CVLAMP: Compliant Vision Language Action Movement Primitives

Vamsi Krishna Origanti[1,2], Alexander Fabisch[1], Frank Kirchner[1,2]

IEEE/RSJ IROS 2026
Pittsburgh, PA, USA · Sept 27–Oct 1, 2026
Late Breaking Results Poster Session
IROS 2026 PITTSBURGH

Motivation & Problem Statement

Contact-rich, dual-arm manipulation demands synchronized, force-aware motion — but today's VLA policies regress dense, low-level action chunks with no explicit notion of contact or compliance, leaving safety and interpretability implicit exactly where they matter most.

- Contact-rich manipulation needs force-aware motion, not just accurate poses.
- Dense action-chunk policies are hard to interpret and unstable under contact uncertainty.
- The dual-arm IMRK/KUKA iiwa platform motivates synchronized pose, wrench, and gripper control; this work currently controls a single arm.
- A structured, low-dimensional primitive keeps the controller auditable — every commanded quantity has a physical meaning.

Key Contributions

- A hybrid VLA-primitive architecture fusing a PaliGemma/Gemma flow-matching backbone with Cartesian-state and force-history encoders;
- A synchronized pose/wrench/gripper primitive with an explicit phase governor for safe closed-loop execution;
- A composite training objective and staged, single-GPU adaptation recipe for a released π0.5 checkpoint;
- A complete ROS 2 implementation, validated in MuJoCo on the DFKI IMRK platform and designed for transfer to the real KUKA iiwa hardware;
- An explicit, auditable phase governor (RUN/SLOW/HOLD/ABORT) in place of a learned, opaque safety signal.

Related Work

Vision-language-action policies:

VLA policies such as $\pi_{0.5}$ [1] use a flow-matching action expert on a PaliGemma-style backbone [2], regressing low-level actions with no explicit contact representation. More recently, ForceVLA [3], FD-VLA [4], and FM-VLA [5] add force-awareness directly into a VLA backbone (a force-aware MoE, a distilled force token, and force-history memory).

Movement primitives and compliant control:

DMPs [6] and their Cartesian/orientation extensions [7] give compact, compliant motion representations suited to force-modulated execution; we use the movement_primitives library [8] directly for DMP fitting and rollout, but these approaches lack the semantic grounding of modern VLA models.

Multimodal contact-rich manipulation:

Lee et al. [9] learn joint vision-touch representations for contact-rich insertion via self-supervision, showing the value of fusing visual and force modalities. CVLAMP instead predicts an explicit, structured primitive with a rule-based governor, keeping closed-loop safety interpretable rather than latent.

Where CVLAMP sits:

PrimitiveVLA [10] sequences pre-extracted primitives via a separate VLM planner, and Language Movement Primitives [11] uses a VLM to set DMP parameters; CVLAMP instead regresses continuous primitive parameters directly from perception, end-to-end, with an explicit, rule-based phase governor rather than a dense force-conditioned action chunk.

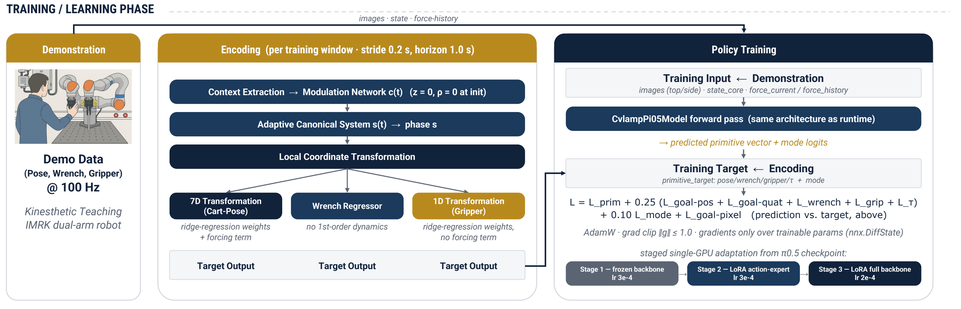
[1] Physical Intelligence et al. 2025 [2] Beyer et al. 2024 [3] Yu et al. 2025 [4] Zhao et al. 2026 [5] Li et al. 2026 [6] Ijspeert et al. 2013 [7] Ude et al. 2014 [8] Fabisch 2024 [9] Lee et al. 2019 [10] Li et al. 2026 [11] Dai et al. 2026 [12] Hu et al. 2021

Key Idea / Method Overview

CVLAMP predicts compact, force-aware Cartesian motion primitives, not dense action chunks, enabling compliant, interpretable manipulation that transfers directly from MuJoCo simulation to the real KUKA iiwa dual-arm platform (right-arm control in this work).

