DiffusionVL: Translating Any Autoregressive Models into Diffusion Vision Language Models

Lunbin Zeng[1*], Jingfeng Yao[1*], Bencheng Liao[1], Hongyuan Tao[1], Wenyu Liu[1], and Xinggang Wang[1✉]

[1]Huazhong University of Science and Technology, Wuhan, China

Abstract. Diffusion-based decoding has recently emerged as an appealing alternative to autoregressive (AR) generation, offering the potential to update multiple tokens in parallel and reduce latency. However, diffusion vision language models (dVLMs) still lag significantly behind mainstream autoregressive vision language models. This is due to the scarcity and weaker performance of base diffusion language models (dLLMs) compared with their autoregressive counterparts. This raises a natural question: Can we build high-performing dVLMs directly from existing powerful AR models, without relying on dLLMs? We propose **DiffusionVL**, a family of dVLMs obtained by translating pretrained AR models into the diffusion paradigm via an efficient diffusion finetuning procedure that changes the training objective and decoding process while keeping the backbone architecture intact. Through an efficient diffusion finetuning strategy, we successfully adapt AR pretrained models into the diffusion paradigm. This approach yields two key observations: (1) The paradigm shift from AR-based multimodal models to diffusion is remarkably effective. (2) Direct conversion of an AR language model to a dVLM is also feasible, achieving performance comparable to that of the same AR model finetuned with standard autoregressive visual instruction tuning. To enable practical open-ended generation, we further integrate block decoding, which supports arbitrary-length outputs and KV-cache reuse for faster inference. Our experiments demonstrate that despite training with **less than 5% of the data** required by prior methods, DiffusionVL achieves a comprehensive performance improvement, with **a 34.4% gain** on the MMMU-Pro (vision) benchmark and **37.5% gain** on the MME (Cog.) benchmark, alongside a **2× inference speedup**. The model and code are released at `https://github.com/hustvl/DiffusionVL`.

Keywords: Vision Language Model · Diffusion Language Model · Autoregressive Model

1 Introduction

Vision language models (VLMs) [5, 9, 23, 32] have achieved significant success in multimodal understanding. Most of them are autoregressive (AR) models

* Equal contribution; ✉ Corresponding author: xgwang@hust.edu.cn.

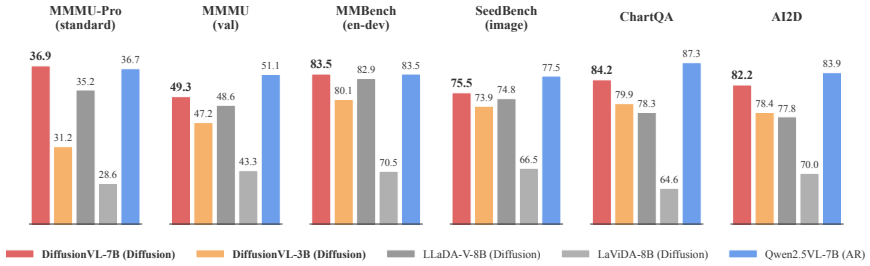


Fig. 1: Performance Comparison. Our DiffusionVL achieves state-of-the-art (SOTA) performance among diffusion vision language models including [20, 44] and competitive performance with Qwen2.5VL [31].

with the next-token prediction (NTP) paradigm. A key limitation of the NTP paradigm is its inability to inherently support parallel inference, which limits its applicability in real-time scenarios.

Recently, the diffusion paradigm [20, 44, 45] has emerged as a promising alternative, offering more efficient parallel decoding potential than the NTP paradigm. However, existing diffusion VLMs (dVLMs) still lag behind advanced AR-VLMs on standard multimodal benchmarks (Fig. 1). A key bottleneck is that today's diffusion language models (dLLMs) are generally less capable and less mature than strong AR language models (AR-LMs), which limits the ceiling of dVLMs built on top of dLLMs. For instance, LLaDA-8B lags behind Qwen2.5-7B by 42.0% on the code task HumanEval [8]. Influenced by the conventional visual instruction tuning, existing attempts still assume that building dVLMs must rely on cross-modal training from a dLLM of the same paradigm. In fact, dVLMs and AR-VLMs are structurally identical; the only difference lies in their attention patterns and training/inference behaviors. Since the architecture does not dictate the paradigm, building dVLMs from dLLMs is not the only option. Given these challenges and findings, a compelling question emerges: *Can we build high-performing dVLMs* directly *from existing powerful AR models, without relying on dLLMs?*

To answer this question, we explore translating any pretrained autoregressive models into diffusion vision language models (see Fig. 2). We propose DiffusionVL, whose core technical contribution is demonstrating that a simple diffusion finetuning approach can achieve this translation.

Specifically, we convert the next-token prediction paradigm of the original AR model into a diffusion paradigm, which we refer to as "diffusion finetuning". Its key advantage is that it enables the construction of DiffusionVL from any AR model without any architectural modifications. For AR-VLMs, since these models are already vision-language aligned, we directly apply full-parameter diffusion finetuning to convert AR-VLMs to dVLMs. For AR-LMs, we build the vision-language model in two stages. We first train only the connector during the pretraining stage to align the vision and text spaces, with the autoregressive paradigm ensuring training stability [23]. We then conduct diffusion finetuning

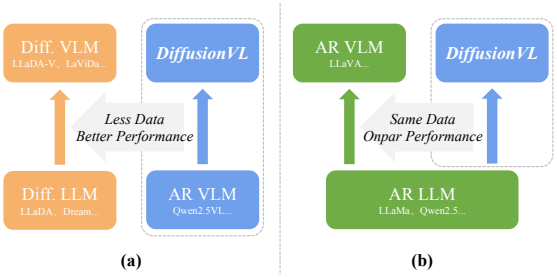


Fig. 2: (a) Translating AR-VLMs to dVLMs (paradigm shift only). (b): Translating AR-LMs to dVLMs (modality shift and paradigm shift). Any autoregressive models with different modalities can be effectively translated to diffusion VLMs.

in the second stage to complete the paradigm conversion. Furthermore, we adopt the block decoding strategy consistent with [2], which not only supports response generation of arbitrary lengths but also effectively reuses KV-cache for enhanced efficiency. Building on these technical designs, we introduce the DiffusionVL family, which can be translated from AR models of any scale and modality, while maintaining efficient inference speeds.

