FE-VLA: FORCE ENCODER VISION-LANGUAGE-ACTION POLICIES FOR COMPLIANT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action (VLA) models have driven remarkable progress in general-purpose robotic manipulation, yet they remain largely kinematics-centric and struggle in contact-rich scenarios where excessive force causes catastrophic failure, such as crushing force-sensitive objects. We advance VLA capabilities by introducing a framework for force-conditioned manipulation that treats force awareness as both a sensory input and a perceptual output. Our approach integrates a lightweight fingertip force sensor into the VLA prefix via a force encoder, allowing the model to condition its actions on real-time force feedback. An auxiliary force-prediction head taps directly into the VLM's perceptual trunk to predict the maximum safe grasping force, enabling the model to anticipate an object's physical limits from vision and force information before contact is established. A reactive safety layer then intercepts motor commands and freezes the gripper when live sensor readings exceed the predicted threshold. Evaluated across five force-sensitive everyday objects, our approach reduces peak applied force by approximately 50%, while maintaining comparable task success.

1 INTRODUCTION

Humans naturally perceive, touch, and assess objects through years of lived experience, instantly judging their fragility and texture to decide how to interact with them safely. The ability to manipulate everyday objects with this same safety and reliability is a central challenge in robotics. Recent progress in vision-language-action (VLA) models has dramatically expanded the range of tasks that robot systems can address without per-task engineering, demonstrating generalization across object categories, workspaces, and free-form language instructions (Black et al., 2024; Wen et al., 2024; 2025). Despite this progress, a critical dimension of real-world manipulation remains unaddressed by these models: the regulation of contact forces during grasping. Humans instinctively calibrate grip force to an object's fragility, cradling a ripe tomato with the lightest touch and cinching firmly around a cucumber. Robots operating in unstructured environments enjoy no such intuition: a miscalibrated squeeze can crush produce, crack eggs, or rupture soft packaging, turning an otherwise sound manipulation policy into a liability. Force awareness is therefore not an optional refinement but a functional requirement for deployment in domestic and agricultural settings.

Current VLA architectures, including π_0 (Black et al., 2024), OpenVLA (Kim et al., 2024), RT-2 (Zitkovich et al., 2023), and $\pi_{0.5}$ (Intelligence et al., 2025) map visual observations and language instructions to sequences of motor commands (typically 6-DoF end-effector poses or joint-angle trajectories) without any explicit output channel for contact force. Language-grounded manipulation planners such as VoxPoser (Huang et al., 2023) compose 3D spatial constraints from language models but likewise rely on kinematic primitives rather than force constraints for execution. Recent work has begun incorporating tactile sensing into VLA frameworks: Huang et al. (2025) introduce a hybrid position-force controller that interprets force-related language modifiers (*e.g.*, "gently") as run-time force targets; Cheng et al. (2025) align tactile representations semantically with vision and language to improve contact-rich grasping success rates. Complementarily, Gao et al. (2024) demonstrate that vision-language models fine-tuned on physical concept annotations can accurately reason about object fragility and deformability. Despite this growing body of work, force-aware VLA models targeting everyday fragile produce remain scarce, as do standardized datasets with force signals directly compatible with VLA training pipelines. We build on π_0 (Black et al., 2024) because its dual-stream architecture, comprising a pretrained VLM prefix and a separate flow-matching action

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

expert, provides a natural injection point for force tokens and a clean attachment site for the auxiliary force-prediction head, without changing the action decoder's architecture. Our main contributions are:

- **Force-sensing encoder.** A lightweight convolutional module that converts a short window of fingertip force readings into a small set of tokens and injects them into the VLA input stream alongside image and language tokens. The encoder is fully compatible with pretrained VLA checkpoints and requires no additional attention layers.
- **Force-prediction auxiliary head.** A small network that reads the VLM prefix output and predicts the maximum safe grasping force for each episode. It is trained with a simple regression loss and adds negligible inference overhead.
- **Inference-time safety layer.** A real-time gripper safety mechanism that halts closing motion when live sensor readings exceed a threshold derived from the predicted max force.
- **Force-labelled grasping dataset.** A purpose-collected dataset of teleoperated grasping demonstrations on the SO-ARM101 arm, covering five force-sensitive everyday objects (orange, lime, grape, cucumber, egg), totalling 500 episodes, each frame contains 30 samples of force readings.

The remainder of this paper is organized as follows. Section 2 reviews related work on force-controlled grasping, force sensing for VLAs, and physically grounded language models. Section 3 describes the VLA backbone, force-sensing encoder, force-prediction head, our approach, and inference-time safety layer. Section 3.7 details dataset collection, and model training. Section 4 presents experimental results and ablations. Section 5 concludes with discussion and limitations.

2 Related Work

Generalist robot policies trained on web-scale data have achieved strong open-vocabulary manipulation performance (Zitkovich et al., 2023; Kim et al., 2024; Black et al., 2024), but treat manipulation as a purely kinematic problem with no explicit modelling of contact forces or object fragility. Prior work has explored tactile feedback for slip detection and grasp stability prediction (Han et al., 2025). More recently, VLA frameworks have been augmented with contact sensing: Tactile-VLA (Huang et al., 2025) deeply fuses vision, language, action, and tactile signals with a hybrid position-force controller that adapts strategy based on tactile feedback. OmniVTLA (Cheng et al., 2025) aligns tactile representations semantically with vision and language to improve contact-rich grasping success rates; TLA (Hao et al., 2025) and ForceMimic (Liu et al., 2025) extend imitation learning with explicit force supervision. Concurrent work ForceVLA (Yu et al., 2025) introduces a force-aware Mixture-of-Experts module that integrates visual-language embeddings with real-time 6-axis force feedback; ForceVLA2 (Li et al., 2026) builds on this with force-based VLM prompts and a Cross-Scale MoE architecture for closed-loop force regulation. Our approach differs: rather than a MoE fusion module or 6-axis wrist sensor, we integrate a lightweight fingertip uniaxial sensor via a 1D-CNN encoder and an auxiliary force-limit regression head trained jointly with the action expert, pairing vision-grounded force prediction with a reactive inference-time safety layer.

