification / retrieval**		
	IN-1K	**Obj.**	**IN-v2**	**IN-A**	**IN-R**	**IN-S**	**K400**	**MSR-VTT T→V**	**MSR-VTT V→T**
Stage 3	**83.2**	75.7	77.3	84.4	94.6	72.5	64.3	43.0	39.3
Stage 4 (video only)	83.0	77.6	77.4	87.0	94.9	72.6	65.6	43.9	41.0
Stage 4	83.1	**79.7**	**77.6**	**87.9**	**95.0**	**72.8**	**66.3**	**44.5**	**41.9**

more object-centric and semantically aligned visual representations. Adding dynamic-resolution training further increases the average score to 73.49 (+12.81%), which is the largest single-step improvement in the ablation. The improvement is particularly strong on ImageNet-A (+17.81%), suggesting that resolution diversity substantially improves robustness to unusual object appearances, scales, and visual distributions. Data augmentation provides a smaller but consistent additional gain, increasing the average score from 73.49 to 74.40. This shows that stronger visual perturbations remain useful even after improving caption quality and resolution diversity. Finally, data filtering reduces the training set from 25M to 20M image-text pairs but further improves the average score to 76.78 (+2.38%). The largest gain from filtering again appears on ObjectNet (+5.34%), highlighting that removing noisy or weakly aligned samples is especially important for robustness-oriented transfer.

Training with Video: Table 5 compares different Stage 4 training strategies. The first row reports the Stage 3 checkpoint before any additional video-enhanced training, which serves as the baseline. Starting from this checkpoint, using video-only data in Stage 4 improves performance on both image and video benchmarks. For image zero-shot classification, it brings clear gains on robustness-oriented benchmarks such as ObjectNet (+1.9) and ImageNet-A (+2.6), suggesting that video data provides complementary supervision through viewpoint changes, motion patterns, object transformations, and more diverse real-world contexts. Meanwhile, video-only training also improves video classification and retrieval, increasing K400 [53] from 64.3 to 65.6, MSR-VTT [54] text-to-video retrieval from 43.0 to 43.9, and MSR-VTT video-to-text retrieval from 39.3 to 41.0. However, video-only Stage 4 training is not optimal: it slightly decreases ImageNet-1K performance and remains below the mixed image-video strategy on most benchmarks. When Stage 4 combines video data with image replay, the model achieves the best overall performance. It further improves ObjectNet to 79.7 and ImageNet-A to 87.9, while also achieving the best video results, with 66.3 on K400, 44.5 on MSR-VTT text-to-video retrieval, and 41.9 on MSR-VTT video-to-text retrieval. These results indicate that video data improves transformation-aware and out-of-distribution robustness, while mixing image data helps preserve the broad static-image alignment learned during large-scale image-text pretraining.

Data scaling: We conduct a controlled data-scaling study with TuringViT-18L under the same training recipe, including recaptioning, dynamic-resolution training, data augmentation, and filtering. All models are trained for 10 epochs on 20M, 100M, and 850M image-text pairs, respectively, and evaluated on six zero-shot classification benchmarks: ImageNet-1K, ObjectNet, ImageNet-v2, ImageNet-A, ImageNet-R, and ImageNet-Sketch. As shown in Figure 7, performance improves consistently as the amount of training data increases.The gains are especially pronounced on robustness-oriented benchmarks such as ObjectNet and ImageNet-A, indicating that data scaling primarily improves the

