组会分享 PI-FAST

2025-09-26 李佩泽

FAST: Efficient Action Tokenization for Vision-Language-Action Models

Karl Pertsch[*,1,2,3], Kyle Stachowicz[*,2],
Brian Ichter[1], Danny Driess[1], Suraj Nair[1], Quan Vuong[1], Oier Mees[2], Chelsea Finn[1,3], Sergey Levine[1,2]

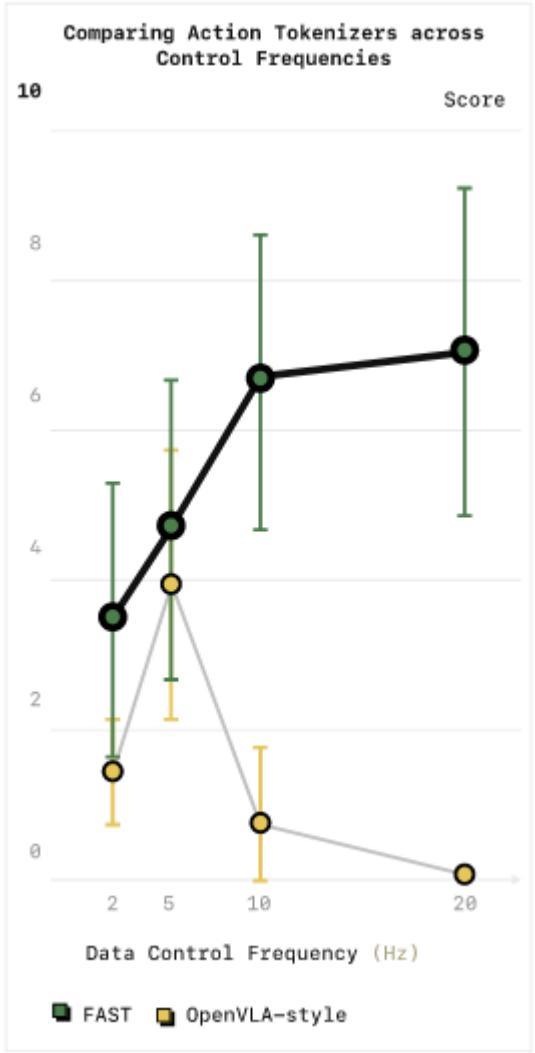
[1]Physical Intelligence, [2]UC Berkeley, [3]Stanford