Extensive experiments validate the performance and efficiency of DiffusionVL. By seamlessly translating AR-VLMs into DiffusionVL, our method achieves state-of-the-art (SOTA) performance among current dVLMs. Remarkably, this is accomplished using less than 5% of the training data required by prior approaches, thereby significantly narrowing the performance gap with advanced, data-intensive AR-VLMs. Specifically, DiffusionVL achieves a comprehensive performance improvement, with a 34.4% gain on the MMMU-Pro (vision) benchmark and 37.5% gain on the MME (Cog.) benchmark. When translating AR-LMs to DiffusionVL, we have drawn the following conclusions through detailed controlled experiments: under the same training data and configuration, the DiffusionVL finetuned from AR-LM consistently outperforms its counterpart finetuned from dLLM on downstream multimodal benchmarks. Furthermore, even when compared to AR-VLMs with autoregressive finetuning of the same paradigm, our DiffusionVL can still achieve comparable performance on downstream benchmarks. For inference, equipped with the block decoding strategy, DiffusionVL achieves a 2.0× speedup over previous dVLMs, demonstrating the great potential of our method. In summary, our contributions are threefold as follows:

- We validate the effectiveness and feasibility of translating any pretrained autoregressive models into diffusion vision language models, providing an efficient and low-cost approach for developing high-performance diffusion vision language models.
- We incorporate a block decoding strategy that enables arbitrary-length generation and efficient KV-cache reuse, addressing two key limitations of existing diffusion vision language models.
- Extensive experimental results demonstrate that our method yields a state-of-the-art DiffusionVL, which not only narrows the performance gap with advanced autoregressive vision language models but also achieves a 2.0× speedup compared to existing diffusion vision language models.

2 Related Work

2.1 Masked Diffusion Models

Diffusion models have achieved great success in vision generation and other computer vision tasks [12, 21, 22, 30, 41, 42, 52, 53]. At the same time, recent work has begun to explore the potential of diffusion in text generation. Prior work [4,14,26] on masked discrete diffusion models (MDMs) has already validated the effectiveness of this paradigm in small-scale text pretraining scenarios. These early studies demonstrated that MDMs can achieve perplexity levels comparable to those of AR-LMs while enabling parallel inference. Building on this basis, recent advanced models such as LLaDA [29] and Dream [43] have further validated the effectiveness and scalability of MDMs in modern large language model (LLM) settings.

In the visual domain, diffusion vision language models (dVLMs) are typically built from a pre-trained diffusion language model and converted through visual instruction finetuning. Dimple [45] designs a novel training paradigm that combines an autoregressive phase with a subsequent diffusion phase. LLaDA-V [44] directly explores the effectiveness of large-scale visual finetuning under the diffusion paradigm based on LLaDA. LaViDa [20] further introduces complementary masking and prefix cache to improve training and inference efficiency. These models follow the LLaVA [23] paradigm and have explored the potential of the mask diffusion paradigm in the visual and multimodal domains. However, they are limited by their inability to perform variable-length generation and to reuse the KV-cache efficiently. A notable performance gap also remains compared with advanced AR-VLMs. Beyond multimodal understanding, recent work extends the masked diffusion paradigm to unified frameworks: MMaDA [40] and LaViDa-O [19] demonstrate that the masked diffusion paradigm, when combined with finetuning and reinforcement learning, can support both image and text generation and editing within a single model.

2.2 Interpolation between AR and Diffusion

Due to limitations of the MDMs in KV-cache reuse and arbitrary-length generation, several works have explored how to combine the autoregressive (AR) and masked diffusion paradigms for text generation. SSD-LM [15] introduced a block formulation of Gaussian text diffusion. AR-Diffusion [38] further extended SSD-LM by incorporating a left-to-right noise schedule. BD3-LM [2] interpolates between discrete masked diffusion and autoregressive models by using inner-block diffusion and inter-block autoregressive decoding, and has served as a foundation for follow-up research. However, these early block diffusion methods are limited to small-scale text pre-training; their scalability to large language models and extension to the vision-language domain remained unproven.

Recently, some works have attempted to use autoregressive models as initialization and convert them into block diffusion paradigm. To scale this block diffusion paradigm to large text models, SDAR [10], Fast-dLLM-V2 [36] verified

that finetuning from AR models can build efficient block diffusion language models. SDLM [24] further extended the block diffusion paradigm to next-sequence prediction. More recently, LLaDA 2.0 [6] scales block diffusion to 100B parameters and optimizes inference speed, achieving performance competitive with advanced open-source AR-LMs while delivering significantly faster inference. However, such AR-diffusion interpolation remains underexplored in vision language models; our work addresses this gap by translating any autoregressive models into dVLMs through a simple diffusion finetuning. Concurrent to our work, A2D-VL [3] also adapts pretrained AR-VLMs for parallel diffusion decoding with block-size and noise-level annealing, whereas DiffusionVL obtains strong dVLMs through a simpler diffusion finetuning recipe and further extends the conversion to AR-LMs and different model families.

3 Method

3.1 Preliminaries

In the autoregressive paradigm, text generation is modeled as next-token prediction. Given an input sequence $\{x^1, \ldots, x^L\}$, the training objective is to minimize the cross-entropy loss.

$$\mathcal{L}_{\mathrm{AR}}(x;\theta) = -\mathbb{E}_x \left[\sum_{i=1}^{L} \log P_\theta(x^i \mid x^{<i}) \right], \tag{1}$$

where the model $P_\theta(\cdot|x^{<i})$ aims to maximize the conditional probability of the current position by using the preceding context $x^{<i} = x^1, \ldots, x^{i-1}$.

By contrast, masked diffusion models can be subdivided into two paradigms: full diffusion and block diffusion. The full diffusion paradigm adds and removes noise simultaneously throughout the entire sequence. For a time $t \in (0, 1)$, the input sequence is masked with a probability of t, yielding a noisy sequence x_t. The model $P_\theta(\cdot|x_t)$ is trained to minimize the expected masked position prediction loss.

$$\mathcal{L}_{\mathrm{DM}}(x;\theta) = -\mathbb{E}_{t,x_0,x_t} \left[\frac{1}{t} \sum_{i=1}^{L} \mathbf{1}\big[x_t^i = [\mathrm{MASK}]\big] \log P_\theta(x_0^i \mid x_t) \right], \tag{2}$$

where $t \sim \mathcal{U}(0, 1)$ and the loss is only calculated for masked positions.

The block diffusion paradigm divides the entire sequence into several blocks of equal size and performs block-wise noise addition and denoising within each block. The loss is also only calculated for masked positions.

$$\mathcal{L}_{\mathrm{BDM}}(x;\theta) = -\mathbb{E}_{t,x_0,x_t} \left[\sum_{i=1}^{L} \mathbf{1}\big[x_t^i = [\mathrm{MASK}]\big]\, \alpha\, \log P_\theta(x_0^i \mid x_{<i}, x_{D(i)}) \right], \tag{3}$$

where $x_{D(i)}$ denotes all contexts in the block containing position i, $x_{<i}$ are clean contexts from earlier blocks, α represents the loss scale which is calculated by $\frac{\alpha_t'}{1-\alpha_t}$, and α_t' is the instantaneous rate of change of α_t in continuous time.

