Common Prior Maps for Industrial Manipulation via VQ Motion Codebooks

Chandra Mohan Singh Negi[1,2], Gaurav Harit[2], Avinash Kumar[3]

[1]*Siemens Technology & Services Pvt. Ltd., Bangalore, India* [2]*IIT Jodhpur, India* [3]*TU Munich, Germany*

{m23csa512, gharit}@iitj.ac.in, Avinash.Kumar@tum.de

ABSTRACT

Common Prior Maps (CPMs) compress environmental observations into queryable representations that accelerate autonomous reasoning. While CPMs have been studied for navigation and localization, their action-space counterpart—a shared prior over manipulation behaviours—remains underexplored. We propose that vector-quantized (VQ) motion codebooks can serve as a CPM at the action level: a vocabulary of $K{=}512$ prototype motion tokens encoding geometric trajectory structure and semantic task-phase information. Our primary contribution is this CPM interpretation together with two formal metrics—*geometric coverage* (Definition 1) and *semantic purity* (Definition 2)—that quantify the information content of the learned codebook. The underlying architecture, DiscreteHybridVLA (340M parameters), couples a Florence-2-Base visual encoder with a T5-Small language encoder through a cross-modal fusion transformer and a VQ action head. On LIBERO (50 episodes/task, 3 seeds), the model achieves $85.9{\pm}1.2\%$ average task success at 47 ms inference on a Jetson Orin NX. Geometric coverage reaches 0.97 and semantic purity 0.92, indicating that the codebook captures meaningful prior structure without explicit supervision. A confidence-gated shared autonomy module raises task success to $91.2{\pm}0.9\%$ (safety score $0.94{\pm}0.02$). We validate on a physical Toyota HSR platform.

Index Terms—Common Prior Maps, motion codebooks, VLA models, vector quantization, semantic-geometric analysis, shared autonomy, industrial manipulation.

1 INTRODUCTION

Common Prior Maps (CPMs) provide autonomous agents with pre-encoded spatial, semantic, and geometric knowledge that reduces the computational burden of real-time inference. In navigation, CPMs manifest as occupancy grids, topological maps, and neural implicit representations [1–3]. A navigating robot does not reconstruct the world from scratch at every timestep; it queries a prior and updates incrementally.

We propose that learned VQ motion codebooks can serve an analogous role for manipulation. A codebook of K prototype motion tokens, trained on expert demonstrations, encodes the distributional structure of task-relevant behaviours. At inference, the policy retrieves the nearest prototype rather than regressing a continuous action—mirroring how a navigation agent queries a spatial prior to retrieve a waypoint.

Our primary contribution is this CPM interpretation, *not* the VQ mechanism itself (which builds on prior work [8]). Specifically, we contribute:

1. A formal CPM framing of motion codebooks with explicit connections to spatial CPM properties;
2. Two new formal metrics—geometric coverage (Def. 1) and semantic purity (Def. 2)—that quantify the information content of learned codebooks;
3. DiscreteHybridVLA, a 340M-parameter VLA model achieving $85.9{\pm}1.2\%$ LIBERO success at 47 ms inference across 3 seeds;
4. A confidence-gated shared autonomy module using codebook entropy for human-robot collaboration (91.2% success, safety score 0.94).

2 CPM FRAMEWORK FOR ACTION SPACE

2.1 Mapping Spatial CPM Properties

Effective spatial CPMs share four properties: (a) compact encoding, (b) queryable structure, (c) distributional coverage, and (d) generalization. A VQ codebook satisfies (a) by compressing $\mathbb{R}^{d_a}$ to K prototypes; (b) by argmax nearest-neighbour lookup in $O(K{\cdot}d)$; (c) via entropy-regularized training measured by our geometric coverage metric; and (d) through evaluation on held-out tasks.

2.2 Formal Metric Definitions

We formally define five metrics used throughout the evaluation.