3 Method

3.1 Hardware Setup

Robotic arm. We use the SO-ARM101, an open-source 6-DoF robotic arm from the *LeRobot* platform Cadene et al. (2024). To measure contact forces, we integrate Honeywell FMAMS-DXX025WC2C3 (FMA) micro-force sensors (0–25 N, 12-bit digital output) into both gripper fingers. The sensors are mounted flush with a 0.5 mm hard plastic layer to increase the contact surface, followed by tape and a silicone interface layer to distribute point loads and prevent sensor overload, as shown in Figure 1. An ATmega328P micro-controller polls the sensors via software I^2C and streams readings to the host at 200 Hz over USB-serial.

Two USB webcams (640×480, 30 fps) provide visual input: one mounted at the wrist for a close-up view of the gripper and object, and one fixed above the workspace for global scene context.

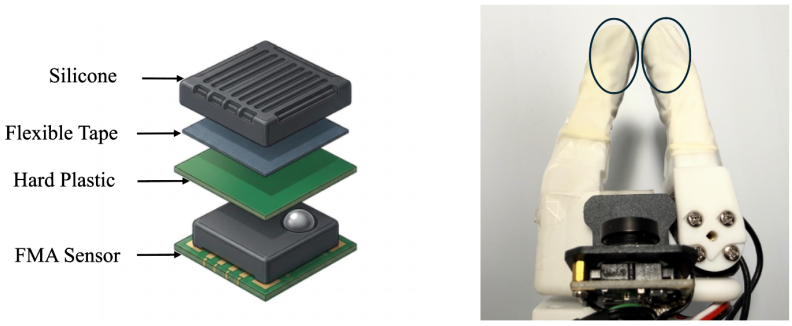


Figure 1: FMA fingertip force sensors integrated into the SO-ARM101 parallel gripper. (left) Layers of material used to place the sensor. (right) Sensor placement indicated by oval shape on the gripper fingers. To save sensors from liquid damage, we added a rubber finger protector on top of the sensor.

Both streams feed directly into the backbone ViT encoder. The system uses no depth sensor; force prediction is conditioned solely on the RGB observations and the language instruction.

The complete experimental setup, hardware stack, including component models and key specifications, is summarised in Appendix A, Appendix B (Table 6).

3.2 System Overview

Figure 2 shows the end-to-end architecture. At every control step, the robot receives an RGB image, a high-frequency window of fingertip force sensor samples, a natural-language task instruction, and a proprioceptive state vector. These are encoded by modality-specific encoders and concatenated as a prefix sequence fed into a shared VLM perceptual trunk (PaliGemma).

Our architecture splits the force awareness task into two paths: (i) **Sensory Input:** The raw force signal is encoded via a 1D-CNN and injected as tokens into the VLM prefix, conditioning the entire transformer stack on the live force state. (ii) **Perceptual Output:** A dedicated *force-prediction head* taps directly into the VLM prefix output. By leveraging the VLM's pre-trained visual-semantic reasoning, this head predicts the maximum safe grasping force $\hat{F}_{\max}$ for the object in view, effectively anticipating physical limits from vision alone even before contact is made.

Downstream, the *flow-matching action expert* attends to the VLM's perceptual features to generate a chunk of future motor commands. At execution time, a *reactive safety layer* intercepts the gripper command and freezes its position if the live sensor reading exceeds the predicted threshold $\hat{F}_{\max}$.

3.3 Background: Flow-Matching VLA Backbone

We instantiate our framework on π_0 (Black et al., 2024), which uses a pretrained vision-language model (VLM) as a semantic backbone and a separate *action expert* trained with flow matching (Lipman et al., 2023) to generate continuous action chunks at high control frequency. The two modules we introduce (force encoder and force-prediction head) attach to the VLM prefix stream (perceptual trunk).

Dual-stream architecture. The backbone processes inputs through two parallel token streams at every transformer layer. The *prefix* stream (PaliGemma) carries perceptual and linguistic context; the *suffix* stream (Gemma action expert) carries the proprioceptive state and the noisy action trajectory. All suffix tokens attend to the entire prefix, but prefix tokens are masked from attending to the suffix, so the VLM cannot "see" the denoising state.

Prefix: images, force, and language. Let $o_t \in \mathbb{R}^{C\times H_{\text{im}}\times W_{\text{im}}}$ be the RGB image, $\mathbf{f}_t \in \mathbb{R}^{N_s}$ the normalised force window (N_s=30 samples, pre-divided by F_{sensor}=25 N), and ℓ the language instruction. Image encoder ψ_{vis}, force encoder ψ_{force} (Section 3.4), and language embedding ψ_{lang}

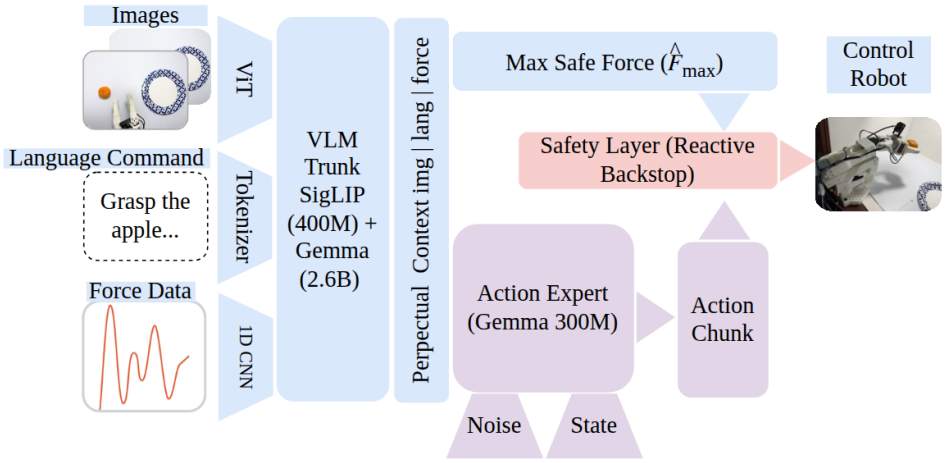


Figure 2: System architecture. **Prefix stream** (VLM Trunk): RGB images, force signals (via 1D-CNN), and language are fused into the VLM perceptual trunk. A dedicated head taps the prefix output to predict $\hat{F}_{\max}$ ($\mathcal{L}_{\text{force}}$), enabling visual anticipation of physical limits. **Suffix stream** (Action Expert): The action expert attends to the VLM's perceptual features to generate high-frequency action chunks $\hat{\mathbf{a}}_{1:H}$ ($\mathcal{L}_{\text{FM}}$). **Safety Layer**: A reactive backstop intercepts motor commands and freezes the gripper if live force sensor readings exceed the predicted threshold.

produce token sequences concatenated in order to form the prefix:

$$\mathbf{Z}_{\text{pre}} = \left[\, \psi_{\text{vis}}(o_t) \parallel \psi_{\text{force}}(\mathbf{f}_t) \parallel \psi_{\text{lang}}(\ell) \,\right] \in \mathbb{R}^{(N_I+N_f+N_L)\times d_v} \tag{1}$$

where N_I, N_f=4, N_L are image, force, and language token counts, and d_v=2048 is the VLM hidden width. All prefix tokens attend bidirectionally. When the force encoder is disabled the ψ_{force} column is omitted.

