Chain-of-Thought in Vision-Language-Action Models

Based on "CoT-VLA: Visual Chain-of-Thought Reasoning for Vision-Language-Action Models" [43]

Alexander Epple* Universität Ulm
Patricia Stöhr† Universität Ulm
Timo Ropinski‡ Universität Ulm

Abstract

Fundamentally, Vision-Language-Action Models are pretrained Vision-Language-Models repurposed for generalizable robotic control. Vanilla approaches typically lack intermediate reasoning steps, which limits performance and their ability to handle complex tasks. To address this limitation, the *CoT-VLA* architecture introduces sub-goal images as a visual reasoning step, enabling the model to "think visually" before predicting action sequences. This results in significant performance gains compared to concurrent, non-reasoning methods such as OpenVLA [16]; beating the baselines by +6% on simulated benchmarks and +17% in real-world tasks. Additionally, the method enables using action-less videos for pretraining, yielding additional gains. However, inference overhead and sub-goal generation quality remain open challenges, pointing toward latent reasoning and world models as promising future directions.

1 Introduction

Embodied robotics has progressed significantly in recent times. A critical milestone in the advancement of Vision-Language-Action Models (VLAs) occurred when *RT-2* [45] managed to place a Coke can on an image of Taylor Swift. Given the out-of-distribution nature of the prompt, this demonstrated that VLAs could be a scalable, generalizable direction of research. These kinds of models use pretrained Vision-Language-Models (VLMs) for scene understanding and instruction following [16, 10], but crucially take actions without intermediate reasoning. Without reasoning, performance on long-horizon tasks is limited. Intermediate reasoning, or Chain-of-Thought (CoT), has been shown to be effective, especially in Large Language Models (LLMs) [36], so applying it to embodied systems is a natural extension.

CoT-VLA [43] implements this idea in the form of visual sub-goal generation, which serves as an intermediate reasoning step before predicting action chunks (see Figure 1). While it was among the first VLAs to employ visual CoT, there have been many attempts since to use visual, textual and latent reasoning [39, 42, 3]. Every method and modality has its own unique challenges and advantages and so does *CoT-VLA*. One aspect is autoregressive image generation, because unlike other methods [26, 6], images and actions are both generated by the VLM in the form of tokens. This means image generation and action prediction need to be discrete, rather than using methods such as flow-matching, which operates in continuous space.

Finally, VLAs also promise to advance the domain of medical robotics, opening the door to automated surgery among other applications, such as surgical assistance or autonomous endoscopy [20, 11]. In these high-stakes environments, precision, adaptability and reliability are crucial and, therefore, make reasoning abilities even more important. Unfortunately, these applications are also data-scarce, meaning few demonstrations are available and being able to utilize unlabeled data is particularly valuable.

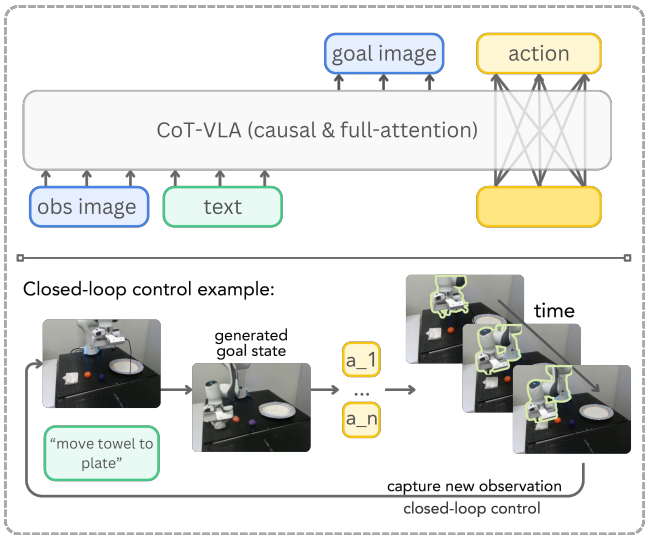

Figure 1: The *CoT-VLA* architecture and a closed-loop control example. Based on visual and textual inputs, a sub-goal image is generated first, followed by a chunk of robot commands (actions). The figure was adapted from the original paper [43].

In this paper, the required concepts are introduced first (2), before describing and positioning the architecture in question (3). Then, the effectiveness of the approach is detailed and compared to subsequent and concurrent architectures (4). Finally, its issues are discussed and an outlook for potential future directions in the form of world models is provided (5).

2 Background

This section covers the fundamental concepts and builds up to the architectural idea behind the *CoT-VLA* model. First, VLAs are introduced (2.1), followed by an explanation of CoT (2.2) and its origin in LLMs. Finally, we detail how to generate image autoregressively and in a discrete manner (2.3).

2.1 Vision-Language-Action Models

VLAs were first coined as a term in *RT-2* [45], but had been introduced previously in *RT-1* [5]. The core idea is to employ a VLM as a robot controller, VLMs being LLMs that have been equipped with a vision backbone. This vision encoder, usually a Vision Transformer (ViT) [9] such as *CLIP* [29], consumes the input image and projects its output into the token space of the LLM. Examples of such models include *LLaVA* [19] and *PaLM-E* [10].