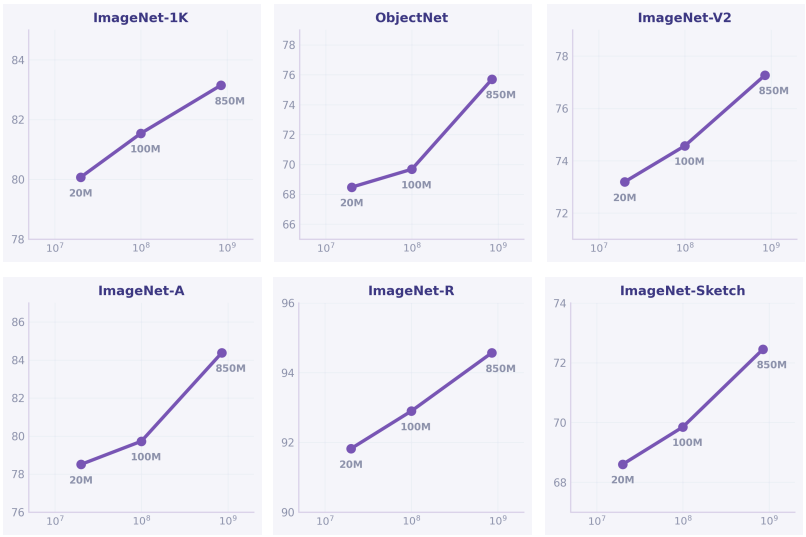


Figure 7: Training data scaling results of TuringViT-18L. We report zero-shot performance as the amount of pretraining data increases from 20M to 850M image-text pairs. All models are trained for 10 epochs using the same recipe, including recaptioning, dynamic-resolution training, data augmentation, and data filtering.

model's ability to handle distribution shifts. The curve shows no clear saturation up to 850M samples, suggesting that further scaling of curated training data may continue to bring additional improvements. This ablation demonstrates that for a CLIP-like model, increasing the number of training examples from 20M to 850M yields significant and systematic improvements across a wide range of robustness benchmarks. The gains follow a power law, and no saturation is observed at 850M, suggesting that further scaling of data (and model capacity) would continue to improve performance.

3.4 Evaluating TuringViT in Downstream VLMs

To further evaluate whether the advantages of TuringViT transfer to multimodal scenarios, we integrate it into downstream VLMs and compare it with representative open-source visual encoders, including SigLIP2 [8] and MobileCLIP2 [38]. Our goal is not to introduce a new VLM recipe, but to provide a controlled comparison of different ViT backbones under the same vision-language training pipeline. Therefore, all models adopt a standard ViT-MLP-LLM architecture, where the visual encoder is connected to the language model through an MLP projector. Most architectural details follow Qwen2.5-VL [10], such as token merger in projector and M-RoPE. For the language model, we use Qwen2.5-1.5B-Instruct [80] as the LLM backbone across all experiments.

Since this study focuses on evaluating the visual encoder, we only conduct supervised fine-tuning rather than full-scale VLM pretraining. The VLM training is divided into two stages. In Stage 1, we train the visual encoder and the projector while keeping the LLM frozen, using mainly captioning, OCR, knowledge, and VQA-style data to align visual features with the language model. In Stage 2, we unfreeze all parameters and further train the full model with a broader mixture of instruction data, including STEM, grounding, video, text-only, and GUI-related samples. All training data are collected from publicly available sources. For parts of the data with low quality or weak alignment, we apply data filtering and recaptioning to improve supervision quality.

For a fair comparison, we adapt SigLIP2 [8] and MobileCLIP2 [38] to the same dynamic-resolution VLM setting. Since these models are not originally designed with native dynamic-resolution support, we add 2D RoPE to their visual encoders and allow their visual backbones to be updated during Stage 1. This adaptation helps align their visual features with the LLM under the same dynamic-resolution input format, reducing the possibility that their performance is limited by implementation mismatch rather than backbone quality.

Table 6: Comparison of downstream VLM performance using different visual encoders under the same VLM training pipeline. Best results are marked in bold.

Benchmark	TuringViT-24L	SigLIP2-L	MobileCLIP2-L
Grounding & Counting			
RefCOCO avg [55, 56]	**90.4**	88.9	88.2
CountBench [57]	**81.1**	79.9	78.6
BLINK [58]	48.8	**49.1**	47.3
Text-rich Perception			
OCRBench V1 [59]	**798**	781	776
OCRBench V2 en [60]	**40.1**	36.2	35.4
OCRBench V2 cn [60]	**32.6**	32.0	31.0
ChartQA test [61]	79.4	79.6	**80.6**
TextVQA val [62]	**78.7**	77.1	76.2
DocVQA val [63]	**88.5**	87.3	87.8
AI2D test [64]	**77.1**	74.2	74.6
Multimodal Reasoning			
MMMU val [65]	**44.2**	41.8	41.6
MathVision mini [66]	**28.6**	24.3	24.7
MathVista mini [67]	60.0	62.0	**62.2**
DynaMath [68]	28.6	27.5	**28.7**
LogicVista [69]	**34.5**	30.2	30.9
General Visual Question Answering			
MMBench-EN [70]	76.3	**76.9**	73.1
MMBench-CN [70]	**74.4**	**74.4**	71.0
RealWorldQA [71]	**64.7**	63.7	62.5
MMStar [72]	**57.3**	55.8	56.2
MMVet [73]	**49.1**	47.0	43.4
HallusionBench [74]	**57.6**	57.4	54.8
MME-RealWorld-Lite [75]	**40.7**	40.3	36.5
SEEDBench [76]	**73.8**	73.5	72.6
Video Understanding			
MVBench [77]	**63.8**	63.1	63.7
MLVU [78]	**57.6**	55.6	53.5
LongVideoBench [79]	50.9	51.2	**51.4**