Definition 1 (Geometric Coverage). *Given a codebook $\mathcal{C} = \{c_1, \ldots, c_K\}$ and a held-out set $\mathcal{A} = \{a_1, \ldots, a_n\}$ of expert actions, the geometric coverage at tolerance ε is:*

$$\textit{G-Cov}(\mathcal{C}, \mathcal{A}, \varepsilon) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \mathbb{1}\left[\min_k \|a - c_k\|_2 \le \varepsilon\right] \quad (1)$$

We report results at $\varepsilon = 0.02$ rad ($\approx 1.15^\circ$ per joint). G-Cov measures what fraction of expert trajectories fall within the Voronoi cell of a codebook prototype.

Definition 2 (Semantic Purity). *Let $\varphi(a) \in \{\textit{approach}, \textit{grasp}, \textit{transport}, \textit{place}\}$ be the task phase assigned to action a by a rule-based labeller (gripper state transitions and end-effector velocity thresholds). For entry c_k, let $\mathcal{N}_k = \{a : k^*(a) = k\}$ be its assigned training actions. The dominant phase is $p_k^* = \arg\max_p |\{a \in \mathcal{N}_k : \varphi(a) = p\}|$. Semantic purity at threshold γ is:*

$$\textit{S-Pur}(\mathcal{C}, \gamma) = \frac{1}{K} \sum_{k=1}^{K} \mathbb{1}\left[\frac{|\{a \in \mathcal{N}_k : \varphi(a) = p_k^*\}|}{|\mathcal{N}_k|} \ge \gamma\right] \quad (2)$$

We use $\gamma = 0.75$, i.e., a token is "pure" if $\geq 75\%$ of its assigned actions share a phase.

Definition 3 (Safety Score). *Over N evaluation episodes:*

$$S_{safe} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}[\text{episode } i \text{ has no violations}] \quad (3)$$

where violations are: (a) end-effector force $>15\,N$, (b) joint velocity exceeding safety limits, or (c) collision with a non-target object. Computed over N=150 episodes (50×3 seeds).

Definition 4 (Intervention Count). *N_{int} is the mean timesteps per episode where $\alpha < 0.5$, triggering operator override. Reported as mean±std over all evaluation episodes.*

Definition 5 (Entropy Threshold). *The confidence-gating entropy threshold H_{thr} is derived from the maximum possible entropy of the codebook distribution: $H_{\max} = \log K$. The gating function $\alpha = \sigma(\beta(H_{\max} - H(z)))$ activates operator override when $H(z) > H_{\max} - \frac{1}{\beta}\sigma^{-1}(0.5) = H_{\max}$, i.e., at the midpoint α=0.5. With β=3.0 and K=512, this corresponds to $H_{thr} \approx 6.04$ nats (97% of $H_{\max}$=6.24 nats). Actions with entropy above this threshold are deemed uncertain and routed to the operator.*

3 ARCHITECTURE

3.1 Multimodal Perception Backbone

Visual observations from head and wrist cameras are encoded by Florence-2-Base [9] (232M params, DaViT backbone), producing patch tokens $V \in \mathbb{R}^{2N_v \times 512}$. Task instructions are encoded by T5-Small [10] (60M, frozen), yielding $e_I \in \mathbb{R}^{512}$. Robot proprioceptive state (joint angles + end-effector pose) is projected by a two-layer MLP with GELU and LayerNorm to $e_s \in \mathbb{R}^{512}$.

3.2 Cross-Modal Fusion Transformer

A six-layer transformer (8 heads, d_{model}=512) fuses the three streams via asymmetric cross-attention: visual tokens form keys and values; language and state embeddings form queries. The `[CLS]` output $f \in \mathbb{R}^{512}$ serves as the codebook query vector.