These multimodal LLMs are pretrained on internet-scale amounts of data and therefore generalize well to unseen data. VLMs already understand concepts such as object categories and can follow language instructions, making them a natural fit for robot control. VLAs can leverage these general capabilities for text instruction and vision processing, but usually require further finetun-

*e-mail:alexander.epple@uni-ulm.de
†e-mail:patricia.stoehr@uni-ulm.de
‡e-mail:timo.ropinski@uni-ulm.de

ing on robot demonstrations to learn the mapping between the multimodal inputs to robot commands. These commands, or actions, can either be generated autoregressively by the LLM in the form of discrete action tokens [45, 16], or in a continuous fashion via diffusion models or flow matching [33].

$$\{\hat{a}_t(,\ldots,\hat{a}_{t+m})\} \sim P_\theta(\{a_t(,\ldots,a_{t+m})\} \mid s_t, l) \tag{1}$$

In the former case, as modeled by Equation (1), individual action tokens $\hat{a}_t$ or chunks of m actions $\{\hat{a}_t,\ldots,\hat{a}_{t+m}\}$ are sampled at a time and translated into control signals for the robot's actuators. The model P, which is parameterized by θ, receives as input the current observation s_t and instruction l. An open-source, State-Of-The-Art (SOTA) example for this architecture is *OpenVLA* [16], which was trained on the *Open X-Embodiment* dataset [28]. Figure 2 shows a general overview of a vanilla VLA.

The continuous methods predict the entire action sequence at once via iterative denoising. For example, π_0 [4] uses flow matching to generate actions. VLAs are, however, limited in their planning abilities. Complex action sequences often have to be solved in one forward pass without intermediate steps. This is why performance suffers on tasks that require long-term planning or multi-step decision making. CoT reasoning, which will be introduced in the next section, has been explored as a possible solution in various works [45, 39, 23].

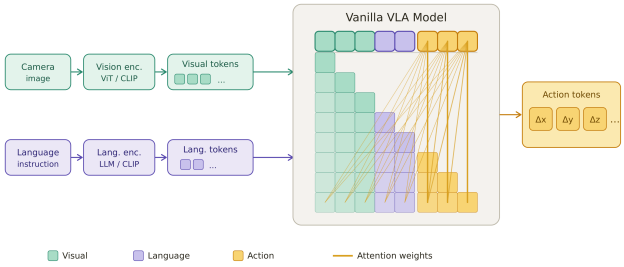

Figure 2: Example of a vanilla VLA architecture. Images from the robot's camera and textual instructions are processed by a VLM, which outputs discrete action tokens with causal attention.

2.2 Chain-of-Thought Reasoning

CoT was first introduced in the realm of LLMs by Wei *et al.* [36] as a prompting technique. Essentially, CoT prompts the LLM to reason through problems step by step, embedding the reasoning chain in the context. Crucially, this is an emergent behavior that does not require additional training or finetuning. The original paper uses few-shot prompting, where examples are provided in the prompt, but zero-shot prompting ("think step by step") can also work [17].

More recently, CoT is also learned and enforced during training directly [40] and has also been explored in the visual domain, with multimodal instead of purely text based reasoning [31]. This can be achieved by, for example, asking the LLM to determine the most important region in form of a bounding box [31] or by first analyzing the image content and then reasoning about it [44].

In the robotics domain, *SayCan* [1] uses text-based reasoning to discern which and how to use predefined skills. Decomposing robot movement into sub-tasks is explored further by *EmbodiedGPT* [23], which uses CoT to generate sub-goals instead of solving the entire task at once. *CoTDiffusion* [26] generates sub-goal images with a diffusion model, which are consumed by a foundation model that is supposed to generate the actions necessary to reach them. Whereas diffusion models generate images from noise, it is also possible to do so autoregressively via next-token prediction, which will be explained in more detail in the next section.

2.3 Autoregressive Image Generation

Discrete image generation has been used in Variational Auto Encoders (VAEs) before the advent of the transformer architecture [35]. Since then, it has gained particular interest, as transformer-based LLMs typically operate on discrete tokens. This makes discrete image generation a natural fit for multimodal LLMs, enabling images as both inputs and outputs. One way to solve the problem of images existing on a continuous manifold is using residual quantization, as introduced in *RQ-VAE* [18].

$$r_0 = z, \quad r_d = r_{d-1} - t_d, \quad t_d = \arg\min_{c \in C} \|r_{d-1} - c\| \tag{2}$$

The idea of residual quantization is that an image patch z can be approximated by a sum of D codebook entries $\sum_{d=1}^{D} t_d$ in a coarse-to-fine manner, as shown in Equation (2). This is done by taking a finite quantization codebook C, which can be thought of as the visual vocabulary, and selecting the entry t_d closest to the remaining residual r_{d-1} at each step. This entry is then subtracted to obtain the next residual r_d.

In *RQ-VAE* two transformers are employed to generate images, which work in lockstep and are referred to as the *RQ-Transformer*. First, for every position t the spatial transformer processes all $t-1$ tokens generated so far with causal attention to produce a context vector. Then, the depth transformer takes this context and autoregressively generates D tokens, which when stacked form an image patch for position t. After the RQ-Transformer, a decoder consumes all quantized tokens to reconstruct an image. Quantization is inherently lossy, which bounds the reconstruction quality by the codebook size $|C|$ and depth D.