3.2 Modalities and Paradigms

Architectures and paradigms are largely decoupled: A single Transformer architecture [34] can yield diverse models by applying different paradigms. Based on this observation, our model retains the same network architecture as existing autoregressive models and is initialized from them; only the training/inference behavior and the attention pattern differ. Accordingly, we design different diffusion finetuning pipelines for AR-VLMs and AR-LMs.

First, we describe the process of finetuning an AR-VLM into a dVLM as a paradigm shift. We demonstrate that this direct paradigm shift is an efficient way to build dVLMs, which aligns with the findings of [10,36] in the text domain. Since our model has already been vision-language aligned, we directly finetune the entire AR-VLM into a dVLM in an end-to-end manner using Eq. (3).

Furthermore, we extend the approach to build a dVLM based on an AR-LM, and describe it as a process of modality shift and paradigm shift. We adopt the traditional two-stage training approach similar to LLaVA [23]. In the pretraining stage, we only train the connector to align the vision embedding space and text embedding space. Since this connector is randomly initialized to align vision and text embeddings, we use the standard autoregressive objective Eq. (1) to stabilize the modality shift process. In the finetuning stage, all components are jointly trained end-to-end using the diffusion finetuning strategy Eq. (3) to achieve modality shift and paradigm shift at the same time.

3.3 Diffusion Finetuning

In this section, we elaborate on the training framework of our diffusion finetuning, as illustrated in Fig. 3.

The image is processed through the vision encoder and projector, and the resulting vision embeddings are concatenated with text embeddings. Each sequence is padded with `<EOS>` tokens to a length divisible by the block size D and split into $B = L/D$ non-overlapping blocks $\{\mathbf{x}_{(1)}, \ldots, \mathbf{x}_{(B)}\}$. Unlike the sequence-level noise in prior dVLMs [20, 44, 45], we adopt a block-wise noise schedule: For each block b containing response or padding tokens, a noise level $t_b \sim \mathcal{U}(0, 1)$ is sampled and each token x^i within it is independently masked as

$$\tilde{x}^i = \begin{cases} \texttt{[MASK]}, & \text{with probability } t_b, \\ x^i, & \text{with probability } 1 - t_b, \end{cases} \qquad \forall\, i \in \text{block } b, \tag{4}$$

while blocks containing image or prompt tokens remain unmasked ($t_b = 0$). The noised sequence $\tilde{\mathbf{x}}$ and clean sequence $\mathbf{x}$ are concatenated along the sequence dimension, with an attention mask $\mathbf{M}$ enforcing intra-block bidirectional and inter-block causal attention. For position i in noised block b, $\mathbf{M}_{ij}$ indicates whether i may attend to j (1) or not (0):

$$\mathbf{M}_{ij} = \begin{cases} 1, & \text{if } j \text{ is in the same noised block as } i, \\ 1, & \text{if } j \text{ is in a clean block } b' \text{ with } b' < b, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

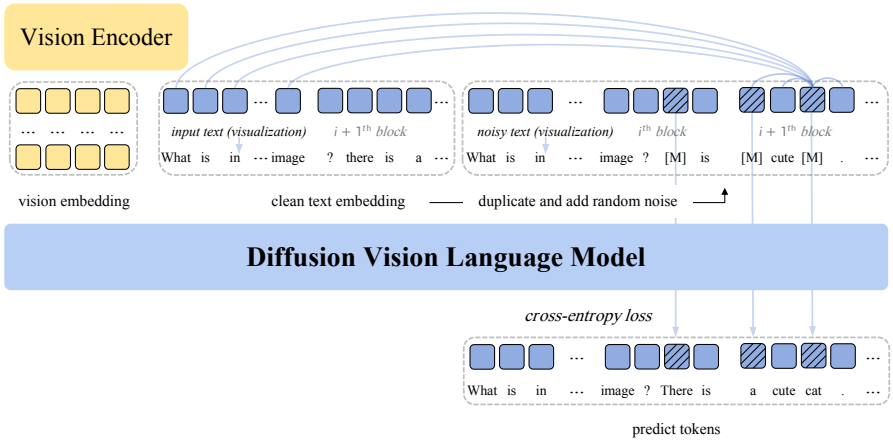


Fig. 3: The diffusion finetuning framework of DiffusionVL. The noisy sequence is concatenated with the original clean sequence along the sequence dimension. Each noisy block attends to all preceding clean blocks (inter-block causal) and all positions within its own block (intra-block bidirectional). The cross-entropy loss is computed only at masked positions.

The model is trained to minimize the cross-entropy loss at masked positions using Eq. (3).

3.4 Diffusion Inference

During inference, our model combines inter-block autoregressive generation with intra-block parallel diffusion decoding, naturally supporting KV-cache reuse and arbitrary-length generation. The full procedure is summarized in Algorithm 1.

Given an input image and a text prompt, we first encode them using the vision encoder, projector, and text embedding layer, and concatenate the resulting embeddings to initialize the context cache $\mathbf{C}_0$. For each subsequent block m, we initialize all D positions as mask tokens and perform up to S iterative denoising steps.

At each step s, let $\mathcal{M}^{(s-1)}$ denote the set of indices still masked in $\mathbf{x}_m^{(s-1)}$. For each $i \in \mathcal{M}^{(s-1)}$, we sample the prediction conditioned on the current block state and preceding context $\mathbf{C}_{m-1}$ via KV-cache reuse (following the hybrid attention pattern in Eq. (5)), and record the confidence $c_{m,i}^{(s)}$. A subset $\mathcal{U}^{(s)}$ is then chosen and unmasked based on $c_{m,i}^{(s)}$.

We adopt two remasking strategies following [10] to determine $\mathcal{U}^{(s)}$: the static low-confidence remasking strategy unmasks the k highest-confidence positions per step; the dynamic low-confidence remasking strategy further augments this set by additionally unmasking all positions whose confidence exceeds a threshold τ, enabling faster decoding on simpler content. Once a block is fully denoised, it

Algorithm 1: Diffusion Inference of DiffusionVL.

Input: image $\mathbf{I}$, prompt $\mathbf{T}$, max length L, block size D, steps S, threshold τ
Output: generated token sequence $\mathbf{Y}$