Suffix: state and noisy actions. The suffix tokens concatenate a proprioceptive state embedding and H noisy-action embeddings (Black et al., 2024):

$$\mathbf{Z}_{\text{suf}} = \left[\, W_s s_t \parallel \mathbf{E}(\mathbf{a}_t, t) \,\right] \in \mathbb{R}^{(1+H)\times d_{\text{exp}}} \tag{2}$$

where $s_t \in \mathbb{R}^{d_s}$ is the proprioceptive state, $W_s \in \mathbb{R}^{d_{\text{exp}}\times d_s}$ is a linear projector, $\mathbf{a}_t$ is the noisy action at diffusion time t, $\mathbf{E}(\mathbf{a}_t, t)$ projects to d_{exp} dimensions and adds a sinusoidal time encoding, H is the action chunk horizon, and d_{exp}=1024.

Flow-matching training objective. Following optimal-transport conditional flow matching (Lipman et al., 2023; Black et al., 2024), the noisy action is the linear interpolant $\mathbf{a}_t = (1-t)\,\mathbf{a}_0 + t\,\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $t \sim \text{Beta}(1.5, 1.0)$. The action expert v_θ is trained to regress the target vector field $u_t = \boldsymbol{\epsilon} - \mathbf{a}_0$:

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t,\,\mathbf{a}_0\sim\mathcal{D},\,\boldsymbol{\epsilon}}\left[\, \left\| v_\theta\big(\mathbf{a}_t,\, t \mid \mathbf{Z}_{\text{pre}},\, s_t\big) - u_t \right\|^2 \right] \tag{3}$$

where θ denotes the trainable parameters of the action expert, $\mathbf{a}_0 \in \mathbb{R}^{H\times d_a}$ is a clean action chunk (d_a=7: six joint angles plus gripper width), $\mathcal{D}$ is the demonstration dataset, and $\mathbf{Z}_{\text{pre}}$ is the prefix context from Eq. 1. At inference, actions are recovered by integrating v_θ from t=1 to t=0 with a 10-step Euler solver.

3.4 Force-Sensing Encoder

Let $\mathbf{f}_t \in \mathbb{R}^{N_s}$ denote the window of N_s=30 raw force readings captured at 200 Hz and normalised to $[0, 1]$ by dividing by the sensor ceiling F_{sensor}=25 N. We encode this signal with a two-layer 1D convolutional network (same-padding throughout) followed by adaptive pooling and a linear up-projection to the VLM hidden dimension d_v. The signal $\mathbf{f}_t$ is treated as a single-channel temporal

sequence of length N_s:

$$\mathbf{h}^{(1)} = \mathrm{ReLU}\big(\mathrm{Conv1d}_{1\to 32,\ k=5,\ \mathrm{same}}(\mathbf{f}_t)\big) \ \in \mathbb{R}^{32\times N_s}, \tag{4}$$

$$\mathbf{h}^{(2)} = \mathrm{ReLU}\big(\mathrm{Conv1d}_{32\to C_h,\ k=3,\ \mathrm{same}}(\mathbf{h}^{(1)})\big) \ \in \mathbb{R}^{C_h\times N_s}, \tag{5}$$

with C_h=64 hidden channels. Adaptive max-pooling then compresses the temporal axis to N_f=4 slots,

$$\mathbf{P} = \mathrm{AdaptiveMaxPool1d}_{N_f}(\mathbf{h}^{(2)})^\top \in \mathbb{R}^{N_f\times C_h}, \tag{6}$$

and a layer-normalised linear projection lifts each slot to the VLM width:

$$\psi_{\text{force}}(\mathbf{f}_t) = \mathrm{LN}_{d_v}\big(\mathrm{LN}_{C_h}(\mathbf{P})\, W_{\text{up}}^\top\big) \ \in \mathbb{R}^{N_f\times d_v} \tag{7}$$

where $W_{\text{up}} \in \mathbb{R}^{d_v\times C_h}$ and LN denotes Layer Normalisation. The resulting N_f force tokens are concatenated into the VLM prefix between the image tokens and language tokens and participate in full bidirectional attention, allowing every transformer layer to condition on the current force signal.

3.5 Force-Prediction Head

Unlike the action expert which generates trajectories via iterative denoising, the force-prediction head f_ϕ provides a direct, low-latency estimate of the maximum safe grasping force by tapping into the VLM's perceptual trunk. By reading from the *VLM prefix output* rather than the action expert, the head leverages the model's pre-trained visual-semantic knowledge to anticipate an object's physical properties (*e.g.*, fragility) from vision alone, even when the current force signal $\mathbf{f}_t$ is zero.

Concretely, let $\mathbf{h}^{\text{vlm}} \in \mathbb{R}^{(N_I+N_f+N_L)\times d_v}$ be the prefix token embeddings produced by the VLM after its final transformer layer. We mean-pool across all prefix tokens to obtain a fixed-size feature vector that summarizes the visual, force, and linguistic context:

$$\bar{\mathbf{h}}_{\text{pre}} = \frac{1}{N_I+N_f+N_L}\sum_{j=1}^{N_I+N_f+N_L} \mathbf{h}_j^{\text{vlm}} \ \in \mathbb{R}^{d_v} \tag{8}$$

A two-layer MLP with sigmoid output then predicts the normalised maximum force:

$$\hat{F}_{\max} = F_{\text{sensor}} \cdot \sigma\big(W_2\,\mathrm{ReLU}(W_1\bar{\mathbf{h}}_{\text{pre}} + b_1) + b_2\big) \ \in (0,\ F_{\text{sensor}}) \tag{9}$$

where σ is the sigmoid function, F_{sensor}=25 N rescales the $[0,1]$ output to physical units, $W_1 \in \mathbb{R}^{d_h\times d_v}$, $b_1 \in \mathbb{R}^{d_h}$ $W_2 \in \mathbb{R}^{1\times d_h}$, $b_2 \in \mathbb{R}$ and d_h=256. This architectural placement ensures that $\hat{F}_{\max}$ is invariant to the stochastic denoising process used for actions, providing a stable safety threshold optimized for visual grounding. During inference, we execute only the VLM prefix pass to retrieve $\hat{F}_{\max}$, drastically reducing the latency of the safety layer compared to full action generation.