The multimodal LLM *VILA-U* [37] builds on *RQ-VAE* and can generate text and image contents fully discretely and autoregressively, with performance comparable to other SOTA VLMs. Unlike most other VLMs, however, it is trained jointly on understanding *and* generation tasks, making it a suitable backbone for *CoT-VLA*, which requires image understanding and generation capabilities.

3 Method

With the necessary background in mind, this section covers the motivation and setup (3.1) of the CoT-VLA model (3.2). It is then positioned and compare to the broader landscape of VLAs, focusing on the various ways CoT has been integrated in other works (3.3). Finally, to show the flexibility and use cases of VLAs, an overview of their applications in the medical domain is given (3.4).

3.1 Problem Statement

In a vanilla VLA such as *OpenVLA*, actions $\hat{a}_t$ are predicted directly from observations s_t and instructions l (Equation (1)). The authors of CoT-VLA hypothesize, however, that an intermediate visual reasoning step can improve the action predictions, by encouraging the model to "think visually" [43]. This is intended to help the model think about how the state should develop before predicting the sequence to achieve it.

$$\hat{s}_{t+n} \sim P_\theta(s_{t+n} \mid s_t, l) \tag{3}$$

$$\{\hat{a}_t,\ldots,\hat{a}_{t+m}\} \sim P_\theta(a_t,\ldots,a_{t+m} \mid s_t, l, \hat{s}_{t+n}) \tag{4}$$

Formally, the idea is to predict a sequence of m actions by conditioning the model P_θ additionally on a sub-goal image $\hat{s}_{t+n}$ (Equation (3)). This image, which is generated by the model before predicting the next action sequence, is n frames into the future (Equation (4)). The next section details how this is done in practice and how action tokens are decoded.

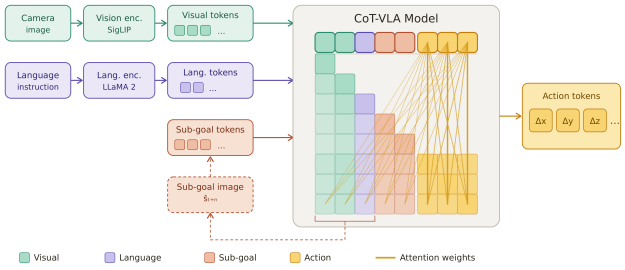

Figure 3: The architecture of *CoT-VLA*. Images from the robot's camera and textual instructions are used to generate a sub-goal image, which is concatenated to inputs. The model outputs discrete action tokens, which fully attend to all other action tokens in a chunk.

3.2 The CoT-VLA Architecture

As described in Section 2.1, VLAs require a backbone to process images and text. In the case of *CoT-VLA*, image generation is also a necessity, which is why Zhao *et al.* opted for the *VILA-U* [37] foundation model.

The sub-goal images are generated before action chunks are sampled, using the same model conditioned on the latest real frame as well as the instruction. Image generation uses causal attention, where each image or text token can only attend to the tokens preceding it. The sub-goal image represents an in-between or goal state when executing the next action chunk. During training, the predicted image tokens are factored into the model's loss, as there are real future frames to compare the generated image against.

With the sub-goal generated, an action chunk is predicted with full attention, so all tokens can attend to each other. The action chunk prediction is conditioned on the sub-goal image, the real image and instruction. Arguably, this echoes how diffusion-based, continuous-action models generate trajectories to some extent, since full attention allows for temporal coherence within the chunk.

Action tokens are finally converted to actuator commands by mapping them to bins, which correspond to concrete values (e.g. rotation of a gripper). Each action dimension can take one of 256 values, represented by the 256 least frequently used tokens in the vocabulary of the language model. During training, actions from real demonstrations are used to calculate the loss.

Unlike other methods [16, 7, 5, 45], it is also possible to leverage action-less training data, in which case only the sub-goal generation loss is applied. See Figure 6 for example real-world rollouts, including generated sub-goals.

3.3 Positioning & Comparison

This section is split into four parts in order to fairly place and compare *CoT-VLA* to its peers. The model is first compared to those lacking reasoning capabilities, followed by those with textual or image-based reasoning, and finally models with reasoning in latent space.

Models without reasoning. Many prior VLA models have not employed the use of CoT at all and predict actions from the inputs directly [33, 4, 5, 45]. Most notably, *OpenVLA* [16] is a foundation model, which predicts discrete action tokens and is built around two vision encoders (*SigLIP* [41] & *DinoV2* [27]), as well as a 7B-parameter LLM (*Llama 2* [34]). Being trained on a large dataset of $\approx$ 1M robot demonstrations [28] and built on SOTA vision and language models, it was ahead of its closed-source competition [45] at the time. *CoT-VLA* improves upon *OpenVLA* and others [33, 4] in most tasks and benchmarks, especially those involving long-context or multi-instruction tasks. This shows that adding intermediate reasoning via CoT improves downstream performance.