The figure below walks through the full pipeline: perception and language are fused by a PaliGemma-style backbone, decoded into a compact motion primitive, and executed under continuous force feedback.

- Observation → Policy: images, language, and force history are tokenized and fused via cross-attention into one primitive prediction.
- Policy → Primitive: a linear head decodes the fusion into a 140-dimensional Cartesian DMP + RBF primitive and a discrete contact mode.
- Primitive → Executor: rolled out at 100 Hz under a phase governor that can slow, hold, or abort on excessive pose/wrench error.
- Training is offline (composite loss, single-GPU frozen-backbone adaptation from the π0.5 checkpoint; LoRA stages planned); the same executor code runs on MuJoCo and is designed to transfer to the real KUKA iiwa dual-arm robot (IMRK platform), pending real-hardware validation.



Primitive representation

$$p_t = \{\theta pose, gpose, \theta wrench, \theta gripper, \tau, ugoal, m\}$$

140-dimensional primitive (pose / wrench / gripper basis 10 / 10 / 10 + 2-dim goal pixel) · 5 mode classes · phase governor evaluated at 100 Hz

Phase governor thresholds

SLOW	HOLD	ABORT
≥ 0.03 m pose error ≥ 8 N wrench error	≥ 0.08 m pose error ≥ 20 N wrench error	≥ 0.15 m pose error ≥ 40 N wrench error

Wrench commanding is disabled in the reported runs; the governor operates on pose error alone until the wrench head is validated.

Experiments and Results

Setup

- Task: pick-and-lift with the IMRK dual-arm robot in MuJoCo, under visual and force uncertainty.
- Demonstrations: canonical HDF5 logs of synchronized Cartesian pose, wrench, gripper, and joint states at 100 Hz, with JPEG camera frames sampled at the same rate (sim currently uses a single top-mounted camera; two physical cameras are planned for real-robot experiments).
- Training shards: extract_samples.py — 0.2 s stride, 1.0 s horizon, discarding camera frames older than 0.25 s; pose/wrench/gripper targets use the runtime primitive representation.
- Staged single-GPU (24 GB) adaptation from π0.5: (1) frozen backbone, lr 3e-4; (2) LoRA action-expert, lr 3e-4; (3) LoRA full backbone, lr 2e-4 for multi-task scaling.
- Only 0.21% of parameters are trainable in Stage 1 (6.9M of 3.36B, frozen backbone).

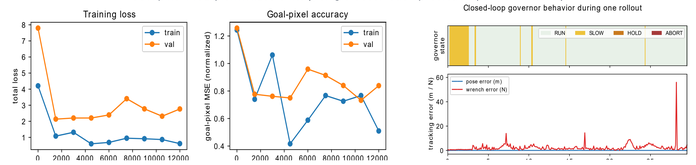
Metrics

- Task success rate.
- Contact robustness (peak / steady-state force error).
- Trajectory smoothness.
- Closed-loop stability (governor state transitions per rollout).
- Planned ablation: shared-phase wrench/gripper coupling.

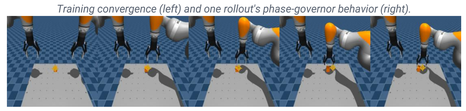
Results

TRAIN	VAL	HELD-OUT
0.00207	0.00246	0.00111

Goal-pixel MSE (normalized units²), single-observation prediction (raw, offline eval).



Training convergence (left) and one rollout's phase-governor behavior (right).



Closed-loop rollout: the model's predicted goal pixel (gold marker) tracks the cube as the arm approaches and grasps it.

Key Takeaways

- CVLAMP bridges VLA-scale semantic grounding with an explicit, auditable phase-governed motion primitive — not a black-box action chunk.
- A 140-dimensional primitive (pose DMP + wrench/gripper RBF profiles + duration + goal pixel + mode) keeps closed-loop safety interpretable and auditable.
- The same executor code runs in MuJoCo and is designed to transfer to the real KUKA iiwa hardware via config-only changes — real-hardware validation is ongoing.
- Single-GPU, frozen-backbone adaptation makes fine-tuning a released π0.5 checkpoint practical without full retraining; staged LoRA scaling is planned next.
- The explicit phase governor turns a hard safety problem into an auditable, rule-based one — every RUN/SLOW/HOLD/ABORT decision is traceable to a measured pose error (wrench commanding is disabled in the reported configuration).

dfki ai Deutsches Forschungszentrum für Künstliche Intelligenz
German Research Center for Artificial Intelligence

[1]Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI GmbH), 28359 Bremen, Germany
[2]Robotics Research Group, University of Bremen, 28359 Bremen, Germany

Universität Bremen