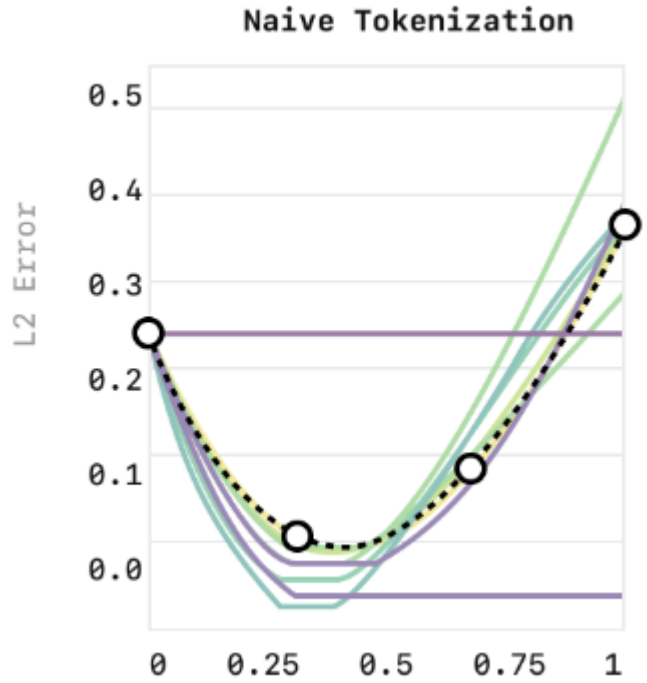
https://pi.website/research/fast

***Abstract*—Autoregressive sequence models, such as Transformer-based vision-language action (VLA) policies, can be tremendously effective for capturing complex and generalizable robotic behaviors. However, such models require us to choose a tokenization of our continuous action signals, which determines how the discrete symbols predicted by the model map to continuous robot actions. We find that current approaches for robot action tokenization, based on simple per-dimension, per-timestep binning schemes, typically perform poorly when learning dexterous skills from high-frequency robot data. To address this challenge, we propose a new compression-based tokenization scheme for robot actions, based on the discrete cosine transform. Our tokenization approach, Frequency-space Action Sequence Tokenization (FAST), enables us to train autoregressive VLAs for highly dexterous and high-frequency tasks where standard discretization methods fail completely. Based on FAST, we release FAST+, a *universal* robot action tokenizer, trained on 1M real robot action trajectories. It can be used as a black-box tokenizer for a wide range of robot action sequences, with diverse action spaces and control frequencies. Finally, we show that, when combined with the π_0 VLA, our method can scale to training on 10k hours of robot data and match the performance of diffusion VLAs, while reducing training time by up to 5x.**

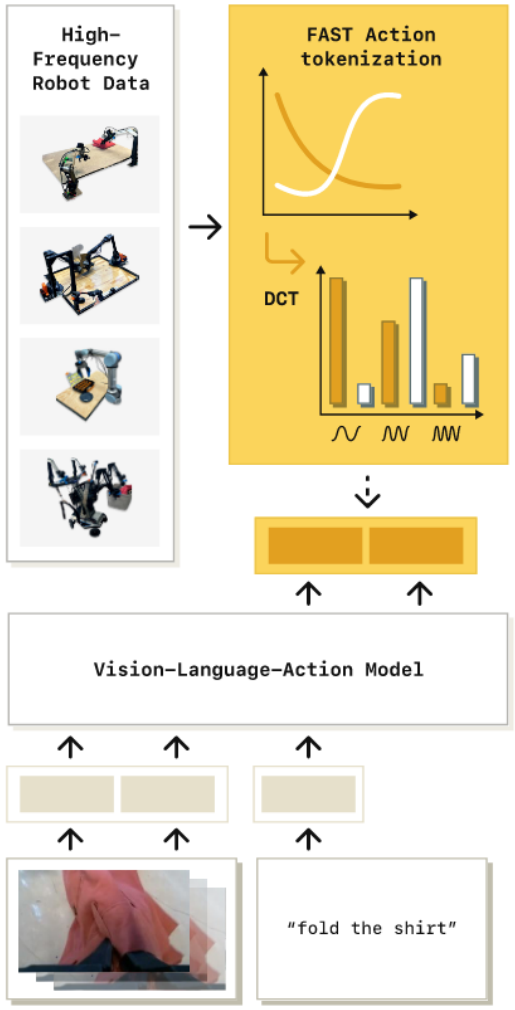
Gap

- In Auto
- Naïve Tokenizer:
- 连续时序随着采样频率升高序列边际信息递减
- 即
  - 通过简单重复上一token也可获得较低的Loss
  - 编码器没有实际获得有意义的信息





Utility and Algorithm



Algorithm 1 FAST Tokenizer

Require: scale γ, (for inference) BPE dictionary Φ

procedure FASTTOKENIZER($a_{1:H}$)

$C_j^i \leftarrow \texttt{DCT}\left(a_{1:H}^i\right)$ ▷ Compute DCT coefficients

$\bar{C}_j^i \leftarrow \texttt{round}\left(\gamma \cdot C_j^i\right)$ ▷ Quantize coefficients

$[T_k] \leftarrow \left[\bar{C}_1^1, \bar{C}_1^2, \ldots, C_2^1, \ldots, C_H^n\right]$ ▷ Flatten tokens

BPE Training:

$\phi \leftarrow \texttt{TrainBPE}(\mathcal{D} := \{[T_k]\})$

Tokenization:

$\left[\bar{T}_1, \ldots, \bar{T}_{\bar{k}}\right] \leftarrow \texttt{BPE}\left([T_1, \ldots, T_k], \phi\right)$

return action_tokens

Pipeline Overview

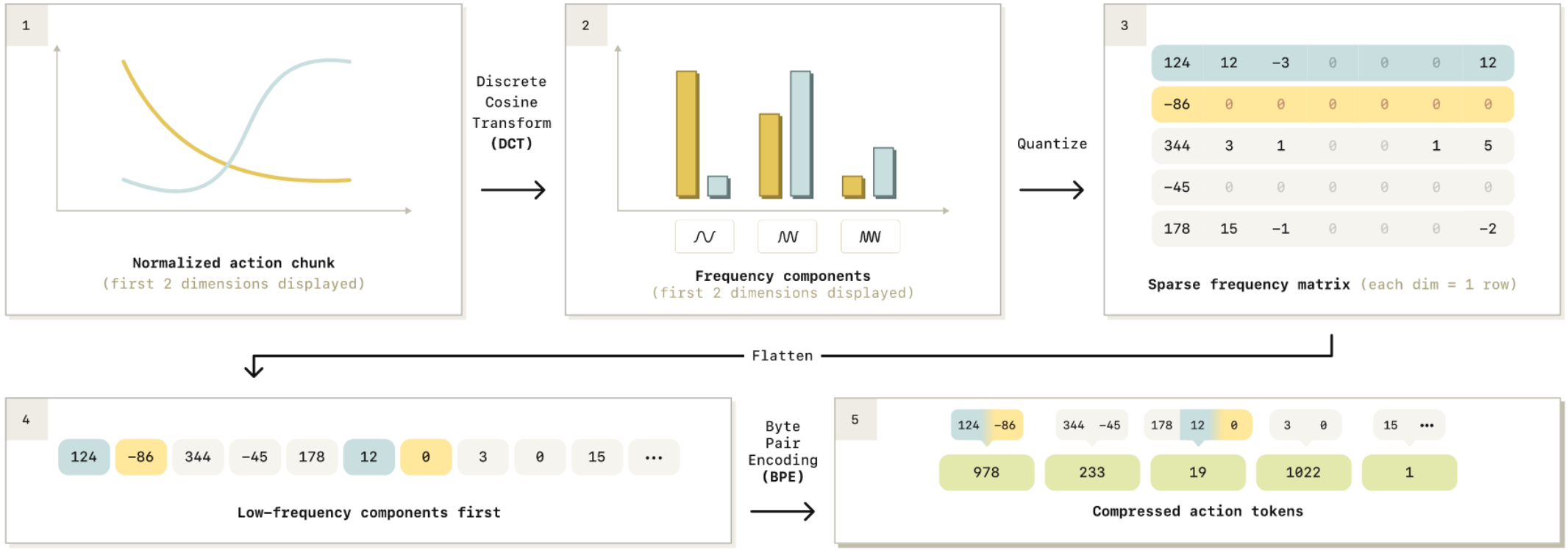
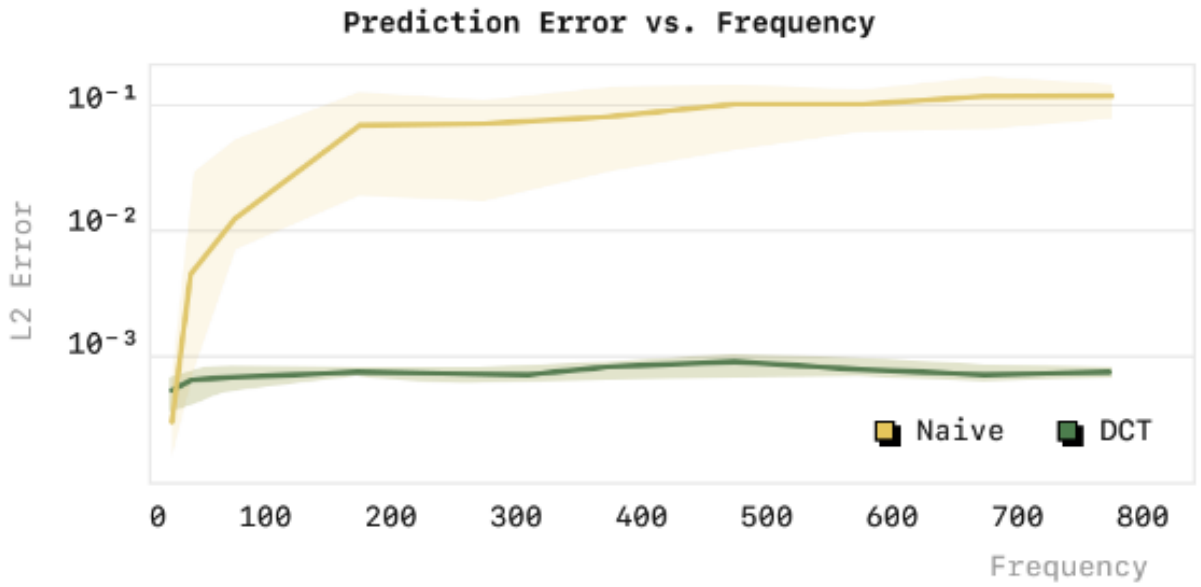