Textual CoT models. In terms of purely text-based CoT, which most current LLMs natively support, several prior and posterior works have made use of this to enhance action generation. *ECoT* builds on *OpenVLA*, adding CoT to the existing model [39] in several ways. Tasks are decomposed into sub-tasks and an overall plan, and textual move reasoning grounds actions in the observations. They also find that naive CoT, where the thinking step has no predefined structure, only marginally improves performance. Similarly, the $\pi_{0.5}$ model [15] decomposes tasks into sub-tasks and outputs bounding boxes for relevant objects in text before committing, yielding significant improvements over its predecessor, π_0. *GraspVLA* [8] predicts bounding boxes for target objects and grasping poses as reasoning steps before an action expert generates action chunks. Finally, *ThinkAct* [14] implements a Reinforcement Learning (RL) approach to action prediction, using a reward signal to guide reasoning directly. Overall, the performance of *ThinkAct*, *ECoT* and *GraspVLA* is mostly on par with *CoT-VLA*, with only the more recent $\pi_{0.5}$ beating it.

Visual CoT models. While *CoT-VLA* [43] was among the first to integrate visual CoT into VLA models, several works have also explored the idea. *DreamVLA* [42] uses "dream queries" to predict dynamic regions, depth maps and semantic annotations. More recently, *UD-VLA* [6] generates sub-goal images and actions with a diffusion model in tandem. *CoTDiffusion* [26] also generates sub-goal images using a diffusion model, which a foundation model uses to generate action sequences. Compared to these visual CoT models, *CoT-VLA* improves upon *CoTDiffusion*, but trails both *DreamVLA* and *UD-VLA* by a similar, sizable margin, with the gap being especially pronounced in long-horizon tasks. Chen *et al.* attribute *UD-VLA*'s strong results to the fused sub-goal and action diffusion.

Latent CoT models. Lastly, there have also been efforts to move reasoning entirely into latent space. Building on the idea of *ThinkAct* [14], its evolution *Fast-ThinkAct* [13] moves textual reasoning into latent space. The main difference to the previous version is a student-teacher setup, which enables distilling the reasoning traces into compact latent reasoning tokens. These latent tokens capture the intended trajectory and significantly improve latency and performance. The very recent *LaRA-VLA* [3] goes one step further, as it features both textual and image-based latent reasoning. The model is trained in several stages, where it first learns text-based CoT reasoning and predicting latent states of future frames. Then, the reasoning steps, namely sub-task generation, bounding box prediction and motion reasoning, are progressively moved into latent space. Action generation is coarse and discrete at first, and later replaced by a flow-matching-based action expert. Compared to latent CoT models, *CoT-VLA* does fall behind quite significantly. *Fast-ThinkAct* is much better at long-horizon tasks, and *LaRA-VLA* outdoes *CoT-VLA* in all tasks by a wide margin. It should also be noted that *LaRA-VLA* has an even lower inference-time latency than *Fast-ThinkAct*.

3.4 VLAs In The Medical Domain

There are many possible avenues where robots could be used in medical applications, such as autonomous robot surgery [12], autonomous endoscopy [11] or surgery assistance [20].

RoboNurse-VLA [20] is a robotic scrub nurse system, that enables autonomous surgical instrument handovers. Their model is largely based on *OpenVLA* [16], but uses *SAM 2* [30] as the vision encoder and instructions are in the form of speech commands. The processing pipeline first detects bounding boxes of relevant instruments and the surgeon's hand, before applying segmentation and

Table 1: Performance comparison on the LIBERO benchmark [21] grouped by CoT paradigm. Results reported as success rate (%). The best results are **bold**, the results of CoT-VLA are *italicized*. Results taken directly from the respective papers evaluated on LIBERO.

CoT Type	Method	Spatial	Object	Goal	Long	Avg.
No CoT	Octo [33]	78.9	85.7	84.6	51.1	75.1
	OpenVLA [16]	84.7	88.4	79.2	53.7	76.5
	VLA-JEPA [32]	96.2	99.6	97.2	95.8	97.2
Textual CoT	ThinkAct [14]	88.3	91.4	87.1	70.9	84.4
	$\pi_{0.5}$ [15]	**98.8**	98.2	98.0	92.4	96.8
Visual CoT	*CoT-VLA* [43]	*87.5*	*91.6*	*87.6*	*69.0*	*83.9*[1]
	DreamVLA [42]	97.5	94.0	89.5	89.5	92.6
	UD-VLA [6]	94.1	95.7	91.2	89.6	92.7
Latent CoT	Fast-ThinkAct [13]	92.0	97.2	90.2	79.4	89.7
	LaRA-VLA [3]	96.4	**99.8**	**98.6**	**96.6**	**97.9**

[1] *CoT-VLA* reports 81.1% in the original paper; the value shown is an average of the four tasks for consistency.

masking. While this is not explicitly CoT, it does serve a similar role to intermediate reasoning. The finetuned *RoboNurse-VLA* achieved near perfect success rates on all evaluation tasks compared to finetuned baselines.

He *et al.* introduce *CapsDT* [11], an endoscopy VLA to control capsule robots in stomachs with a magnet that is attached to a robot arm. Endoscopy capsule robots are camera systems enclosed with a permanent magnet, which the patient swallows to enable internal imaging [25]. Their model is diffusion-based and trained on a self-collected dataset, but does not use any kind of CoT. They then compared their model against baselines finetuned on their dataset. While their success rates on various navigation tasks are only $\approx 20\%$, they nonetheless handily beat the baselines. It should be noted, however, that these tasks are extremely challenging and the baselines have near-zero success rates.