We evaluate the resulting VLMs on a diverse set of multimodal benchmarks that are sensitive to visual encoder quality, including grounding and counting, text-rich perception, multimodal reasoning, general VQA, and video understanding. These benchmarks cover spatial localization, OCR and document perception, diagram and mathematical reasoning, real-world visual question answering, hallucination-related evaluation, and temporal understanding. This suite allows us to examine whether the visual advantages of TuringViT transfer to downstream multimodal tasks.

As shown in Table 6, TuringViT achieves the best performance on most benchmarks under the same VLM setup, with particularly clear advantages in grounding and counting, text-rich perception, and general VQA. These categories are closely tied to the quality of the visual encoder: grounding and counting require accurate object-level and spatial representations, text-rich perception depends on fine-grained high-resolution details, and general VQA benefits from robust visual features across diverse real-world inputs. The strong results in these categories suggest that the benefits of TuringViT's architecture, curated supervision, and native dynamic-resolution training transfer effectively from visual pretraining to downstream VLM usage.

The advantage is less uniform on multimodal reasoning and video understanding. This is expected, as these tasks are not determined by the visual encoder alone. Multimodal reasoning benchmarks often require substantial language-side reasoning, symbolic manipulation, and instruction-following ability after the relevant visual evidence is extracted, making their final scores strongly dependent on

the LLM and SFT data mixture. Similarly, long-form video benchmarks place additional demands on temporal aggregation, memory, and sequence-level reasoning, while ViT pretraining is still primarily based on image and short-clip supervision. Nevertheless, TuringViT remains competitive in these settings and achieves leading results on several reasoning and video benchmarks. Overall, these results show that TuringViT is not merely an efficient replacement for softmax-based visual encoders, but a stronger VLM backbone whose gains are most pronounced on tasks where visual representation quality is the primary bottleneck.

4 Applications and Broader Impact

4.1 Unified Visual Foundation for VLA 2.0, Intelligent Cockpit, and Iron Robotics

TuringViT has been integrated into multiple XPeng AI systems, including VLA 2.0 for autonomous driving, multimodal systems for intelligent cockpit and driving-parking integration, and the foundation model of the XPeng Iron robot. These systems operate in different environments and require different forms of visual understanding, but all rely on a strong and efficient visual encoder as their perception foundation.

In **VLA 2.0**, the visual encoder needs to process multi-camera, multi-frame, and dynamic road-scene observations. It provides visual tokens that support spatial understanding, object-state estimation, and downstream policy decisions. Since the VLA system directly serves real-time vehicle-side decision making, the visual encoder must preserve strong perception capability while satisfying strict inference-efficiency requirements.

In **intelligent cockpit and driving-parking integrated systems**, the visual encoder supports multimodal understanding of both in-cabin and out-of-cabin environments. These tasks include scene recognition, OCR, navigation-related reasoning, parking and driving scene understanding, and interaction with user instructions and vehicle-control systems. In this setting, the visual representation must not only capture visual content accurately, but also align effectively with language, navigation, and control signals.

In the **XPeng Iron robot**, the visual encoder further provides the perceptual foundation for embodied intelligence. The robot needs to understand object categories, spatial relations, manipulation-relevant regions, occlusions, and dynamic changes in the environment, and then connect these visual signals with language instructions and action planning. Compared with vehicle-side scenarios, robotic tasks place stronger emphasis on close-range object interaction and action-relevant visual details, requiring fine-grained and transferable visual representations.

Across these systems, TuringViT serves as a shared visual foundation while allowing task-specific variants when needed. Different downstream systems can adjust model size, input resolution, deployment operators, or training data based on the same architecture and training methodology, without redesigning the visual encoder from scratch. This flexibility also extends to deeper VLM integration. Since TuringViT is designed to be VLM-native from the beginning, visual-side modules required by the final VLM can be introduced during ViT training, and a later generation-oriented alignment stage can be added to better match next-token prediction and instruction-following objectives. In XPeng's Turing VLM, such downstream-aware variants reduce integration mismatch, enter multimodal training more smoothly, and achieve further improvements on downstream VLM benchmarks.