$\mathbf{C}_0 \leftarrow \texttt{Concat}\big(\texttt{Proj}(\texttt{VEnc}(\mathbf{I})),\ \texttt{TEmbed}(\mathbf{T})\big),\ \mathbf{Y} \leftarrow \varnothing$ `// init`
for $m \leftarrow 1$ **to** $\lfloor L/D \rfloor$ **do**
  $\mathbf{x}_m^{(0)} \leftarrow [\texttt{MASK}]^D$
  for $s \leftarrow 1$ **to** S **do**
    $\hat{x}_{m,i}^{(s)} \sim P_\theta(\cdot \mid \mathbf{x}_m^{(s-1)}, \mathbf{C}_{m-1})$,
    $c_{m,i}^{(s)} \leftarrow P_\theta(\hat{x}_{m,i}^{(s)} \mid \mathbf{x}_m^{(s-1)}, \mathbf{C}_{m-1})$
    for all $i \in \mathcal{M}^{(s-1)}$
    $k \leftarrow \min(\lceil D/S \rceil, |\mathcal{M}^{(s-1)}|)$;
    $\mathcal{U}^{(s)} \leftarrow \text{top-k}(\{c_{m,i}^{(s)}\}, k)$ `// static`
    if dynamic **then** $\mathcal{U}^{(s)} \leftarrow \mathcal{U}^{(s)} \cup \{i \in \mathcal{M}^{(s-1)} \mid c_{m,i}^{(s)} > \tau\}$
    $x_{m,i}^{(s)} \leftarrow \hat{x}_{m,i}^{(s)}$ if $i \in \mathcal{U}^{(s)}$, else `[MASK]` `// unmask`
  end
  $\mathbf{C}_m \leftarrow \mathbf{C}_{m-1} \cup \mathbf{x}_m^{(S)};\ \mathbf{Y} \leftarrow \mathbf{Y} \cup \mathbf{x}_m^{(S)}$
  if $\langle \text{EOS} \rangle \in \mathbf{x}_m^{(S)}$ **then break**
end
return $\mathbf{Y}$

is appended to the context cache, and generation proceeds to block $m+1$ until an $\langle \text{EOS} \rangle$ token is produced.

4 Experiment

4.1 Implementation Details

Model architecture. For building dVLMs from AR-VLMs, we use Qwen2.5VL-3B-Instruct and Qwen2.5VL-7B-Instruct [5] as our base models. For finetuning from AR-LMs and dLLMs, we select Qwen2.5-7B-Instruct [31] and LLaDA-8B-Instruct [29], respectively. The vision encoder is SigLip2-400M [33]. The projector is a randomly initialized two-layer MLP.
Training data. For building dVLMs from AR-VLMs, we use only the 738K instruction-following samples from LLaVA-Next [17] for end-to-end finetuning. For building dVLMs and AR-VLMs from AR-LMs and for building dVLMs from dLLMs, we adopt a two-stage data setup: the 580K-sample pretraining dataset from LLaVA-Pretrain [23] in the pretraining stage, and the 738K instruction-following samples from LLaVA-Next in the finetuning stage.
Hyperparameters. We use AdamW with a cosine decay scheduler. The pretraining stage uses a learning rate of 1×10^{-3}; the finetuning stage uses 1×10^{-5} for the LLM and projector and 2×10^{-6} for the vision encoder. The default training block size is 8. During inference, we adopt static low-confidence remasking with the number of denoising steps equal to the block size.

Model	Size	Type	Samples	MMBench [en-dev]	MMMU [val]	MMMU-Pro [std.]	MMMU-Pro [vision]	MMStar [test]	MME [cog.]	MME [perp.]
AutoRegressive Vision Language Models										
LLaVA	7B	AR	-	38.7	-	-	-	-	-	809
LLaVA-1.5	7B	AR	-	64.3	-	-	-	-	-	1510
Cambrian-1	8B	AR	-	75.9	42.7	-	-	-	-	1547
LLaVA-OV	7B	AR	7.8M	80.8	48.8	-	-	61.7	418	1580
Qwen2.5VL	3B	AR	>9M	79.1	46.8	31.2	22.1	55.9	620	1533
	7B	AR	>9M	**83.5**	**51.1**	**36.7**	**33.4**	**63.9**	**646**	**1680**
Diffusion Vision Language Models										
LaViDa-L	8B	Diff.	1.6M	70.5	43.3	28.6	-	-	341	1365
Dimple	7B	Diff.	1.3M	-	45.2	-	-	-	432	1514
LLaDA-V	8B	Diff.	16.5M	82.9	48.6	35.2	18.6	60.1	491	1507
DiffusionVL	3B	Diff.	738K	80.1	47.2	31.2	20.2	55.9	594	**1539**
	7B	Diff.	738K	**83.5**	**49.3**	**36.9**	**25.0**	**63.2**	**675**	1519

Table 1: Benchmark Performance Comparison (Part 1). The top-2 results are highlighted separately for AR and Diffusion models. Best results are in **bold**, second-best are underlined.

Evaluation benchmarks. We evaluate on a diverse set of vision-language benchmarks using the LMMS-Eval [48] library with its default prompts:
- *General knowledge:* MMMU [46], MMMU-Pro [47], MMStar [7], MME [13], SeedBench [18], MMBench [25], RealworldQA [39].
- *Chart and document understanding:* AI2D [16], ChartQA [28].
- *Mathematical reasoning:* MathVista [27], MathVerse [50], MathVision [1].
- *Detail image captioning:* DetailCaps [11].
- *Multi-image understanding:* MuirBench [35].

4.2 Efficient dVLM Construction from AR-VLMs

Tab. 1 and Tab. 2 present the downstream multimodal benchmark results of DiffusionVL-3B and DiffusionVL-7B, which are derived from diffusion finetuning of Qwen2.5VL-3B-Instruct and Qwen2.5VL-7B-Instruct. We compare our models' results with representative AR-VLMs and open-source dVLMs.

As shown in Tab. 1 and Tab. 2, DiffusionVL-7B achieves comprehensive performance that surpasses existing open-source dVLMs LaViDa-L [20], Dimple [45], and LLaDA-V [44] on multimodal benchmarks. This is achieved despite finetuning with only 738K samples, i.e., less than 5% of the data used for LLaDA-V, demonstrating strong vision-language capability and high training efficiency. Furthermore, leveraging the strong pretrained foundation of AR-VLMs, DiffusionVL-3B even outperforms the larger LaViDa-L-8B and Dimple-7B with less training data, further validating the effectiveness of our proposed method. Moreover, our DiffusionVL significantly narrows the gap between existing dVLMs and advanced AR-VLMs, demonstrating an efficient approach to building dVLMs.

Model	Size	Type	Samples	SeedBench [img]	SeedBench [vid]	AI2D	ChartQA	Realworld QA	MuirBench
AutoRegressive Vision Language Models									
LLaVA	7B	AR	-	37.0	23.8	-	-	-	-
LLaVA-1.5	7B	AR	-	66.1	37.3	-	-	-	-
Cambrian-1	8B	AR	-	74.7	-	73.0	73.3	64.2	-
LLaVA-OV	7B	AR	7.8M	<u>75.4</u>	<u>56.9</u>	81.4	80.0	<u>66.3</u>	41.8
Qwen2.5VL	3B	AR	>9M	74.8	55.1	<u>81.6</u>	<u>84.0</u>	65.4	<u>47.7</u>
	7B	AR	>9M	**77.5**	**61.3**	**83.9**	**87.3**	**68.5**	**59.6**
Diffusion Vision Language Models									
LaViDa-L	8B	Diff.	1.6M	66.5	-	70.0	64.6	-	-
Dimple	7B	Diff.	1.3M	-	-	74.4	63.4	-	-
LLaDA-V	8B	Diff.	16.5M	<u>74.8</u>	<u>53.7</u>	77.8	78.3	<u>63.2</u>	**48.3**
DiffusionVL	3B	Diff.	738K	73.9	52.2	<u>78.4</u>	<u>79.9</u>	61.6	<u>47.2</u>
	7B	Diff.	738K	**75.5**	**54.4**	**82.2**	**84.2**	**68.0**	44.8

Table 2: Benchmark Performance Comparison (Part 2). DiffusionVL-7B achieves top-tier performance among Diffusion models and closes the gap with top AR models. Best in **bold**, second-best <u>underlined</u>.