The output $\hat{F}_{\max}$ represents the *maximum safe grasping force* in Newtons for the object class currently visible. It is a scalar, which allows the safety layer to operate independently of the 6-DoF action dimension.

3.6 Joint Training Objective

Let $\mathcal{D} = \{(o^{(i)}, \ell^{(i)}, s^{(i)}, \mathbf{f}_t^{(i)}, \mathbf{a}_0^{(i)}, F_{\max}^{*(i)})\}_{i=1}^N$ be the training set, where $F_{\max}^{*(i)} \in \mathbb{R}_{>0}$ is the per-episode maximum safe grasping force label calculated post-hoc (Section 3.7). The force-regression loss is the mean squared error (MSE) in normalised force space:

$$\mathcal{L}_{\text{force}}(\phi) = \mathbb{E}_i\left[\left(\frac{f_\phi(\bar{\mathbf{h}}_{\text{pre}}^{(i)})}{F_{\text{sensor}}} - \frac{F_{\max}^{*(i)}}{F_{\text{sensor}}}\right)^2\right] \tag{10}$$

The total loss is the weighted sum of the flow-matching action loss and the auxiliary force-head loss:

$$\mathcal{L}(\theta,\phi) = \mathcal{L}_{\text{FM}}(\theta) + \lambda_{\text{aux}}\,\mathcal{L}_{\text{force}}(\phi) \tag{11}$$

where $\mathcal{L}_{\text{FM}}$ (Eq. 3) supervises the action expert and $\mathcal{L}_{\text{force}}$ supervises the auxiliary force head. We use λ_{aux}=0.1 to balance the high-frequency motor control task with the scalar fragility estimation task. All parameters (the VLM prefix, force encoder, action expert θ, and force head ϕ) are fine-tuned jointly via AdamW with a cosine learning-rate schedule, so $\mathcal{L}_{\text{force}}$ also shapes the prefix representations.

3.6.1 INFERENCE-TIME FORCE SAFETY LAYER.

At each control step the safety layer intercepts the gripper command and freezes closing motion when the live sensor reading F_{live} exceeds the safety threshold $\tau_{\text{safe}} = \hat{F}_{\max}$, the maximum safe grasping force predicted by the force head:

$$g'_{\text{cmd}} = \begin{cases} g_{\text{prev}} & \text{if } F_{\text{live}} \geq \tau_{\text{safe}} \text{ and } g_{\text{cmd}} < g_{\text{prev}}, \\ g_{\text{cmd}} & \text{otherwise.} \end{cases} \tag{12}$$

Opening commands are never restricted. Because $\hat{F}_{\max}$ is predicted from vision before contact, the threshold is object-aware rather than a fixed constant. Figure 4 shows the resulting peak force comparison.

3.7 DATA COLLECTION

Uniaxial fingertip sensors on the SO-ARM101 gripper log at 200 Hz (0–25 N range) alongside 30 fps RGB from wrist and overhead cameras during teleoperated leader–follower demonstrations, both time-stamped to the nearest frame. Each episode is a single grasp-and-lift attempt over five object classes (cucumber, egg, grape, lime, orange; dimensions in Appendix C). We collect 100 episodes per class (500 total, 129,822 frames), each frame paired with a 30-sample force window for a dense, vision-aligned force record with 10 second episode length.

Ground-truth safe force label. $F_{\max}^{*(i)} = 0.95 \cdot \max_t F_t^{(i)}$: 95% of episode i's peak two-fingertip element-wise-max force (the same signal as encoder input $\mathbf{f}_t$ and safety-layer reading F_{live}), with the 5% margin guarding against sensor spikes.

Training, validation, and test protocol. All 500 teleoperated demonstrations (129,822 force-labelled frames) are used for training; given the cost of real-world data collection, no portion of this offline dataset is held out as a validation split. Instead, model quality is assessed exclusively through live, on-robot evaluation: every trial reported in Section 4 is a fresh rollout on the physical SO-ARM101 arm, procedurally distinct from any training demonstration, rather than a held-out sample of the training set.

4 EXPERIMENTS

In this section, we evaluate the effectiveness of our approach on contact-critical manipulation tasks involving force-sensitive and deformable objects. These tasks test the model's ability to extract physical commonsense from visual and linguistic cues, as well as its capacity to preemptively regulate grip forces using a standard position-controlled robotic arm. Specifically, our experiments are designed to address the following research questions:

- Q1: Does introduction of force awareness into VLA reduce force applied to the object without reducing reliability?
- Q2: What is the relative contribution of the force encoder versus the force-prediction head, and can we safely emulate force control on standard position-controlled hardware?
- Q3: To what extent does the VLA backbone implicitly encode physical commonsense, enabling zero-shot force limit prediction for unseen objects?

Each model is evaluated over **20 trials per object** on the real robot; a trial succeeds if the object is lifted and placed on the plate, and retries are excluded to avoid biasing the success rate. In total we conduct **1000 real-robot evaluations** across **8 object classes** (five in-distribution, three OOD): **500** in-distribution trials (Tables 1, 3: 5 models $\times$ 5 objects $\times$ 20 trials, comparing FE-VLA against π_0-base, $\pi_{0.5}$-base, ForceVLA, and Tactile-VLA), **200** ablation trials (Table 4: 2 additional FE-VLA configurations $\times$ 5 objects $\times$ 20 trials), and **300** OOD trials (Table 5: 5 models $\times$ 3 unseen objects $\times$ 20 trials).

4.1 Primary Result: Grasp Success and Force Reduction (Q1)

Table 1 reports success rate (SR, %) per object class. FE-VLA achieves the highest average SR overall (85.00, vs. 75.00/74.00/76.00/70.00 for π_0-base/$\pi_{0.5}$-base/ForceVLA/Tactile-VLA). The largest per-object gaps are on Egg and Grape (30 pp each): $\pi_{0.5}$-base fails 45% of Egg grasps while ours reduces this to 15%, and ForceVLA succeeds only 60% of Grape grasps while ours reaches 90%. The one exception is Cucumber, where ForceVLA edges ahead of FE-VLA (90.00 vs. 85.00).