Fig. 4: **Overview of the FAST action tokenization pipeline.** Given a normalized chunk of actions, we apply discrete cosine transform (DCT) to convert the signal to the frequency domain. We then quantize the DCT coefficients and use byte-pair encoding (BPE) to compress the flattened sequence of per-dimension DCT coefficients into the final action token sequence. See Section V-B for a detailed description.

余弦编码实验



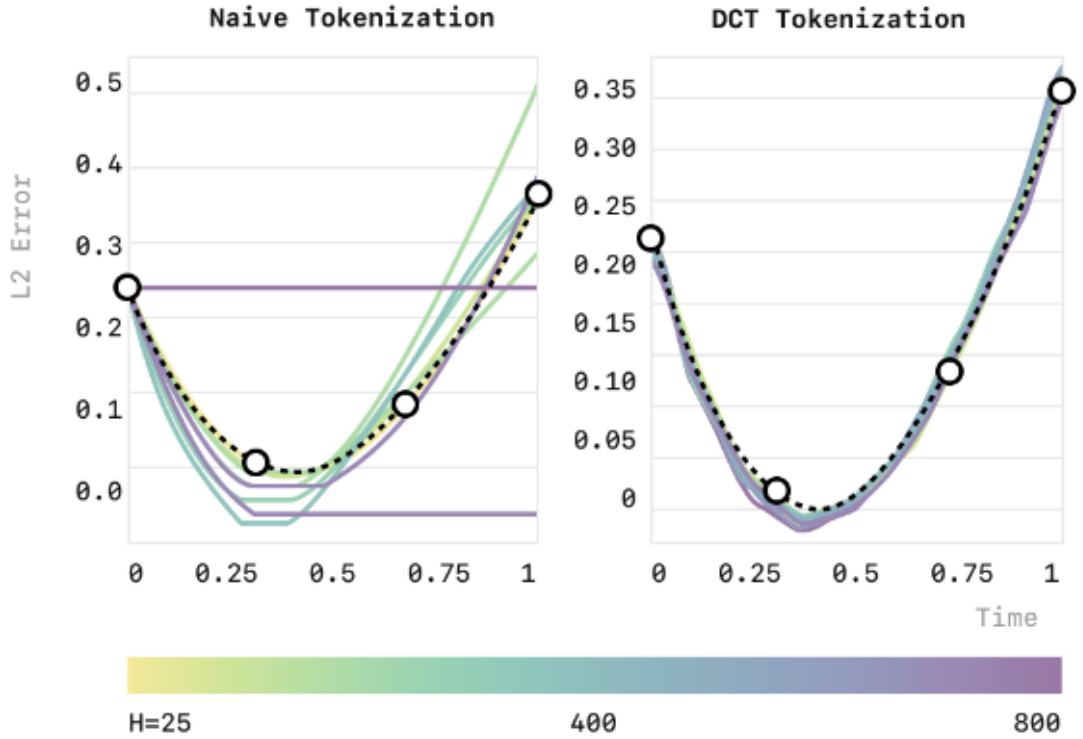


Fig. 3: **Effect of sampling rate on prediction performance.**

数据集压缩实验

B. Comparing Action Tokenizers for VLA Training

Dataset	Action Dimension	Control Frequency	Avg. Token Naive	Avg. Token FAST	Compression
BridgeV2	7	5 Hz	35	20	1.75
DROID	7	15 Hz	105	29	3.6
Bussing	7	20 Hz	140	28	5.0
Shirt Fold	14	50 Hz	700	53	13.2

TABLE I: **Comparison of the average token count per action chunk** for naïve tokenization and FAST. We use 1-second chunks in all datasets. With our method, each chunk requires many fewer tokens, particularly for high-frequency domains such as the T-shirt folding task, indicating that it is more effective at removing redundancy.