4.3 Feasible dVLM Conversion from AR-LMs

In this section, we conduct multiple groups of experiments on different base language models and various finetuning paradigms to investigate the feasibility of finetuning dVLMs from AR-LMs. We perform autoregressive finetuning, block diffusion finetuning, and full diffusion finetuning based on the AR-LM Qwen2.5-7B-Instruct and the dLLM LLaDA-8B-Instruct, respectively, and report the differences in their performance on the downstream multimodal benchmarks under the same training settings and data.

As shown in Tab. 3, DiffusionVL significantly outperforms LLaDA-V (finetuned from LLaDA using block diffusion and full diffusion paradigms). This indicates that building a high-performance dVLM does not require a pre-existing dLLM; we can fully leverage existing powerful AR-LMs to develop such dVLMs. Notably, DiffusionVL exhibits negligible differences from LLaVA on downstream benchmarks. This further confirms that constructing dVLMs from AR-LMs is both feasible and effective.

It is worth noting that the performance difference between the DiffusionVL here and DiffusionVL in Sec. 4.2 does not imply that finetuning from AR-VLMs to dVLMs is the best approach. The DiffusionVL in Sec. 4.2 benefits more from the fact that its base model has already undergone extensive, high-quality vision-language alignment training. We believe that AR-LMs, with longer and higher-quality visual finetuning, also have the potential to build dVLMs achieving performance comparable to dVLMs built from AR-VLMs on downstream benchmarks.

Model	Paradigm	Base LLM (MMLU)	MMMU (val)	MMMU-Pro (std.)	ChartQA	RealWorld QA	MME (cog.)
Conversion from LLaDA-8B (dLLM base)							
LLaDA-V	Full-Diff	65.9	42.4	26.0	20.2	60.1	342
LLaDA-V	Block-Diff.		32.6	17.1	29.8	44.2	261
Conversion from Qwen2.5-7B (AR-LM base)							
LLaVA	AR	71.9	**45.4**	<u>28.1</u>	<u>52.8</u>	<u>60.4</u>	<u>356</u>
DiffusionVL	Block-Diff.		<u>43.7</u>	**28.4**	**53.6**	**60.7**	**371**

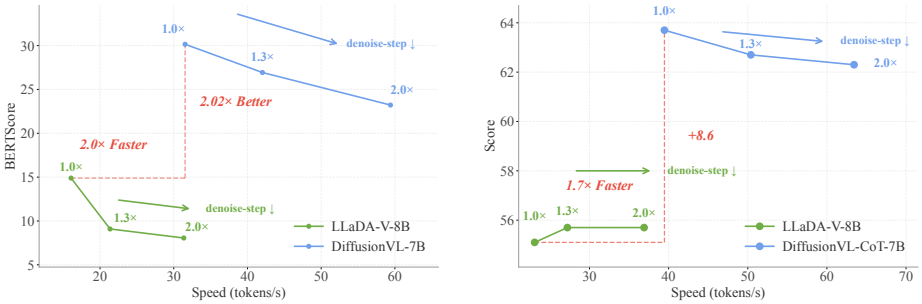
Table 3: A comparison of dVLM construction using different models and paradigms. We compare dVLMs converted from AR-LMs versus those from dLLMs. The Base LLM column indicates MMLU scores of the base language models. The best results across all models are highlighted in **bold**, and the second-best are <u>underlined</u>. The AR, Block-Diff, and Full-Diff paradigms represent the different paradigms we discussed in Sec. 3.1.

4.4 Inference Speed and Quality

To evaluate inference speed and its trade-off with quality, we evaluate DiffusionVL on two representative tasks: detailed image captioning on DetailCaps [11] and mathematical reasoning on multiple benchmarks, including MathVista [27], MathVision [1], and MathVerse [50]. During inference, we employ the static low-confidence remasking strategy to control the number of decoded tokens at each denoising step. For both tasks, we compare against LLaDA-V-8B [44] under its recommended fast-dLLM caching [37] with recomputation every 32 steps. Fig. 4 presents the speed-quality trade-offs on DetailCaps and MathVista, while Tab. 4 summarizes the full results across all math benchmarks. A complementary theoretical FLOPs analysis is provided in the supplementary material.

We first examine image captioning on DetailCaps, limiting responses to 512 tokens and evaluating against ground-truth captions using BERTScore [51]. As shown in Fig. 4(a), our DiffusionVL-7B achieves a BERTScore 2.02× higher than LLaDA-V-8B while enjoying 2.0× faster inference under the same parallelism. Moreover, we observe a clear test-time compute scaling law: increasing the number of denoising steps improves descriptive performance at the cost of inference speed, highlighting a tunable trade-off for DiffusionVL.

Turning to mathematical reasoning, we encounter a different challenge. While our DiffusionVL achieves competitive accuracy, its TPS lags behind LLaDA-V. This limitation, also observed in LaViDA [20], stems from the absence of chain-of-thought (CoT) data in our training corpus. To address this, we randomly sample 100K CoT examples from OpenMMReasoner-SFT [49] and finetune the model with the original dataset, yielding DiffusionVL-CoT. For LLaDA-V, we enable fast-dLLM caching with a generation length of 64 (longer lengths degrade accuracy), while DiffusionVL-CoT uses a maximum length of 1024 to accommodate longer reasoning chains. As Fig. 4(b) illustrates, DiffusionVL-CoT-7B outperforms LLaDA-V-8B by 8.6 points on MathVista while achieving 1.7× faster inference. Tab. 4 further confirms these gains across all three math benchmarks.



(a) DetailCaps captioning performance. **(b)** MathVista benchmark performance.

Fig. 4: Speed and quality trade-offs on detailed captioning and math reasoning. We define the parallelism factor for dVLMs as the average number of tokens generated simultaneously throughout the sequence (e.g., 1× corresponds to single-token sampling). Speed metrics were collected on 8 GPUs and averaged per device.