Table 1: Success rate (SR%) for each object class (20 trials each). Best result per column in bold.

Model	Cucumber	Egg	Grape	Lime	Orange	Average
π_0-base	65.00	75.00	75.00	80.00	80.00	75.00
$\pi_{0.5}$-base	75.00	55.00	75.00	80.00	**85.00**	74.00
ForceVLA	**90.00**	65.00	60.00	**85.00**	80.00	76.00
Tactile-VLA	65.00	65.00	70.00	80.00	70.00	70.00
FE-VLA	85.00	**85.00**	**90.00**	**85.00**	80.00	**85.00**

4.2 Force Prediction Accuracy

Table 2 reports the per-class mean absolute error (MAE) and root mean squared error (RMSE) of the predicted maximum safe grasping force ($\hat{F}_{\max}$) against ground-truth limits. The force-prediction head achieves the lowest error on grapes (MAE = 0.73 N) and oranges (MAE = 0.83 N), whose consistent shape and texture produce reliable visual cues. Error is higher for eggs (MAE = 2.33 N) and limes (MAE = 2.26 N), where surface reflectance and size variability make force limits harder to estimate; nevertheless, the predictions remain within the operational margin required for the safety layer to prevent catastrophic crushing.

Table 2: Force prediction accuracy of the force-prediction head on live evaluation rollouts. Ground Truth is the mean $\pm$ std of the per-episode labels $F^*_{\max}$ in the training set; Predicted is the mean $\pm$ std of $\hat{F}_{\max}$ over rollouts, and MAE/RMSE are measured against the ground-truth mean. Lower is better for MAE and RMSE (20 trials for each object).

Object	Ground Truth (N)	Predicted (N)	MAE (N)	RMSE (N)
Egg	6.22 ± 2.07	4.18 ± 0.88	2.33	2.75
Lime	8.88 ± 3.24	8.06 ± 1.07	2.26	3.05
Orange	4.76 ± 1.58	5.07 ± 0.33	0.83	1.04
Cucumber	8.10 ± 2.80	7.01 ± 0.70	1.89	2.35
Grape	2.47 ± 1.34	1.99 ± 0.26	0.73	0.88

Averaged across the five in-distribution objects, FE-VLA applies substantially less peak force than every baseline: **49.98%** less than π_0 (2.79 N vs. 5.57 N), **50.11%** less than $\pi_{0.5}$ (2.79 N vs. 5.58 N), **48.40%** less than ForceVLA (2.79 N vs. 5.40 N), and **52.02%** less than Tactile-VLA (2.79 N vs. 5.81 N). Per-object peak force distributions are shown in Figure 4 (Appendix G).

4.3 Ablation Study

Ablation setup. We ablate the two main components: the force-sensing encoder and the force-prediction head. We use a 2×2 factorial design, starting from the π_0 baseline with neither component active. Each configuration is evaluated on all five in-distribution objects (20 trials each), isolating the contribution of each component to overall performance.

Component ablation (Q2). Table 4 isolates the contribution of each component. Adding the force-prediction head alone (without the encoder) degrades SR by 7 pp relative to the π_0 baseline (75.00 → 68.00), while adding the force encoder alone improves SR by 4 pp (75.00 → 79.00). The full combination of both components performs best, reaching 85.00 (+10 pp).

Table 3: Force Prediction Comparison: Mean peak applied force (N) per object class and average (20 trials each).

Model	Egg	Lime	Orange	Cucumber	Grape	Average
π_0-base	4.71	7.49	4.60	8.56	2.49	5.57
$\pi_{0.5}$-base	4.60	6.73	6.78	7.07	2.74	5.58
FE-VLA (Ours)	**1.92**	**2.57**	**3.22**	**4.50**	**1.72**	**2.79**
ForceVLA	6.82	6.58	6.01	5.54	2.04	5.40
Tactile-VLA	6.10	6.53	6.14	7.67	2.59	5.81

Table 4: Component ablation: success rate (%) per object (20 trials each). ✓ = enabled; × = disabled.

Encoder	Force Head	Cucumber	Egg	Grape	Lime	Orange	Avg
×	×	65.00	75.00	75.00	80.00	80.00	75.00
×	✓	60.00	70.00	70.00	65.00	75.00	68.00
✓	×	75.00	75.00	85.00	85.00	75.00	79.00
✓	✓	85.00	85.00	90.00	85.00	80.00	85.00

Ablation shows an asymmetric contribution: the encoder alone surpasses both π_0 (79.00, +4 pp) and ForceVLA (76.00, +3 pp), while the head alone underperforms both (68.00; −7/−8 pp). Only the full combination reaches 85.00 (+10 pp vs. π_0, +9 pp vs. ForceVLA), a 6 pp gain over encoder-only, despite ForceVLA's richer 6-axis wrist sensing. This confirms the encoder is the load-bearing component: the head's auxiliary loss $\mathcal{L}_{\text{force}}$ only enriches the prefix representations when grounded by the encoder's live force tokens; without them it has no signal to regress against and instead hurts performance.

4.4 Generalisation to Unseen Objects (OOD), Q3

We test zero-shot generalisation to three object classes not seen during training: kiwi, lemon, and plum. Held out from training entirely; only the model's visual and language understanding is used for prediction.

Table 5: Grasp success rate (%) on out-of-distribution (OOD) object classes (20 trials each). Models are trained only on the 5 in-distribution objects. Best result per column in bold.

Model	Kiwi	Lemon	Plum	OOD Avg
π_0-base	65.00	60.00	**70.00**	65.00
$\pi_{0.5}$-base	65.00	65.00	65.00	65.00
ForceVLA	70.00	60.00	**70.00**	66.67
Tactile-VLA	45.00	50.00	65.00	53.33
FE-VLA	**85.00**	**75.00**	65.00	**75.00**

FE-VLA generalises to unseen objects, outperforming all four baselines on kiwi (+15–40 pp) and lemon (+10–25 pp). On plum, FE-VLA matches $\pi_{0.5}$-base and Tactile-VLA at 65%, while π_0-base and ForceVLA edge slightly ahead at 70%, suggesting that the rounder, less texturally distinct surface makes grasping harder to generalise regardless of force conditioning.