←

• 特化压缩

→

• 非特化压缩

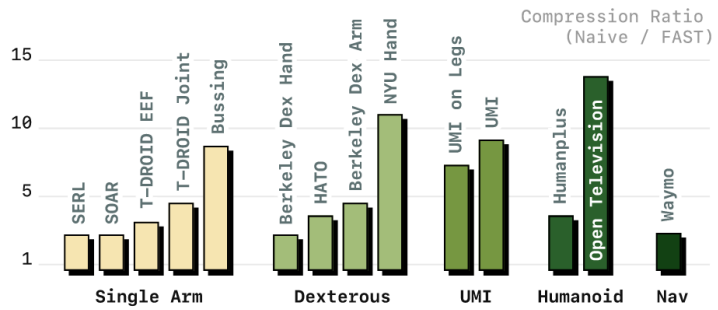
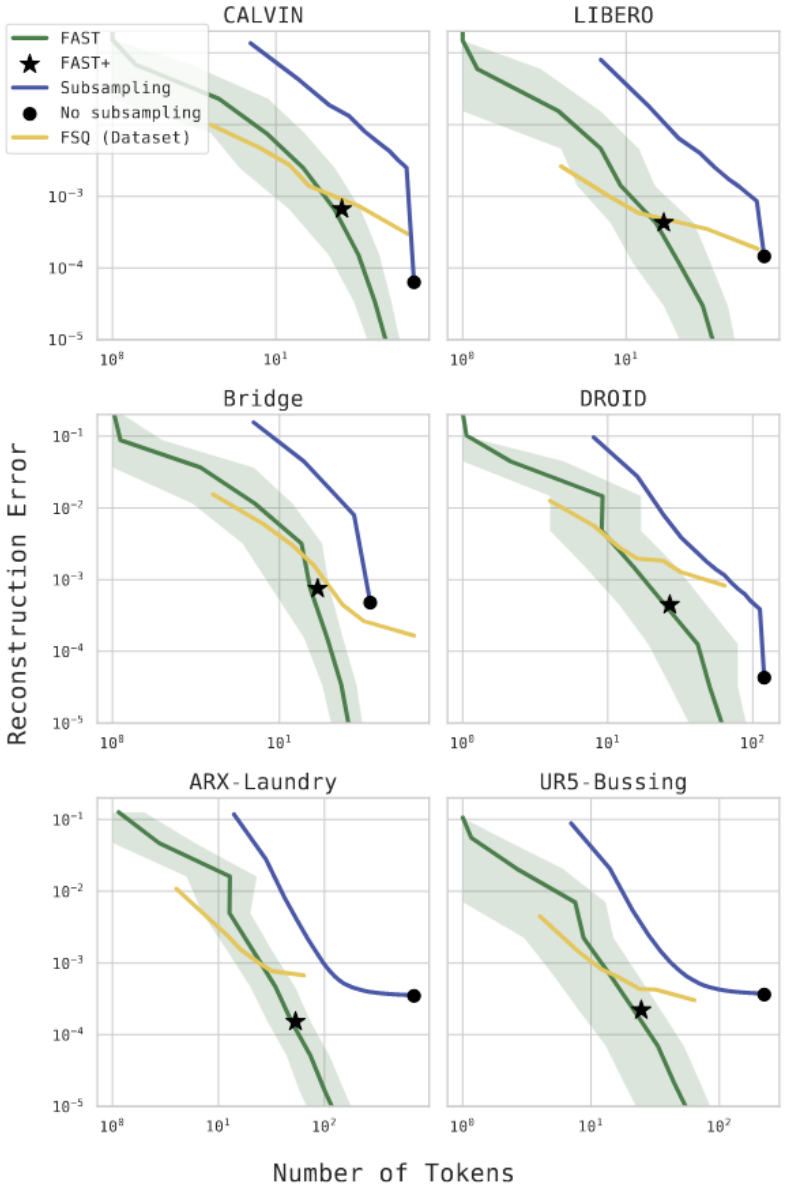


Fig. 8: **Universal tokenizer.** We test the compression rate achieved by our FAST+ tokenizer vs. naïve tokenization across diverse robot datasets, *unseen* during tokenizer training. We find that FAST is effective across a wide range of robot morphologies, action spaces and control frequencies.