Finally, *Cosmos-H-Surgical* [12] is a surgical world model and can predict inverse action dynamics for two frames. A world model is essentially a video generation model with a high-level understanding of physics and spatio-temporal causality. While this model is not a VLA in the traditional sense, it can be used to predict future frames and infer robot actions from pairs of consecutive frames. World models are conceptually similar to CoT, in that both aim to ground action generation in an understanding of how the scene will evolve. Given the scarcity of data in the medical domain, a surgical world model could serve as a valuable foundation for medical VLAs. They also test inverse dynamics prediction models on real surgery robot trajectories and can show that additional training on synthetic data from their world model significantly improves performance. A similar trend toward medical-domain specialization can be observed in image segmentation, where *MedSAM* [22] demonstrates that a model tailored specifically to medical imaging can outperform general-purpose segmentation approaches, suggesting that specialized, domain-adapted models are a viable and established direction for medical applications more broadly.

In summary, VLAs in the medical domain are certainly an area of active research, but little has been done in terms of integrating CoT, visual or otherwise, in these models.

4 Evaluation

To evaluate the effectiveness of the *CoT-VLA* architecture, we compare its performance to contemporary and more recent models on the *LIBERO* [21] benchmark (4.1). Then, to evaluate the architectural design choices, the contribution of individual components are briefly discussed (4.2).

4.1 Performance Comparison

The *LIBERO* [21] benchmark was chosen as a reference point for the analysis that follows, as it is a VLA evaluation standard. Because of this, most contemporary and recent works report results for it. The benchmark features four task suites that evaluate distinct capabilities in a synthetic environment. *LIBERO-Spatial* tests spatial relationship understanding, while *LIBERO-Object* tests recognition and manipulation of diverse object types. Both suites evaluate declarative knowledge. *LIBERO-Goal* tests procedural knowledge and assesses whether the model has internalized the motions and behaviors required to achieve a given goal. Finally, *LIBERO-Long* combines both, evaluating long-horizon tasks that require sustained declarative and procedural reasoning, making it particularly interesting to analyze the effectiveness of CoT. The results for the individual models and categories are summarized in Table 1 and grouped by CoT type.

It is apparent that the addition of any type of CoT is beneficial, as *CoT-VLA* improves upon *OpenVLA* and *Octo* by approximately $+7\%$ overall and does especially well on long-horizon tasks ($+15\%$). As discussed in Section 3.3, *CoT-VLA* is no longer SOTA in the visual CoT domain, as more recent models perform significantly better overall ($+9\%$), which can largely be attributed to their strong long-horizon task performance ($+21\%$). Generally speaking, *CoT-VLA* served as a proof-of-concept that visual CoT is conceptually sound. Text-based and latent reasoning models also scale, though *ThinkAct* and even its latent successor *Fast-ThinkAct* improve only marginally over *CoT-VLA* ($\approx +1\%$ and $+5\%$ respectively). The latent visual and textual reasoning approach by Bai *et al.* [3] performs the best on this benchmark in all but one category. As this is one of the most recent models, it does motivate the idea that multimodal reasoning leads to better results. A clear exception is *VLA-JEPA*, which reaches impressive performance with no explicit reasoning at all. It instead relies on a world-model, an approach that will be discussed in Section 5.3. Finally it should be noted that *CoT-VLA* also performs as well as or better than contemporary methods in real-robot experiments, indicating that the method generalizes well to real-world scenarios.

4.2 Architecture Analysis

Next, the effectiveness of the architecture is discussed, as illustrated in Figure 5. In ablation studies performed by Zhao *et al.* [43], they trained four model variants and evaluated on *LIBERO* [21]. The base model has no action chunking, hybrid attention or CoT. Each subsequent model adds one component in the listed order. All components individually improve performance, justifying the

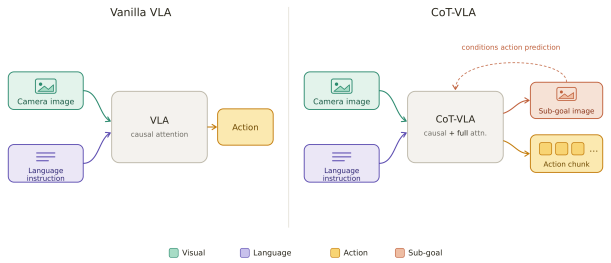

Figure 4: High level architectural comparison between *CoT-VLA* [43] and a vanilla base model. Concretely, the differences are sub-goal image generation, full attention across action tokens and the addition of action chunking.

design choices. For a direct comparison between the base model and the proposed model with all components, refer to Figure 4. In another experiment, they also test the usefulness of pretraining on combined robot demonstrations and action-less videos compared to task-specific finetuning. With a relative success rate improvement of $\approx 47\%$ on a real-world benchmark, the pretraining objective meaningfully aids generalization. It should be noted that this ablation does not isolate the specific contribution of action-less video data, as the comparison is between pretraining with the full data mixture and no pretraining at all. Unfortunately, the relative impact of the action-less portion alone remains unmeasured in the original work. Finally, in an oracle experiment, where generated sub-goal images were replaced with ground truth images, the success rate improved by $\approx 40\%$. Since the tasks in this experiment were out-of-distribution, the authors suggest that the stark difference might stem from insufficient sub-goal generation. It is unclear, however, if the quality of the generated images or the size and distribution of the sub-goal generation dataset is the cause. Overall, the architecture is effective and the design choices are sound, but the oracle experiment gap suggests that the sub-goals and sub-goal generation are the primary bottleneck, which will be explored further in the next section.