5 Discussion

Our results establish that a force encoder and lightweight auxiliary head appended to a VLA backbone can reliably predict per-object safe force limits without degrading manipulation performance. The ablation study (Table 4) shows that the two components contribute differently in isolation. The force encoder alone already yields a net benefit, surpassing the π_0 baseline by 4 pp (79.00 vs. 75.00), indicating that live force context is independently useful for action conditioning. The force-prediction

head alone, however, underperforms the baseline by 7 pp (68.00), confirming that the safety threshold is unreliable without the encoder's force grounding. Only the full combination achieves the largest gain (+10 pp, 85.00), a 6 pp improvement over the encoder-only configuration. This asymmetry suggests that the force encoder is the load-bearing component, while the force-prediction head amplifies performance only when paired with live force input. The same asymmetry holds against ForceVLA (76.00 average), a concurrent baseline with richer 6-axis wrist force-torque sensing: the encoder-only ablation modestly exceeds it (79.00, $+3$ pp), the head-only ablation trails it (-8 pp), and the full FE-VLA configuration surpasses it by 9 pp, suggesting that *how* force is fused into the policy matters more than the sensor's axis count.

A secondary finding is the OOD generalization result (Table 5): FE-VLA achieves an average success rate of 75.00% on held-out objects, a 10 pp improvement over both π_0 and $\pi_{0.5}$ baselines, 8.33 pp over ForceVLA, and 21.67 pp over Tactile-VLA, despite never observing these objects during training. This suggests that the VLM backbone already encodes material and deformability cues that transfer across visual appearance changes.

These noise sources also appear in the force-prediction results (Table 2): error tracks the ground-truth failure point's std, with eggs (6.22 ± 2.07 N) and limes (8.88 ± 3.24 N) giving the highest MAEs (2.33 N, 2.26 N), pointing to label noise rather than model capacity. The uniaxial FMA sensor compounds this, resolving only normal compression and forcing a single scalar onto multi-dimensional failure modes. The force encoder adds negligible latency (four tokens; Appendix E).

5.1 Limitations and Future work

Variance in our force data stems primarily from the inherent heterogeneity of biological specimens (produce and eggs), which makes it difficult to maintain consistent structural properties across batches (see Table 7); even slight variations in maturity or surface texture significantly shift the failure point.

The point-contact sensors on the fingertips can miss contact on uneven or large-radius surfaces where the object engages the finger pad away from the ball actuators. While our approach is robust for the five studied object classes (which share a top-down power-grasp mode), extending it to asymmetric or packaged goods remains future work. On the hardware side, the shared ATmega328P bit-banged I^2C bus introduces timing jitter and halves the sampling rate; a hardware I^2C peripheral would allow sampling up to the sensor's 1 kHz limit for improved reactive control.

Deploying multiple sensors or a 3/6-axis device would improve contact coverage and capture shear forces that the current uniaxial setup cannot measure. We intend to explore multi-axis force sensing to resolve the visual-tactile ambiguities identified in Table 2. Finally, we plan to scale the protocol to a more diverse object set, including electronic components and more fragile items, to further generalize the model.

6 Conclusion

We presented FE-VLA, a framework that makes Vision-Language-Action models force-aware through two lightweight additions to the VLA prefix stream: a 1D-CNN force encoder that injects live fingertip readings as tokens, and an auxiliary force-prediction head that regresses per-object safe force limits directly from VLM perceptual features. A reactive safety layer uses the predicted limit to freeze the gripper before damaging contact occurs, without modifying the underlying action expert.

Across five force-sensitive objects, FE-VLA substantially reduces mean peak applied force relative to the baselines while also achieving the highest average grasp success. Ablation reveals an asymmetric contribution: the force encoder alone already surpasses the π_0 baseline by 4 pp (79.00 vs. 75.00), whereas the force-prediction head alone degrades performance by 7 pp (68.00), since its auxiliary loss only helps when grounded by live force tokens. Their combination yields the full +10 pp gain, also surpassing ForceVLA (a concurrent baseline with richer 6-axis wrist sensing) by 9 pp on average. Zero-shot evaluation on three held-out classes shows an average improvement of 8.33–21.67 pp over all four baselines, indicating the VLM backbone transfers physical commonsense across appearance changes without retraining, demonstrating that force-safe manipulation is achievable on standard position-controlled hardware with low-cost uniaxial MEMS sensors. We hope FE-VLA will be helpful for the broader research community as a simple approach.

7 AI USE STATEMENT

We used a generative AI coding assistant to aid and polish the writing of the manuscript text. The assistant did not generate or fabricate any experimental results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

8 ETHICS STATEMENT

This work does not raise concerns under the ICLR Code of Ethics. Data collection involved only the authors operating a teleoperated robotic arm on inert produce and eggs; no human-subjects research, personal data, or animal-welfare protocols were involved. The force-sensing and safety-layer contributions are explicitly designed to reduce, rather than cause, physical harm during robotic manipulation.

9 REPRODUCIBILITY STATEMENT

We provide the details needed to reproduce our results throughout the paper: the hardware and sensor stack in Appendix B (Table 6), the data collection protocol and ground-truth force-label derivation in Section 3.7, the model architecture and training objective in Sections 3–3.6, and the full training hyperparameters in Appendix D (Table 8). Object physical dimensions for our five in-distribution and three OOD classes are listed in Appendix C (Table 7). Code and the FE-VLA dataset will be released at the project page below: https://fevla2026.github.io.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. URL https://physicalintelligence.company/blog/pi0.

Remi Cadene, Simon Alibert, Alexander Soare, Quentin Gallouedec, Adil Zouitine, and Thomas Wolf. LeRobot: Making robotics more accessible with end-to-end learning. https://github.com/huggingface/lerobot, 2024. Hugging Face open-source robot learning framework. SO-ARM101 hardware design by The Robot Studio.

Zhengxue Cheng, Yiqian Zhang, Wenkang Zhang, Haoyu Li, Keyu Wang, Li Song, and Hengdi Zhang. Omnivtla: Vision-tactile-language-action model with semantic-aligned tactile sensing. 2025. URL https://arxiv.org/abs/2508.08706.

Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. 2024. URL https://arxiv.org/abs/2309.02561.

Yunhai Han, Rahul Batra, Nathan Boyd, Chaitanya Mehta, Tuo Zhao, Yu She, and Seth Hutchinson. Learning generalizable vision-tactile robotic grasping strategy for deformable objects via transformer. *IEEE/ASME Transactions on Mechatronics*, 30(1):554–561, 2025. doi: 10.1109/TMECH.2024.3400789.