• 复原效果

跨本体/频率测试

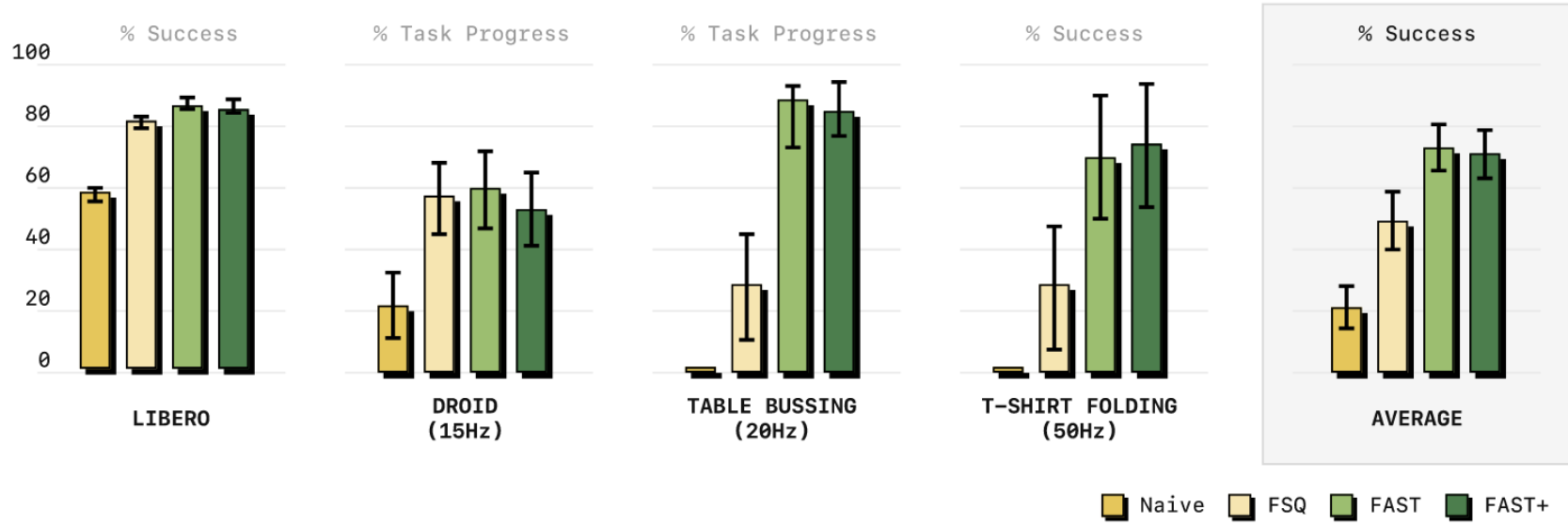