5 Discussion

This section serves to discuss Zhao *et al*'s approach and critique design limitations, several of which the authors acknowledge themselves. First, the action generation mechanism is examined (5.1), followed by the sub-goal image generation (5.2). Finally, the concept of world models, how they relate to VLAs and why they might be a promising research direction are discussed in more detail (5.3).

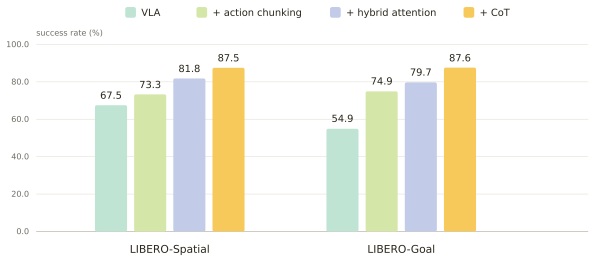

Figure 5: Impact the various architectural design choices of *CoT-VLA* have on *LIBERO-Spatial* and *LIBERO-Goal* performance. The results have been reproduced from the original paper [43].

5.1 Action Generation

The action generation and chunking as implemented in *CoT-VLA* does have its advantages. Reusing tokens from the LLM means no architectural changes are required to produce actions and full within-chunk attention enables temporal coherence and is a meaningful improvement over causal action generation. There are, however, issues related to both discretization and action chunking respectively.

As explained in Section 3.2, actions are discretized into 256 bins. One additional detail worth highlighting is that bins are based on the $\langle 1, 99 \rangle$ percentiles of the training data action distribution. This, in turn, limits the resolution of all actions the model can take and out-of-distribution actions are clipped to the nearest bin. For fine-grained tasks, this enforced rounding error could lead to degraded performance, or accumulating errors.

More importantly, action chunking means there are boundaries that can lead to motion discontinuities. The authors mention this problem themselves and attribute certain failure cases to exactly such discontinuities. Chunks are also executed without intermediate feedback, meaning errors can accumulate over the course of a chunk, as there is no mid-chunk feedback or correction.

Unfortunately, both problems can also reinforce each other: Discretization errors mid-chunk can lead to suboptimal poses, which can be compounded by motion discontinuity in the next chunk. Possible solutions to mitigate these issues include flow matching and single action prediction. Flow matching, being a continuous method, addresses discretization, while per-step prediction eliminates chunk boundaries, though at the cost of efficiency.

While these issues are not as severe as the ones described in the next section, they are not insignificant.

5.2 Inference Latency & Sub-Goal Quality

When it comes to the sub-goal images, they are clearly an effective way to improve performance. Zhao *et al.*'s hypothesis that "thinking visually" enables effective CoT in VLAs holds, as detailed in Section 4.2. Additionally, the finding in *ECoT* that lack of structure in naive textual CoT is detrimental to performance does not appear to transfer to images, as the unstructured visual sub-goals yield good improvements. Finally, everything is end-to-end trainable and being able to leverage action-less data is a significant advantage for the relatively data-scarce setting of robotics.

That being said, the sub-goal generation process does have serious issues when it comes to quality and inference speed.

Images are generated autoregressively, so the model has to perform 256 forward passes before actions are generated. While this bottleneck is somewhat remedied by action chunking, the approach comes, as discussed previously, with its own challenges. Low latency is crucial in real-time environments. As no absolute latency figures are provided, this hints at *CoT-VLA*'s latency being at best subpar. *LaRA-VLA* [3] and *Fast-ThinkAct* [13] prove that (multimodal) CoT and low latency are not mutually exclusive. Concretely, *LaRA-VLA* reports an inference latency of 135 ms per rollout on an NVIDIA A100 GPU [3], a reduction of up to 90% compared to explicit CoT-based approaches such as *ECoT* [39], which requires 4434 ms. *Fast-ThinkAct* improves upon its predecessor [14] with a 89.3% latency reduction, taking 805 ms per rollout [13].

Even though high visual accuracy is no hard requirement for better generalization, it does appear to help, as is evident from the oracle experiment described in Section 4.2. Reasons for the differences could include distribution shift or insufficient image quality, but either way the bottleneck is clearly image generation. When comparing the image generation process to *LaRA-VLA* [3], its success with latent representations, which lack pixel-level detail, indicates that semantic quality might matter more than visual fidelity. That is to say, good (latent) representations are sufficient and high frequency