Peng Hao, Chaofan Zhang, Dingzhe Li, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. Tla: Tactile-language-action model for contact-rich manipulation. 2025. URL https://arxiv.org/abs/2503.08548.

Jialei Huang, Shuo Wang, Fanqi Lin, Yihang Hu, Chuan Wen, and Yang Gao. Tactile-vla: Unlocking vision-language-action model's physical knowledge for tactile generalization. 2025. URL https://arxiv.org/abs/2507.09160.

Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. VoxPoser: Composable 3D value maps for robotic manipulation with language models. In *Conference on Robot Learning (CoRL)*, 2023.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. OpenVLA: An open-source vision-language-action model. 2024.

Yang Li, Zhaxizhuoma, Hongru Jiang, Junjie Xia, Hongquan Zhang, Jinda Du, Yunsong Zhou, Jia Zeng, Ce Hao, Jieji Ren, Qiaojun Yu, Cewu Lu, Yu Qiao, and Jiangmiao Pang. Forcevla2: Unleashing hybrid force-position control with force awareness for contact-rich manipulation, 2026. URL https://arxiv.org/abs/2603.15169.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. 2023. URL https://arxiv.org/abs/2210.02747.

Wenhai Liu, Junbo Wang, Yiming Wang, Weiming Wang, and Cewu Lu. Forcemimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation. 2025.

Junjie Wen, Yichen Zhu, Jinming Li, Minjie Zhu, Kun Wu, Zhiyuan Xu, Ran Cheng, Chaomin Shen, Yaxin Peng, Feifei Feng, et al. Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation. *arXiv preprint arXiv:2409.12514*, 2024.

Junjie Wen, Minjie Zhu, Yichen Zhu, Zhibin Tang, Jinming Li, Zhongyi Zhou, Chengmeng Li, Xiaoyu Liu, Yaxin Peng, Chaomin Shen, and Feifei Feng. Diffusion-vla: Generalizable and interpretable robot foundation model via self-generated reasoning. 2025. URL https://arxiv.org/abs/2412.03293.

Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, Cewu Lu, and Wenqiang Zhang. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. 2025. URL https://arxiv.org/abs/2505.22159.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183, 2023. URL https://openreview.net/forum?id=XMQgwiJ7KSX.

APPENDIX

A EXPERIMENTAL SETUP

Our experimental setup features the SO-ARM101 robotic arm positioned in a tabletop environment, designed for contact-rich manipulation tasks. As shown in Figure 3, two USB webcams provide the necessary visual context: an overhead camera captures the global scene layout, while a wrist-mounted camera offers a close-up view of the gripper and the target object. The dual FMA micro-force sensors are integrated directly into the fingertips, capturing real-time compressive forces during object interaction. This setup allows the FE-VLA policy to seamlessly fuse multi-view visual features with high-frequency tactile feedback to safely grasp fragile items.

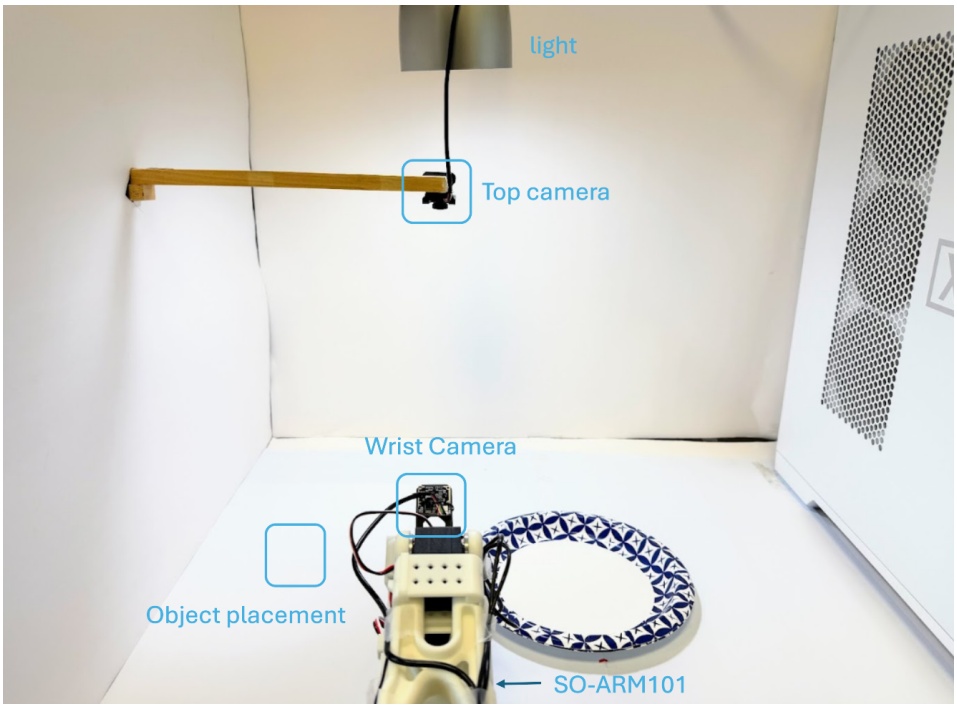


Figure 3: Experimental setup showing the SO-ARM101 robotic arm, the overhead and wrist-mounted cameras, and the tabletop workspace used for force-sensitive grasping tasks.

B Hardware Stack

All components interface to a host workstation (NVIDIA A800, Ubuntu 22.04) over USB. Joint angles from the SO-ARM101 bus and force readings from the ATmega328P sensor node are time-stamped and merged at 50 Hz for policy inference.

Table 6: Complete hardware stack for the force-augmented VLA system.

Component	Model	Key specification
Robotic arm	SO-ARM101 (LeRobot Cadene et al. (2024))	6-DoF, 30 cm reach
Servo motors	Feetech STS3215 $\times 7$	19 kg cm, 1 Mbaud bus
Force sensors	FMAMSDXX025WC2C3	0–25 N, I^2C, 200 Hz
Sensor MCU	ATmega328P (16 MHz)	5 V, software I^2C, USB-serial
RGB camera	USB webcam	640×480, 30 fps
Host GPU	NVIDIA A800	Policy inference + training
OS / framework	Ubuntu 22.04	

C Dataset Description: Object Physical Dimensions

We collect data across five force-sensitive object classes chosen to span a broad range of fragility, deformability, and surface texture: **egg** (brittle shell, highly fragile), **grape** (thin skin, very soft), **lime** (firm citrus, moderate compliance), **orange** (thick peel, moderate compliance), and **cucumber** (elongated, relatively rigid). These five objects constitute the *training set*; 100 teleoperated episodes are collected per object class (500 episodes total).