Model	MathVista		MathVerse		MathVision	
	Score↑	TPS↑	Score↑	TPS↑	Score↑	TPS↑
DiffusionVL-7B	**64.30**	5.21	**36.29**	14.38	**19.08**	17.74
LLaDA-V-8B	55.10	<u>23.07</u>	31.35	<u>23.18</u>	17.11	<u>22.75</u>
DiffusionVL-CoT-7B	<u>63.70</u>	**39.47**	<u>32.74</u>	**42.31**	<u>18.75</u>	**39.26**

Table 4: Comprehensive comparison on math reasoning benchmarks. We report both accuracy (Score) and inference throughput in tokens per second (TPS). Best results are in **bold** and second-best are <u>underlined</u>.

4.5 Ablation Study

AR vs. diffusion finetuning. To isolate the effect of diffusion conversion itself, we compare diffusion finetuning with a matched AR finetuning baseline using the same Qwen2.5VL-3B initialization, data, and compute. As shown in Tab. 5, the two finetuning paradigms achieve comparable multimodal performance. This suggests that the downstream capability of DiffusionVL is largely inherited from the strong AR initialization, while diffusion finetuning trades a small quality gap on some benchmarks for parallel decoding and higher DetailCaps throughput.

Generality across model families. As shown in Tab. 6, DiffusionVL can be instantiated from different AR-VLM families. In addition to Qwen2.5VL, we extend the conversion to Qwen3-VL and InternVL3.5, which use different vision towers, tokenizers, and connectors. The results show that stronger AR base models yield stronger DiffusionVL variants, supporting the generalizability of our formulation for translating autoregressive models into dVLMs.

Different block size performance. To further explore the impact of different block sizes on the performance of diffusion finetuning, we conduct four groups of diffusion finetuning ablation experiments based on the Qwen2.5VL-3B-Instruct,

Model	MMMU [val]	MMStar [test]	MME [cog./perp.]	ChartQA	AI2D	DetailCaps (BERT/TPS)
Qwen2.5VL-3B	46.8	55.9	620/1533	84.0	81.6	29.6/36.08
+ AR finetuning	46.8	58.6	651/1589	84.7	80.7	32.1/34.17
+ Diffusion finetuning	47.2	55.9	594/1539	79.9	78.4	30.9/46.37

Table 5: AR vs. diffusion finetuning. We compare AR finetuning and diffusion finetuning under matched data and compute.

Model	MMMU [val]	MMMU-Pro [std.]	MMMU-Pro [vision]	MME [cog./perp.]	ChartQA	AI2D
DiffusionVL-Qwen2.5VL-3B	47.2	31.2	20.2	594/1539	79.9	78.4
DiffusionVL-Qwen3VL-4B	52.8	36.1	23.1	650/1636	80.9	83.2
DiffusionVL-InternVL3.5-4B	55.1	38.4	28.6	609/1651	85.1	81.8

Table 6: Cross-family conversion. DiffusionVL conversion generalizes across different AR-VLM model families.

setting the training block size from 1 to 16. During inference, we set the number of denoising steps equal to the block size to ensure models with different block sizes use the same number of denoising steps when generating the same tokens.

As shown in Tab. 7(a), smaller block sizes yield marginally better accuracy, while larger ones offer greater parallel potential and less computational overhead, leading to substantially higher throughput (1.55× from block size 1 to 16). Block size 8 achieves the best DetailCaps BERTScore and offers a good speed–quality balance, so we adopt it as our default configuration.

Different noise performance. To validate the effectiveness of our block-level noise scheduling strategy, we compare it against the conventional sequence-level noise used in prior dVLMs. In sequence-level noise, masking is applied uniformly across the entire sequence. In contrast, our block-level noise applies a consistent noise ratio to each block independently. This design better aligns with the block decoding process during inference. As shown in Tab. 8, block-level noise generally outperforms sequence-level noise across multiple benchmarks.

Different diffusion paradigms. To determine the optimal diffusion paradigm for VLMs, we perform block diffusion finetuning and full diffusion finetuning from the same Qwen2.5VL-7B-Instruct. Tab. 8 demonstrates that block diffusion yields superior performance on multimodal benchmarks. Furthermore, Tab. 4 shows that DiffusionVL-CoT (block diffusion) consistently achieves higher throughput than LLaDA-V (full diffusion), highlighting the inference efficiency advantage of the block diffusion paradigm. These results indicate that block diffusion strikes a superior balance between quality and efficiency, making it the preferred paradigm for building dVLMs from AR models.

Different thresholds for dynamic remasking. To explore more extreme acceleration, we adopt the dynamic low-confidence remasking strategy and evaluate BERTScore/TPS on DetailCaps under different thresholds with a fixed

(a) Training Block Size					*(b) Dynamic Remasking*		
Benchmark	b=1	b=4	b=8†	b=16	**Threshold** τ	**TPS↑**	**BERT↑**
ChartQA	**82.8**	80.9	79.9	77.4	1.0 (static†)	31.5	**30.2**
MMMU-Pro (std.)	**32.1**	31.4	31.2	31.0	0.8	34.6	30.1
MMMU-Pro (vis.)	**22.2**	19.9	20.2	18.1	0.6	37.8	29.7
MMStar	**57.8**	56.5	55.9	55.2	0.4	46.6	27.4
DetailCaps	29.8	29.9	**30.9**	29.9	0.3	55.5	24.8
TPS (tok/s)	26.9	38.8	39.0	**41.6**	0.2	**71.8**	18.6

Table 7: Ablation on block size and dynamic remasking. (a) Effect of training block size on performance and throughput. (b) Dynamic low-confidence remasking at varying thresholds with fixed 8 denoising steps. † denotes our default. Best per group in **bold**. TPS measured per GPU on DetailCaps.

Setting	**MMMU-Pro (std.)**	**MMMU-Pro (vis.)**	**MMStar**	**ChartQA**	**MME (cog.)**	**MME (perp.)**
Noise Strategy						
Block†	**36.94**	**24.97**	**63.18**	**84.20**	**675.4**	1519
Sequence	36.30	24.86	61.86	83.88	670.4	**1527**
Diffusion Paradigm						
Block (Bd=8)†	**36.94**	**24.97**	**63.18**	**84.20**	**675.4**	1519
Full	36.47	19.88	62.08	52.00	660.7	**1607**

Table 8: Ablation on noise strategy and diffusion paradigm (Based on Qwen2.5VL-7B-Instruct). † denotes our default configuration. Best results per group are in **bold**.

denoising step of 8. This setting minimizes the number of tokens denoised in each fixed-schedule step, allowing a clearer comparison between dynamic and static remasking.

Tab. 7(b) shows that smaller dynamic thresholds allow the model to decode all tokens meeting the threshold condition at each denoising step, thereby achieving more significant acceleration. However, such acceleration comes at the cost of a certain degree of performance degradation for the model.