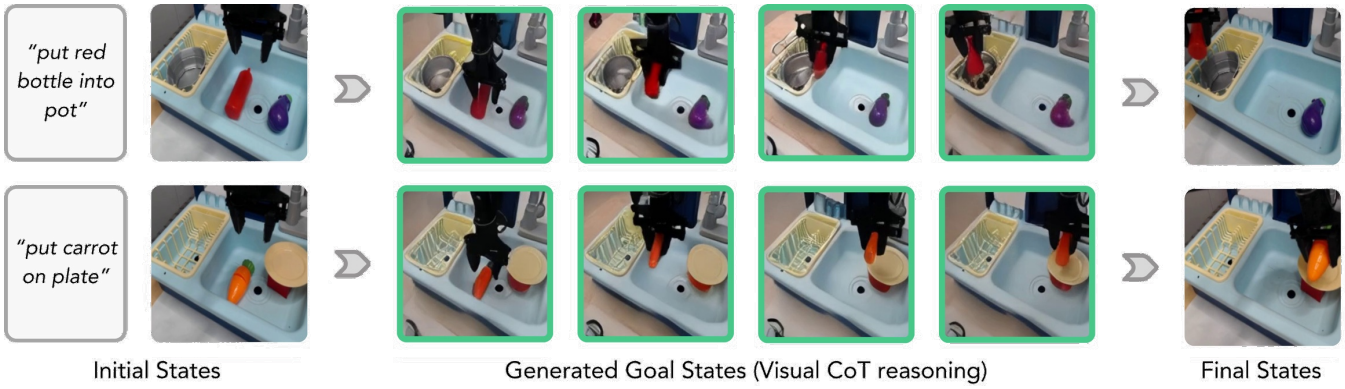

Figure 6: Real-world task execution examples, adapted from the original paper [43]. The initial state on the left consist of image s_0 and instruction l, sub-goal reasoning steps $\hat{s}_t$ are in the middle and final state s_T can be seen on the right.

details are not required. The authors acknowledge these limitations and propose advances in image generation and world models to be promising solutions, the latter of which is going to be discussed in more detail next.

5.3 Connection to World Models

The general intuition behind world models, as introduced in Section 3.4, is to teach a model to understand the physical world by predicting its future state, so it can plan how to handle new situations. Thus, *CoT-VLA* is conceptually closely related, as it learns to predict future states to aid in planning action trajectories. Considering predicting a single frame is beneficial, this raises the question what modeling richer and deeper future representations can achieve.

One example for such world models is *V-JEPA 2* [2, 24], which learns to predict future video frames in latent space entirely self-supervised. World dynamics are learned from observations alone at a semantic level, rather than high-frequency, pixel-level details (e.g., how individual leaves on a tree move). The training objective is masked latent prediction, using a vision encoder parametrized as a ViT [9] for its architecture. The pretraining dataset consists of roughly $1M$ hours of video and $1M$ images. Then, they use the latent world model to build *V-JEPA 2-AC*, an action predictor that learns to output the actions required to reach a given goal state. Training consists of both teacher forcing and short-horizon rollouts. During inference, energy minimization over candidate actions is used to select the best next action. After fine-tuning on only 62h of action-labeled data, *V-JEPA 2-AC* surpasses *Octo* [33] on zero-shot manipulation tasks. *V-JEPA 2-AC* is not conditioned on language instructions and can only be prompted with visual goals, but its strong zero-shot capabilities prove world models are an effective tool for VLAs.

V-JEPA 2 learns purely in latent space, which again indicates that latent predictions are sufficient for embodied systems. This is also evident in *VLA-JEPA* [32], a world-model-based VLA built using *V-JEPA 2*. The VLA outputs latent action tokens, which condition the latent world model for next-state prediction. The same tokens also condition the flow-matching action head for continuous trajectory prediction. *VLA-JEPA* learns from both human videos and robot data via the joint objective of next state alignment and robot action prediction. Notably, even though it uses no explicit reasoning steps, its performance on *LIBERO* is comparable to the strongest contenders (see Table 1). This underlines the idea that world dynamics prediction and reasoning are closely related concepts.

Whereas *VLA-JEPA* uses a world model in combination with a VLM, *DreamZero* [38] is a language-conditioned World-Action-Model (WAM), which jointly predicts future video frames and actions directly. Unlike VLAs, it is initialized from a video diffusion backbone pretrained on web-scale data and has stronger zero-shot capabilities. Because of this, it can also more quickly adapt to unseen environments and tasks. *DreamZero* is trained on a small ($500h$) but diverse set of tasks and environments, focusing on breadth rather than repetition. Even without task-specific repetition, it still manages to outperform $\pi_{0.5}$, particularly on unseen tasks. WAMs present a promising direction for the future of embodied systems, as they are end-to-end trainable like VLAs but can leverage rich spatio-temporal priors from web-scale data and seem to generalize better. As WAMs ground action generation in richer future state prediction, they directly address the sub-goal quality ceiling identified by the *CoT-VLA* oracle experiment. Finally, especially in the context of medical robotics, WAMs could prove to be valuable as discussed in Section 3.4. When few or no demonstrations are available, data efficiency and zero-shot generalization matter even more. Whether specialist WAMs analogous to *Cosmos-H-Surgical* [12] will emerge for medical robotics remains an open question.

6 Conclusion

In summary, Zhao *et al.*'s contribution to VLAs is the addition of visual Chain-of-Thought by generating sub-goal images before predicting action chunks. This allows the model to reason about the future before acting, which results in better performance than counterparts without reasoning abilities. Another meaningful advantage is the ability to leverage data without action annotations during pretraining to boost results.

Although the method works and yields significant results, it does fall short mainly when it comes to inference time and generator quality. The oracle experiment shows a significant performance gap, suggesting substantial room for improvement in sub-goal generation. Additionally, the introduced overhead makes real-time inference challenging. This, combined with the discontinuities due to action chunking make this method an early proof-of-concept, showing the effectiveness and value of visual reasoning.