To assess out-of-distribution (OOD) generalisation, we additionally evaluate on three held-out object classes that were never seen during training: **plum**, **kiwi**, and **lemon**. These objects were selected to probe different combinations of size, mass, and surface compliance not well represented by the

training set. Table 7 reports the physical dimensions and mass of all eight objects, measured with digital calipers across multiple samples. Values are reported as mean ± std.

Table 7: Physical dimensions and mass of grasped objects (mean ± std). Length refers to the longest horizontal extent; Height refers to the vertical extent when resting in natural orientation.

Object	Length (mm)	Height (mm)	Weight (g)
Lime	58.33 ± 4.93	47.00 ± 2.00	65.16 ± 7.29
Cucumber	125.00 ± 3.61	29.33 ± 2.52	69.10 ± 8.65
Orange	40.50 ± 1.00	51.25 ± 1.50	59.95 ± 2.65
Egg	54.75 ± 0.96	44.25 ± 0.96	56.08 ± 2.42
Grape	33.50 ± 1.29	22.25 ± 1.26	11.13 ± 1.37
Plum	46.67 ± 2.08	52.33 ± 1.53	82.93 ± 4.37
Kiwi	73.50 ± 2.12	52.50 ± 0.71	106.60 ± 1.13
Lemon	60.25 ± 2.75	50.25 ± 0.96	75.45 ± 4.07

D Training Hyperparameters

Due to the cost of real-world data collection, all 500 demonstrations were used for training, with no held-out validation split (see Section 3.7). Policy performance was instead evaluated through independent, on-robot rollout trials under randomized initial conditions (1000 real-robot evaluations in total; see Section 4). Table 8 lists the hyperparameters used for all training runs reported in this paper.

Table 8: Training hyperparameters.

Parameter	Value
Optimizer	AdamW
Learning Rate (peak)	2.5×10^{-5}
Learning Rate (decay end)	2.5×10^{-6}
LR Schedule	Cosine Decay with Warmup
Warmup Steps	500
AdamW β_1, β_2	0.9, 0.95
AdamW ϵ	1×10^{-8}
Weight Decay	0.01
Gradient Clip Norm	1.0
Batch Size	4
Training Steps	60,000
Precision	bfloat16
Gradient Checkpointing	enabled
Random Seed	42

E Inference Time

The force safety layer itself is a numerical comparison between the predicted force and the current sensor reading, so its cost is that of executing a single statement. We measured the remaining components directly (Table 9).

The two modules we add are negligible in themselves: a forward pass through the force encoder takes 0.065 ms and through the force head 0.029 ms, together under 0.1 ms. The 31.7 ms attributed to Force Head in Table 9 is dominated by the VLM prefix pass it requires, not by the head itself.

Table 9: Per-component inference latency, measured over two runs.

Metric	Run 1	Run 2
Action chunk	104.32 ms	102.72 ms
Force head	31.72 ms	31.64 ms
Force encoder forward	0.065 ms	0.065 ms
Model ms/step	33.81	33.69

F Safety Layer Advantages

The safety layer (Section 3.6.1, Eq. 12) compares the live sensor reading against the force head's predicted threshold $\hat{F}_{\max}$; without the force-prediction head there is no threshold to compare against, so the safety layer is effectively disabled. To check how often this reactive cap actually intervenes during normal operation, rather than existing only as an unused backstop, we examined every successful in-distribution FE-VLA trial (encoder and safety head both enabled) for whether and how often it fired.

Table 10: Reactive safety layer activation on successful in-distribution FE-VLA trials.

Object	Success	Success eps w/ safety	% eps activated	Mean acts/episode
Cucumber	17/20 (85.0%)	12/17	70.6%	8.71
Egg	17/20 (85.0%)	4/17	23.5%	3.47
Grape	18/20 (90.0%)	13/18	72.2%	16.11
Lime	17/20 (85.0%)	7/17	41.2%	2.53
Orange	16/20 (80.0%)	6/16	37.5%	1.06

Overall, 42 of the 85 successful episodes (49.4%) triggered the reactive safety layer at least once, with wide per-object variation, from 23.5% of episodes (Egg) to 72.2% (Grape), and activation counts ranging from roughly 1 to 16 per episode. The safety layer is therefore not a rarely-used formality: it actively intervenes in close to half of all successful grasps, most often on objects requiring sustained or high-force contact (Cucumber, Grape).

G Result: Peak Force Comparison

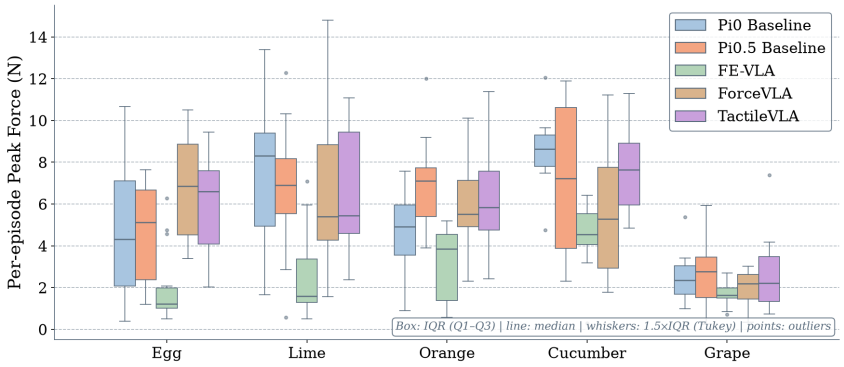


Figure 4: Per-episode peak applied force by model and object. Boxplots summarize per-trial peak forces over the successful trials of each condition (out of 20 trials per object). Boxes show the interquartile range (Q1–Q3); the centre line is the median; whiskers extend to 1.5×IQR (Tukey definition); points denote outliers beyond the whiskers. FE-VLA (green) shifts the distribution downward relative to all four baselines (π_0: blue, $\pi_{0.5}$: orange, ForceVLA: tan, Tactile-VLA: purple) on every object, reducing both the median and the spread.