4.2 Efficient Scaling and Broader Impact

The practical significance of TuringViT is not limited to any single downstream system. It is efficient to deploy within XPeng's Turing chip ecosystem, but its broader value is that it provides a scalable path for efficient ViT development. As VLM and VLA systems move toward higher-resolution perception, multi-image inputs, and video understanding, the length of visual sequences will continue to increase. If visual encoders continue to rely primarily on standard softmax attention, such scaling will quickly introduce substantial training and inference cost. By using a more efficient form of visual sequence modeling, TuringViT makes the continued scaling of high-resolution and dynamic-resolution ViTs more practical.

This scaling capability appears not only in the model architecture, but also in the data and training methodology. Our scaling-law analysis shows that TuringViT achieves clear and predictable gains as

the curated data scale increases, indicating that the model is still far from saturated. This suggests that TuringViT is not merely a specific model instance that works at the current data and model scale, but a framework that can continue to improve as data, model capacity, and task complexity grow.

More importantly, the core methodology of TuringViT does not depend on XPeng's internal hardware or private systems. Although TuringViT is friendly to deployment on Turing chips, its efficient visual architecture, data construction pipeline, and dynamic-resolution training strategy are general and transferable to other hardware platforms and application scenarios. For research and application teams across the industry, this means that they do not have to rely solely on a small number of large-scale pretrained general-purpose visual encoders. They can instead follow a similar methodology to train, customize, and deploy strong visual foundation models under controlled resource budgets.

Therefore, the application value of TuringViT is twofold. Internally, it supports XPeng's autonomous driving, cockpit, and robotics systems as a unified visual foundation. Externally, it provides the broader multimodal AI community with a reproducible, scalable, and deployable path for visual encoder development. **TuringViT turns SOTA-level ViTs from a capability accessible only to high-resource teams into a foundation module that more teams can build, customize, and benefit from**.

5 Conclusion and Future Work

This technical report presents **TuringViT**, a VLM-native visual foundation model designed to make strong customized Vision Transformers more accessible under practical training and deployment constraints. Instead of relying on fully quadratic softmax attention and fixed-resolution pretraining, TuringViT combines a linear-attention-dominant backbone, native dynamic-resolution training, and a supervision-rich image-video curation pipeline. This design enables efficient high-resolution visual encoding while preserving strong representation quality for downstream VLM and VLA applications.

Our experiments show that TuringViT achieves competitive or superior performance against strong open-source ViT baselines while using substantially less training data. The model also demonstrates favorable latency scaling under high-resolution and dynamic-resolution inputs, making it suitable for real-world multimodal systems where both visual fidelity and efficiency are critical. Beyond benchmark performance, the results suggest that careful architecture design and data curation can significantly improve the data efficiency of visual foundation model training.

We believe TuringViT provides a practical step toward democratizing the training and customization of SOTA-level visual encoders. In future work, we plan to further scale the curated image-video data engine, strengthen temporal and embodied visual understanding, and explore broader integration with VLM/VLA systems in autonomous driving, robotics, and intelligent interaction scenarios. We hope this report encourages continued research on efficient, VLM-native visual foundation models and helps visual encoders play a larger role in next-generation multimodal applications.

6 Contributors

We sincerely thank every member of the team for their dedication and valuable contributions. This work reflects our ongoing efforts in advancing VLM/VLA-related applications, and we hope that TuringViT will play an increasingly important role in this direction. In addition, our VISTA data curation pipeline is designed to lower the barrier to training visual foundation models.

Advisors: Hang Zhang, Xianming Liu

Project Lead: Qiman Wu

Contributors: Hanlin Chen[*] , Lyujie Chen[*] , Rui Xin[*] , Jianlei Zheng , Mingyuan Wang , Jiahui Hu , Da Zhu , Yuecheng Ma , Yuhua Wei , Yizhao Wang , Hua Zhou , Yuheng Zhang , Anhua Liu , Shaman Tang , Yue He , Pengfei Diao , Shuang Su , Haotong Xin[†] , Weichao Huang

[*]Core contribution. The first three authors are listed in alphabetical order.

[†]Research Intern at XPENG.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, et al. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

[2] Zhe Chen, Wenhai Wang, Enze Xie, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *CoRR*, abs/2312.14238, 2023. URL https://arxiv.org/abs/2312.14238.

[3] Dong Guo, Faming Wu, Feida Zhu, et al. Seed1.5-vl technical report. 2025. URL https://arxiv.org/abs/2505.07062.