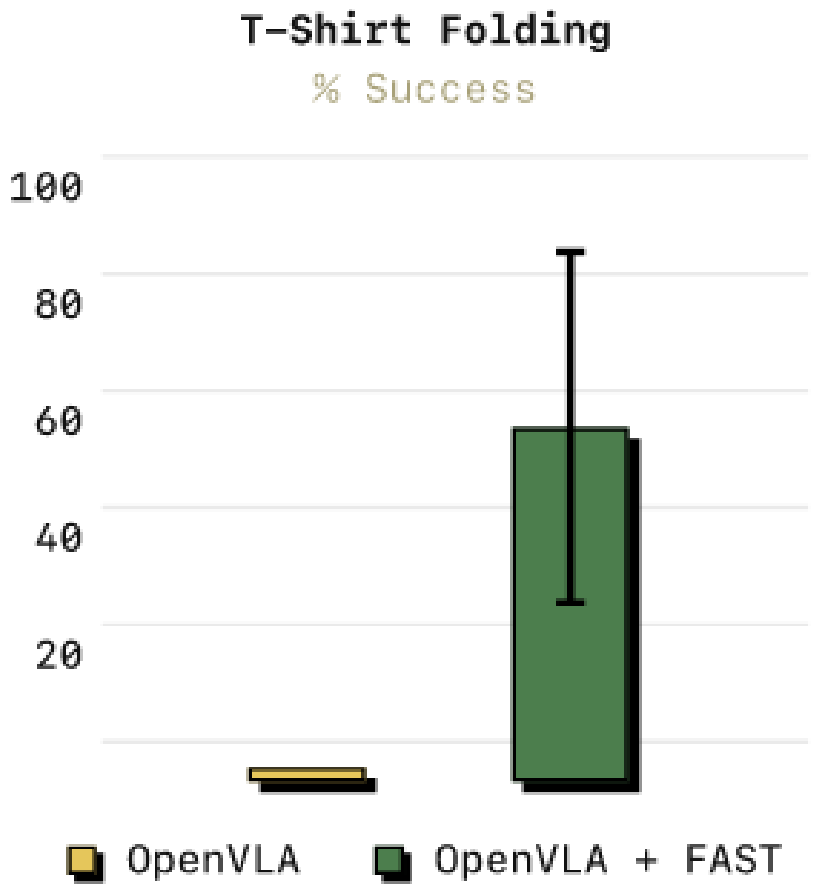
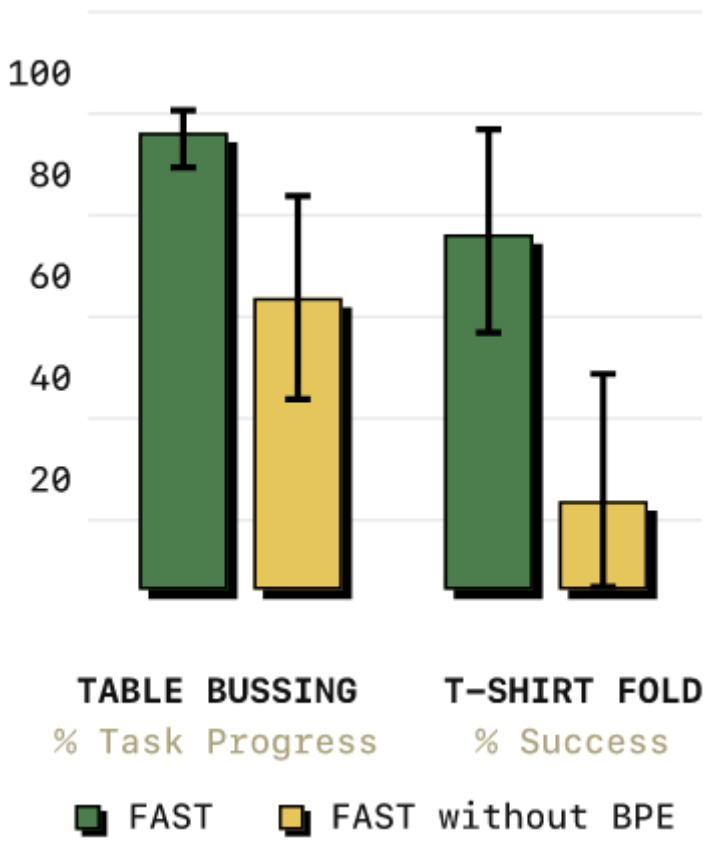