Theoretical FLOPs analysis. In addition to wall-clock throughput, we report theoretical inference TFLOPs in Tab. 9. We compare an *algorithm-only* setting with fixed L_p=2048 and an *end-to-end* setting using the full image-text input pipeline on an 850×600 image, where DiffusionVL and LLaDA-V produce 315 and 3654 visual tokens, respectively. DiffusionVL is consistently more efficient in the end-to-end setting, and the FLOPs gap grows rapidly for long outputs as block-wise diffusion decoding dominates. For short outputs, prefill cost narrows

L_g	*Algorithm-only (L_p=2048, LLM only)*			*End-to-end (real image + text input)*		
	DiffusionVL	LLaDA-V (fast-dLLM)	ratio	DiffusionVL	LLaDA-V (fast-dLLM)	ratio
1	28.6	28.4	1.0×	7.2	63.2	8.8×
32	31.9	43.8	1.4×	11.3	80.6	7.1×
128	44.8	360.1	8.0×	23.7	517.6	21.9×
512	97.0	4437.6	45.8×	73.5	5460.2	74.3×

Table 9: Theoretical inference TFLOPs. We compare DiffusionVL and LLaDA-V across generation lengths L_g.

the algorithmic gap, explaining why TPS alone can understate the theoretical efficiency advantage.

5 Conclusion and Future Work

Conclusion. This paper focuses on a novel problem in the building of dVLMs: Is it possible to construct dVLMs based on existing powerful autoregressive models? In response, we propose DiffusionVL, a dVLM family that translates from any powerful autoregressive models. We obtain two key observations: (1) The paradigm shift from AR-VLMs to dVLMs is remarkably effective. (2) Direct conversion of an AR-LM to a dVLM is also feasible. Furthermore, we introduce a block decoding strategy into dVLMs that supports arbitrary-length generation and KV-cache reuse. With this integrated design, despite training with less than 5% of the data required by prior methods, DiffusionVL translated from AR-VLMs achieves state-of-the-art performance among existing dVLMs, alongside a 2.0× inference speedup. DiffusionVL translated from AR-LMs not only outperforms the dVLMs built from dLLMs but also achieves performance competitive with AR-VLMs finetuned under the same autoregressive paradigm.

Future Work. During our research on DiffusionVL, we observe that the AR-to-diffusion paradigm conversion is remarkably straightforward. We plan to explore whether this conversion paradigm can extend beyond finetuning to other training stages, including pretraining and reinforcement learning. Moreover, our inference experiments reveal that the diffusion paradigm yields significant efficiency gains in long chain-of-thought (CoT) scenarios, which motivates future work on the throughput benefits that DiffusionVL could bring to reinforcement learning.

Acknowledgements

This work was partly supported by the National Natural Science Foundation of China (NSFC U25B2067).

References

1. Ahmad, M.A., Ahmed, T., Aslam, M., Rehman, A., Alamri, F.S., Bahaj, S.A., Saba, T.: Mathvision: An accessible intelligent agent for visually impaired people to understand mathematical equations. IEEE Access **13**, 6155–6165 (2024)
2. Arriola, M., Gokaslan, A., Chiu, J.T., Yang, Z., Qi, Z., Han, J., Sahoo, S.S., Kuleshov, V.: Block diffusion: Interpolating between autoregressive and diffusion language models. arXiv preprint arXiv:2503.09573 (2025)
3. Arriola, M., Venkat, N., Granskog, J., Germanidis, A.: Adapting autoregressive vision language models for parallel diffusion decoding (2025), `https://runwayml.com/research/autoregressive-to-diffusion-vlms`, accessed: 30 June 2026
4. Austin, J., Johnson, D.D., Ho, J., Tarlow, D., Van Den Berg, R.: Structured denoising diffusion models in discrete state-spaces. Advances in neural information processing systems **34**, 17981–17993 (2021)
5. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
6. Bie, T., Cao, M., Chen, K., Du, L., Gong, M., Gong, Z., Gu, Y., Hu, J., Huang, Z., Lan, Z., Li, C., Li, C., Li, J., Li, Z., Liu, H., Liu, L., Lu, G., Lu, X., Ma, Y., Tan, J., Wei, L., Wen, J.R., Xing, Y., Zhang, X., Zhao, J., Zheng, D., Zhou, J., Zhou, J., Zhou, Z., Zhu, L., Zhuang, Y.: Llada2.0: Scaling up diffusion language models to 100b (2025)
7. Chen, L., Li, J., Dong, X., Zhang, P., Zang, Y., Chen, Z., Duan, H., Wang, J., Qiao, Y., Lin, D., et al.: Are we on the right way for evaluating large vision-language models? Advances in Neural Information Processing Systems **37**, 27056–27087 (2024)
8. Chen, M., Tworek, J., Jun, H., Yuan, Q., de Oliveira Pinto, H.P., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G., Mishkin, P., Chan, B., Gray, S., Ryder, N., Pavlov, M., Power, A., Kaiser, L., Bavarian, M., Winter, C., Tillet, P., Such, F.P., Cummings, D., Plappert, M., Chantzis, F., Barnes, E., Herbert-Voss, A., Guss, W.H., Nichol, A., Paino, A., Tezak, N., Tang, J., Babuschkin, I., Balaji, S., Jain, S., Saunders, W., Hesse, C., Carr, A.N., Leike, J., Achiam, J., Misra, V., Morikawa, E., Radford, A., Knight, M., Brundage, M., Murati, M., Mayer, K., Welinder, P., McGrew, B., Amodei, D., McCandlish, S., Sutskever, I., Zaremba, W.: Evaluating large language models trained on code (2021)
9. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., 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. pp. 24185–24198 (2024)
10. Cheng, S., Bian, Y., Liu, D., Jiang, Y., Liu, Y., Zhang, L., Wang, W., Guo, Q., Chen, K., Qi*, B., Zhou, B.: Sdar: A synergistic diffusion–autoregression paradigm for scalable sequence generation (2025)
11. Dong, H., Li, J., Wu, B., Wang, J., Zhang, Y., Guo, H.: Benchmarking and improving detail image caption. arXiv preprint arXiv:2405.19092 (2024)
12. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)

13. Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., Ji, R., Shan, C., He, R.: Mme: A comprehensive evaluation benchmark for multimodal large language models (2025)
14. Gong, S., Agarwal, S., Zhang, Y., Ye, J., Zheng, L., Li, M., An, C., Zhao, P., Bi, W., Han, J., et al.: Scaling diffusion language models via adaptation from autoregressive models. arXiv preprint arXiv:2410.17891 (2024)
15. Han, X., Kumar, S., Tsvetkov, Y.: Ssd-lm: Semi-autoregressive simplex-based diffusion language model for text generation and modular control. arXiv preprint arXiv:2210.17432 (2022)
16. Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., Farhadi, A.: A diagram is worth a dozen images. In: European conference on computer vision. pp. 235–251. Springer (2016)
17. Li, B., Zhang, K., Zhang, H., Guo, D., Zhang, R., Li, F., Zhang, Y., Liu, Z., Li, C.: Llava-next: Stronger llms supercharge multimodal capabilities in the wild (May 2024)
18. Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., Shan, Y.: Seed-bench: Benchmarking multimodal llms with generative comprehension. arXiv preprint arXiv:2307.16125 (2023)
19. Li, S., Gu, J., Liu, K., Lin, Z., Wei, Z., Grover, A., Kuen, J.: Lavida-o: Elastic large masked diffusion models for unified multimodal understanding and generation (2025)
20. Li, S., Kallidromitis, K., Bansal, H., Gokul, A., Kato, Y., Kozuka, K., Kuen, J., Lin, Z., Chang, K.W., Grover, A.: Lavida: A large diffusion language model for multimodal understanding. arXiv preprint arXiv:2505.16839 (2025)
21. Li, Y., Xiong, K., Guo, X., Li, F., Yan, S., Xu, G., Zhou, L., Chen, L., Sun, H., Wang, B., et al.: Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. arXiv preprint arXiv:2506.08052 (2025)
22. Liao, B., Chen, S., Yin, H., Jiang, B., Wang, C., Yan, S., Zhang, X., Li, X., Zhang, Y., Zhang, Q., et al.: Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 12037–12047 (2025)
23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in neural information processing systems **36**, 34892–34916 (2023)
24. Liu, Y., Cao, Y., Li, H., Luo, G., Chen, Z., Wang, W., Liang, X., Qi, B., Wu, L., Tian, C., et al.: Sequential diffusion language models. arXiv preprint arXiv:2509.24007 (2025)
25. Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., et al.: Mmbench: Is your multi-modal model an all-around player? In: European conference on computer vision. pp. 216–233. Springer (2024)
26. Lou, A., Meng, C., Ermon, S.: Discrete diffusion modeling by estimating the ratios of the data distribution. arXiv preprint arXiv:2310.16834 (2023)
27. Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.W., Galley, M., Gao, J.: Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv preprint arXiv:2310.02255 (2023)
28. Masry, A., Long, D.X., Tan, J.Q., Joty, S., Hoque, E.: Chartqa: A benchmark for question answering about charts with visual and logical reasoning. arXiv preprint arXiv:2203.10244 (2022)
29. Nie, S., Zhu, F., You, Z., Zhang, X., Ou, J., Hu, J., Zhou, J., Lin, Y., Wen, J.R., Li, C.: Large language diffusion models. arXiv preprint arXiv:2502.09992 (2025)
30. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)

31. Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., Qiu, Z.: Qwen2.5 technical report (2025)
32. Tao, H., Liao, B., Chen, S., Yin, H., Zhang, Q., Liu, W., Wang, X.: Infinitevl: Synergizing linear and sparse attention for highly-efficient, unlimited-input vision-language models (2025)
33. Tschannen, M., Gritsenko, A., Wang, X., Naeem, M.F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al.: Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. arXiv preprint arXiv:2502.14786 (2025)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
35. Wang, F., Fu, X., Huang, J.Y., Li, Z., Liu, Q., Liu, X., Ma, M.D., Xu, N., Zhou, W., Zhang, K., et al.: Muirbench: A comprehensive benchmark for robust multi-image understanding. arXiv preprint arXiv:2406.09411 (2024)
36. Wu, C., Zhang, H., Xue, S., Diao, S., Fu, Y., Liu, Z., Molchanov, P., Luo, P., Han, S., Xie, E.: Fast-dllm v2: Efficient block-diffusion llm. arXiv preprint arXiv:2509.26328 (2025)
37. Wu, C., Zhang, H., Xue, S., Liu, Z., Diao, S., Zhu, L., Luo, P., Han, S., Xie, E.: Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. arXiv preprint arXiv:2505.22618 (2025)
38. Wu, T., Fan, Z., Liu, X., Zheng, H.T., Gong, Y., Jiao, J., Li, J., Guo, J., Duan, N., Chen, W., et al.: Ar-diffusion: Auto-regressive diffusion model for text generation. Advances in Neural Information Processing Systems **36**, 39957–39974 (2023)
39. xAI: Grok-1.5 vision preview: Connecting the digital and physical worlds with our first multimodal model (2024)
40. Yang, L., Tian, Y., Li, B., Zhang, X., Shen, K., Tong, Y., Wang, M.: Mmada: Multimodal large diffusion language models. arXiv preprint arXiv:2505.15809 (2025)
41. Yao, J., Wang, C., Liu, W., Wang, X.: Fasterdit: Towards faster diffusion transformers training without architecture modification. Advances in Neural Information Processing Systems **37**, 56166–56189 (2024)
42. Yao, J., Yang, B., Wang, X.: Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 15703–15712 (2025)
43. Ye, J., Xie, Z., Zheng, L., Gao, J., Wu, Z., Jiang, X., Li, Z., Kong, L.: Dream 7b: Diffusion large language models. arXiv preprint arXiv:2508.15487 (2025)
44. You, Z., Nie, S., Zhang, X., Hu, J., Zhou, J., Lu, Z., Wen, J.R., Li, C.: Llada-v: Large language diffusion models with visual instruction tuning. arXiv preprint arXiv:2505.16933 (2025)
45. Yu, R., Ma, X., Wang, X.: Dimple: Discrete diffusion multimodal large language model with parallel decoding. arXiv preprint arXiv:2505.16990 (2025)
46. Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., et al.: Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9556–9567 (2024)

47. Yue, X., Zheng, T., Ni, Y., Wang, Y., Zhang, K., Tong, S., Sun, Y., Yu, B., Zhang, G., Sun, H., et al.: Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. arXiv preprint arXiv:2409.02813 (2024)
48. Zhang, K., Li, B., Zhang, P., Pu, F., Cahyono, J.A., Hu, K., Liu, S., Zhang, Y., Yang, J., Li, C., Liu, Z.: Lmms-eval: Reality check on the evaluation of large multimodal models (2024)
49. Zhang, K., Wu, K., Yang, Z., Li, B., Hu, K., Wang, B., Liu, Z., Li, X., Bing, L.: Openmmreasoner: Pushing the frontiers for multimodal reasoning with an open and general recipe. arXiv preprint arXiv:2511.16334 (2025)
50. Zhang, R., Jiang, D., Zhang, Y., Lin, H., Guo, Z., Qiu, P., Zhou, A., Lu, P., Chang, K.W., Qiao, Y., et al.: Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In: European Conference on Computer Vision. pp. 169–186. Springer (2024)
51. Zhang*, T., Kishore*, V., Wu*, F., Weinberger, K.Q., Artzi, Y.: Bertscore: Evaluating text generation with bert. In: International Conference on Learning Representations (2020)
52. Zhu, L., Huang, Z., Liao, B., Liew, J.H., Yan, H., Feng, J., Wang, X.: Dig: Scalable and efficient diffusion models with gated linear attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7664–7674 (June 2025)
53. Zou, Y., Yao, J., Yu, S., Zhang, S., Liu, W., Wang, X.: Turbo-vaed: Fast and stable transfer of video-vaes to mobile devices. arXiv preprint arXiv:2508.09136 (2025)