[4] Physical Intelligence, Bo Ai, Ali Amin, et al. $\pi_{0.7}$: a steerable generalist robotic foundation model with emergent capabilities. 2026. URL https://arxiv.org/abs/2604.15483.

[5] En Yu, Haoran Lv, Jianjian Sun, et al. Dm0: An embodied-native vision-language-action model towards physical ai. 2026. URL https://arxiv.org/abs/2602.14974.

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[7] Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, et al. Mobileclip: Fast image-text models through multi-modal reinforced training. 2024. URL https://arxiv.org/abs/2311.17049.

[8] Michael Tschannen, Alexey Gritsenko, Xiao Wang, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. 2025. URL https://arxiv.org/abs/2502.14786.

[9] Peng Wang, Shuai Bai, Sinan Tan, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[10] Shuai Bai, Keqin Chen, Xuejing Liu, et al. Qwen2.5-vl technical report. 2025. URL https://arxiv.org/abs/2502.13923.

[11] Yi Wang, Kunchang Li, Yizhuo Li, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[13] Ze Liu, Yutong Lin, Yue Cao, et al. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[14] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, et al. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020.

[15] Christoph Schuhmann, Romain Beaumont, Cade Vencu, et al. LAION-5b: An open large-scale dataset for training next generation image-text models. 2022.

[16] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, et al. Datacomp: In search of the next generation of multimodal datasets. 2023. URL https://arxiv.org/abs/2304.14108.

[17] Alex Fang, Albin Madappally Jose, Amit Jain, et al. Data filtering networks. In *International Conference on Learning Representations*, volume 2024, pages 36221–36237, 2024.

[18] Lin Chen, Jinsong Li, Xiaoyi Dong, et al. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pages 370–387. Springer, 2024.

[19] Haotian Liu, Chunyuan Li, Yuheng Li, et al. Llavanext: Improved reasoning, ocr, and world knowledge, 2024.

[20] Zhe Chen, Jiannan Wu, Wenhai Wang, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024.

[21] Mostafa Dehghani, Basil Mustafa, Josip Djolonga, et al. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *Advances in Neural Information Processing Systems*, 36:2252–2274, 2023.

[22] Beichen Zhang, Pan Zhang, Xiaoyi Dong, et al. Long-CLIP: Unlocking the long-text capability of CLIP. *arXiv preprint arXiv:2403.15378*, 2024.

[23] Ivona Najdenkoska, Mohammad Mahdi Derakhshani, Yuki M Asano, et al. TULIP: Token-length upgraded CLIP. *arXiv preprint arXiv:2410.10034*, 2024.

[24] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, et al. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3531–3539, 2021.

[25] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[26] Songlin Yang, Bailin Wang, Yikang Shen, et al. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

[27] Sinong Wang, Belinda Z Li, Madian Khabsa, et al. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[28] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, et al. Dinov3. 2025. URL https://arxiv.org/abs/2508.10104.

[29] Jianlin Su, Murtadha Ahmed, Yu Lu, et al. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[30] Byeongho Heo, Song Park, Dongyoon Han, et al. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pages 289–305. Springer, 2024.

[31] Biao Zhang, Rico Sennrich. Root mean square layer normalization. *Advances in neural information processing systems*, 32, 2019.

[32] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

[33] Yiping Wang, Yifang Chen, Wendan Yan, et al. Cliploss and norm-based data selection methods for multimodal contrastive learning. *Advances in Neural Information Processing Systems*, 37:15028–15069, 2024.

[34] Yuxin Fang, Quan Sun, Xinggang Wang, et al. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024.

[35] Quan Sun, Yuxin Fang, Ledell Wu, et al. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[36] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, et al. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

[37] Zilong Huang, Qinghao Ye, Bingyi Kang, et al. Classification done right for vision-language pre-training. *Advances in Neural Information Processing Systems*, 37:96483–96504, 2024.

[38] Fartash Faghri, Pavan Kumar Anasosalu Vasu, Cem Koc, et al. Mobileclip2: Improving multi-modal reinforced training. *arXiv preprint arXiv:2508.20691*, 2025.

[39] Dong Guo, Faming Wu, Feida Zhu, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

[40] Jia Deng, Wei Dong, Richard Socher, et al. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[41] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, et al. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[42] Andrei Barbu, David Mayo, Julian Alverio, et al. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.

[43] Dan Hendrycks, Kevin Zhao, Steven Basart, et al. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271, 2021.