Fig. 6: **Comparison of policy performance using different tokenization approaches.** We find that tokenization approaches that compress action targets (FAST, FSQ) lead to substantially more efficient training than the naïve binning tokenization used in prior VLAs. Overall, we find that FAST leads to more effective policy training than FSQ, particularly on dexterous real-robot tasks. Our universal tokenizer, FAST+, matches the performance of dataset-specific tokenizers. We report mean and 95% CI.

跨VLA实验 & 消融BPE实验





语言跟随实验 与PI0直接比较实验

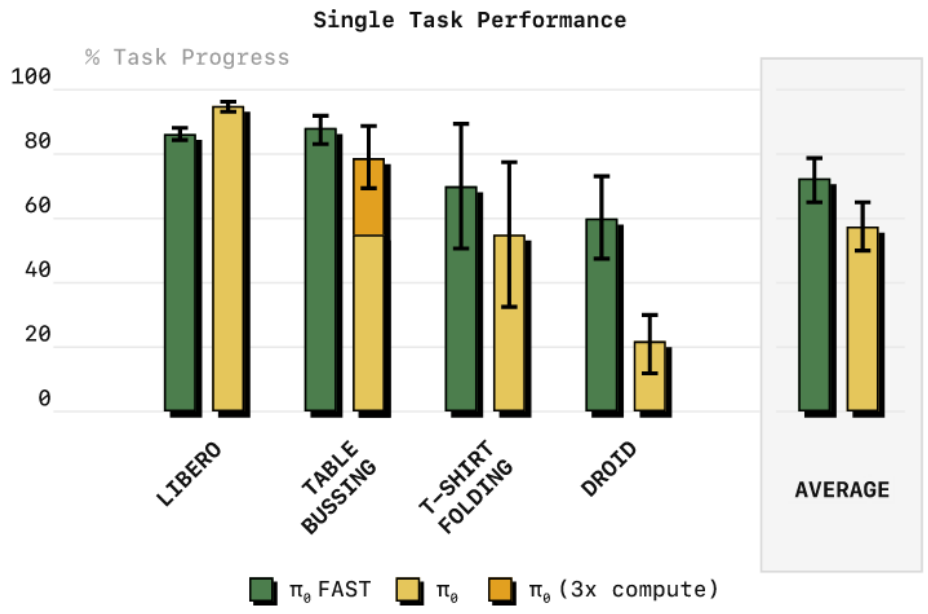


Fig. 9: **Comparison of diffusion π_0 [7] to our π_0 model with FAST decoding on single-task training.** On small datasets (Libero, T-Shirt Folding), both perform comparably. On large datasets (Table Bussing), FAST converges faster. In DROID, we find that FAST follows language instructions better. We report mean and 95% CI.

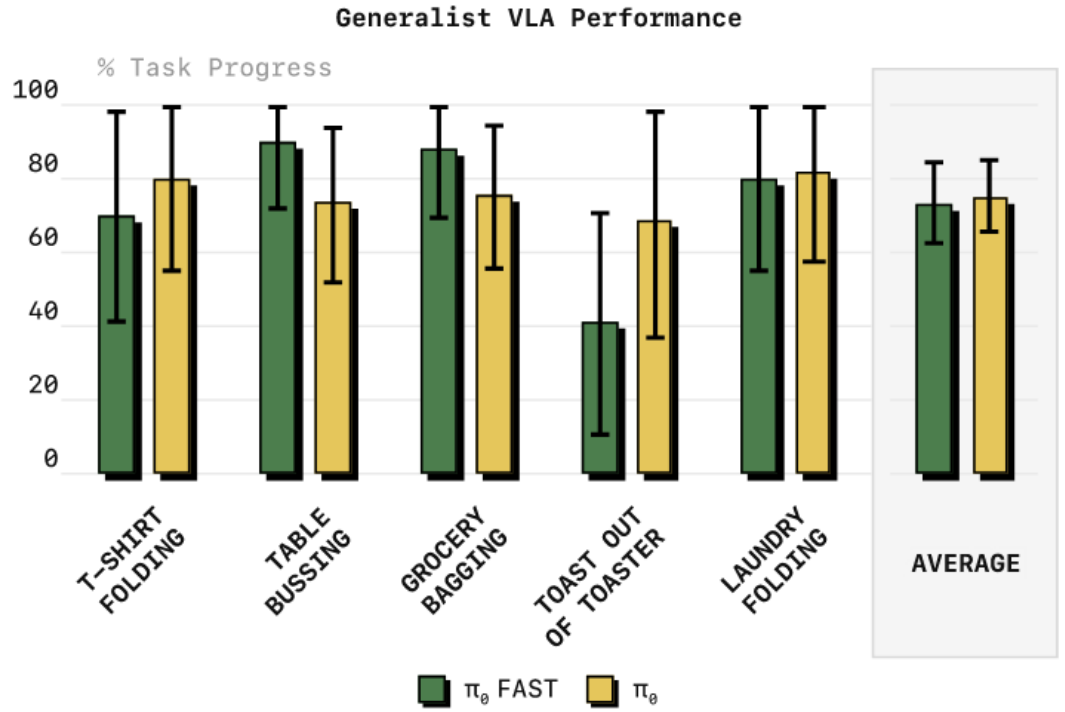


Fig. 11: **Comparison of π_0-FAST and diffusion π_0 [7] generalist policies.** π_0-FAST matches the performance of diffusion π_0 while requiring significantly less compute for training. Reported: mean and 95% CI.