While the overall field is moving into the direction of latent reasoning or richer future state modeling such as WAMs, the core intuition of *CoT-VLA* is continued in spirit. Grounding actions in predicted future states, be they images, latent representations or world models, is sound and seems promising for future work. Especially in environments constrained by data availability such as the medical domain, being able to use unlabeled data is crucial. It remains to be seen in what direction embodied systems develop next, but both reasoning and grounding actions in predicted future states are here to stay.

REFERENCES

[1] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

[2] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[3] S. Bai, J. Lyu, W. Zhou, Z. Li, D. Wang, L. Xing, X. Zhao, P. Wang, Z. Wang, C. Chi, et al. Latent reasoning vla: Latent thinking and prediction for vision-language-action models. *arXiv preprint arXiv:2602.01166*, 2026.

[4] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[5] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[6] J. Chen, W. Song, P. Ding, Z. Zhou, H. Zhao, F. Tang, D. Wang, and H. Li. Unified diffusion vla: Vision-language-action model via joint discrete denoising diffusion process. *arXiv preprint arXiv:2511.01718*, 2025.

[7] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[8] S. Deng, M. Yan, S. Wei, H. Ma, Y. Yang, J. Chen, Z. Zhang, T. Yang, X. Zhang, W. Zhang, et al. Graspvla: a grasping foundation model pre-trained on billion-scale synthetic action data. *arXiv preprint arXiv:2505.03233*, 2025.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[10] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[11] X. He, M. Su, X. Jiang, L. Bai, and H. Ren. Capsdt: Diffusion-transformer for capsule robot manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 414–419. IEEE, 2025.

[12] Y. He, P. Guo, M. Xu, Z. Li, A. Myronenko, D. Imans, B. Liu, D. Yang, M. Gu, Y. Ji, Y. Jin, R. Zhao, B. Shen, and D. Xu. Cosmos-h-surgical: Learning surgical robot policies from videos via world modeling, 2026.

[13] C.-P. Huang, Y. Man, Z. Yu, M.-H. Chen, J. Kautz, Y.-C. F. Wang, and F.-E. Yang. Fast-thinkact: Efficient vision-language-action reasoning via verbalizable latent planning. *arXiv preprint arXiv:2601.09708*, 2026.

[14] C.-P. Huang, Y.-H. Wu, M.-H. Chen, F. Wang, and F.-E. Yang. Thinkact: Vision-language-action reasoning via reinforced visual latent planning. *Advances in Neural Information Processing Systems*, 38:82782–82802, 2026.

[15] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al. $\pi_0.5$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[16] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[17] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[18] D. Lee, C. Kim, S. Kim, M. Cho, and W.-S. Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11523–11532, 2022.

[19] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[20] S. Li, J. Wang, R. Dai, W. Ma, W. Y. Ng, Y. Hu, and Z. Li. Robonurse-vla: Robotic scrub nurse system based on vision-language-action model. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3986–3993. IEEE, 2025.

[21] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *arXiv preprint arXiv:2306.03310*, 2023.

[22] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang. Segment anything in medical images. *Nature communications*, 15(1):654, 2024.

[23] Y. Mu, Q. Zhang, M. Hu, W. Wang, M. Ding, J. Jin, B. Wang, J. Dai, Y. Qiao, and P. Luo. Embodiedgpt: Vision-language pre-training via embodied chain of thought. *Advances in Neural Information Processing Systems*, 36:25081–25094, 2023.

[24] L. Mur-Labadia, M. Muckley, A. Bar, M. Assran, K. Sinha, M. Rabbat, Y. LeCun, N. Ballas, and A. Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026.

[25] NaviCam. NaviCam® SB System - AnX Robotics — anxrobotics.com. https://anxrobotics.com/products/navicam-sb-capsule-system/. [Accessed 15-05-2026].

[26] F. Ni, J. Hao, S. Wu, L. Kou, J. Liu, Y. Zheng, B. Wang, and Y. Zhuang. Generate subgoal images before act: Unlocking the chain-of-thought reasoning in diffusion model for robot manipulation with multimodal prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13991–14000, 2024.

[27] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[28] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

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

[30] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pages 28085–28128, 2025.

[31] H. Shao, S. Qian, H. Xiao, G. Song, Z. Zong, L. Wang, Y. Liu, and H. Li. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *Advances in Neural Information Processing Systems*, 37:8612–8642, 2024.

[32] J. Sun, W. Zhang, Z. Qi, S. Ren, Z. Liu, H. Zhu, G. Sun, X. Jin, and Z. Chen. Vla-jepa: Enhancing vision-language-action model with latent world model. *arXiv preprint arXiv:2602.10098*, 2026.

[33] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

[34] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[35] A. Van Den Oord, O. Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[36] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[37] Y. Wu, Z. Zhang, J. Chen, H. Tang, D. Li, Y. Fang, L. Zhu, E. Xie,

H. Yin, L. Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024.

[38] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[39] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

[40] E. Zelikman, Y. Wu, J. Mu, and N. Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

[41] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[42] W. Zhang, H. Liu, Z. Qi, Y. Wang, X. Yu, J. Zhang, R. Dong, J. He, H. Wang, Z. Zhang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. *Advances in Neural Information Processing Systems*, 38:24195–24228, 2026.

[43] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1702–1713, 2025.

[44] G. Zheng, B. Yang, J. Tang, H.-Y. Zhou, and S. Yang. Ddcot: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *Advances in Neural Information Processing Systems*, 36:5168–5191, 2023.

[45] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.