[44] Dan Hendrycks, Steven Basart, Norman Mu, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021.

[45] Haohan Wang, Songwei Ge, Zachary Lipton, et al. Learning robust global representations by penalizing local predictive power. *Advances in neural information processing systems*, 32, 2019.

[46] Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. Microsoft coco: Common objects in context. 2015. URL https://arxiv.org/abs/1405.0312.

[47] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, et al. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. 2016. URL https://arxiv.org/abs/1505.04870.

[48] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, et al. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[49] Bolei Zhou, Hang Zhao, Xavier Puig, et al. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[50] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, et al. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[51] René Ranftl, Alexey Bochkovskiy, Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021.

[52] Tete Xiao, Yingcheng Liu, Bolei Zhou, et al. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018.

[53] Will Kay, Joao Carreira, Karen Simonyan, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[54] Jun Xu, Tao Mei, Ting Yao, et al. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016.

[55] Licheng Yu, Patrick Poirson, Shan Yang, et al. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer, 2016.

[56] Junhua Mao, Jonathan Huang, Alexander Toshev, et al. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2016.

[57] Roni Paiss, Ariel Ephrat, Omer Tov, et al. Teaching CLIP to count to ten. *arXiv preprint arXiv:2302.12066*, 2023.

[58] Xingyu Fu, Yushi Hu, Bangzheng Li, et al. BLINK: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024.

[59] Yuliang Liu, Zhang Li, Mingxin Huang, et al. OCRBench: On the hidden mystery of OCR in large multimodal models. *arXiv preprint arXiv:2305.07895*, 2023.

[60] Ling Fu, Biao Yang, Zhebin Kuang, et al. OCRBench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *arXiv preprint arXiv:2501.00321*, 2024.

[61] Ahmed Masry, Do Xuan Long, Jia Qing Tan, et al. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, 2022.

[62] Amanpreet Singh, Vivek Natarajan, Meet Shah, et al. Towards VQA models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019.

[63] Minesh Mathew, Dimosthenis Karatzas, C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2200–2209, 2021.

[64] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, et al. A diagram is worth a dozen images. In *European Conference on Computer Vision*, pages 235–251. Springer, 2016.

[65] Xiang Yue, Yuansheng Ni, Kai Zhang, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. *arXiv preprint arXiv:2311.16502*, 2024.

[66] Ke Wang, Junting Pan, Weikang Shi, et al. Measuring multimodal mathematical reasoning with MATH-Vision dataset. *arXiv preprint arXiv:2402.14804*, 2024.

[67] Pan Lu, Hritik Bansal, Tony Xia, et al. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[68] Chengke Zou, Xingang Guo, Rui Yang, et al. DynaMath: A dynamic visual benchmark for evaluating mathematical reasoning robustness of vision language models. *arXiv preprint arXiv:2411.00836*, 2024.

[69] Yijia Xiao, Edward Sun, Tianyu Liu, et al. LogicVista: Multimodal LLM logical reasoning benchmark in visual contexts. *arXiv preprint arXiv:2407.04973*, 2024.

[70] Yuan Liu, Haodong Duan, Yuanhan Zhang, et al. MMBench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023.

[71] xAI. Grok-1.5 vision preview. https://x.ai/news/grok-1.5v, 2024.

[72] Lin Chen, Jinsong Li, Xiaoyi Dong, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024.

[73] Weihao Yu, Zhengyuan Yang, Linjie Li, et al. MM-Vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

[74] Tianrui Guan, Fuxiao Liu, Xiyang Wu, et al. HallusionBench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. *arXiv preprint arXiv:2310.14566*, 2023.

[75] Yi-Fan Zhang, Huanyu Zhang, Haochen Tian, et al. MME-RealWorld: Could your multimodal LLM challenge high-resolution real-world scenarios that are difficult for humans? *arXiv preprint arXiv:2408.13257*, 2024.

[76] Bohao Li, Rui Wang, Guangzhi Wang, et al. SEED-Bench: Benchmarking multimodal LLMs with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023.

[77] Kunchang Li, Yali Wang, Yinan He, et al. MVBench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*, 2023.

[78] Junjie Zhou, Yan Shu, Bo Zhao, et al. MLVU: Benchmarking multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

[79] Haoning Wu, Dongxu Li, Bei Chen, et al. LongVideoBench: A benchmark for long-context interleaved video-language understanding. *arXiv preprint arXiv:2407.15754*, 2024.

[80] Qwen, An Yang, Baosong Yang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.