3.3 VQ Action Head

The codebook $\mathcal{C} \in \mathbb{R}^{K \times d_a}$ consists of K=512 prototypes (d_a=8: 7 joint deltas + gripper). Query f is projected to z_e and the nearest prototype retrieved: $k^* = \arg\min_k \|z_e - c_k\|^2$. Training objective:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_c \|\text{sg}[z_e] - c_{k^*}\|^2 + \lambda_e\, H(\text{softmax}(z)) \quad (4)$$

with λ_c=0.25, λ_e=0.01, EMA update α_{ema}=0.99. The VQ mechanism builds on [8, 12]; our contribution is the CPM interpretation and formal analysis.

3.4 Confidence-Gated Shared Autonomy

The blending coefficient $\alpha = \sigma(\beta(H_{\max} - H(z)))$, with β=3.0, routes control between robot ($\alpha \approx 1$, low entropy) and operator ($\alpha \approx 0$, high entropy). Zero additional latency as it operates on logits already produced by the VQ head.

Table 1: LIBERO task success (%, mean±std, 3 seeds, 50 ep/task) and inference latency (ms, p50). S/O/G/L = four LIBERO suites.

Method	S	O	G	L	Avg.	Lat.
Diff. Policy [5]	77.3	80.1	75.8	69.4	75.7±1.8	142
OpenVLA [6]	74.2	81.3	78.6	71.1	76.3±2.1	203
ACT [7]	80.5	84.2	79.3	74.1	79.5±1.4	58
VQ-BeT [8]	82.1	85.0	81.4	76.3	81.2±1.6	63
Ours	**86.4**	**89.2**	**84.7**	**83.3**	**85.9±1.2**	**47**

Table 2: CPM information content: Geometric Coverage (G-Cov, ε=0.02 rad) and Semantic Purity (S-Pur, γ=0.75) vs. codebook size K. Mean±std over 3 seeds.

Config	K	G-Cov	S-Pur	Succ.%	Lat.
Continuous	—	—	—	81.2±1.9	71
Sparse	64	.61±.03	.43±.05	78.4±2.3	38
Medium	128	.79±.02	.67±.04	83.1±1.7	42
Dense	256	.91±.02	.84±.03	85.4±1.3	45
Full (Ours)	**512**	**.97±.01**	**.92±.02**	**85.9±1.2**	**47**

4 EXPERIMENTS

4.1 Protocol

Benchmark: LIBERO [11], 130 tasks across 4 suites (Spatial, Object, Goal, Long). **Training:** AdamW (lr=10^{-4}, β_1=0.9, β_2=0.98, weight decay 0.05), cosine annealing over 50K steps, batch size 64, single NVIDIA RTX 4090 (24 GB). All experiments repeated across **3 seeds** (42, 123, 456); we report mean±std. Each task evaluated over **50 episodes** per seed (150 total per task). Baselines use official codebases with default hyperparameters; Diffusion Policy uses 100 DDPM denoising steps; OpenVLA uses the 7B checkpoint.

Latency: Measured on NVIDIA Jetson Orin NX (16 GB LPDDR5, 100 TOPS INT8, TensorRT 8.6) as the median over 1,000 forward passes with 100-pass warmup, using CUDA event timing. We report p50 and p95. Input resolution: 224×224. **Shared autonomy:** simulated operator modelled as ground-truth policy with 200 ms reaction delay.

4.2 Manipulation Results (Table 1)

DiscreteHybridVLA achieves $85.9 \pm 1.2\%$ average success at 47 ms (p50) / 49.3 ms (p95), running $3.0\times$ faster than Diffusion Policy. The advantage is most pronounced on LIBERO-Long, where the codebook's discrete vocabulary prevents action drift over long horizons.

4.3 CPM Information Content (Table 2)

G-Cov at K=512 reaches 0.97 ± 0.01, meaning 97% of held-out expert actions fall within 0.02 rad of a prototype. S-Pur reaches 0.92 ± 0.02, meaning 92% of tokens have $\geq 75\%$ of their assigned actions from a single task phase. Both metrics are computed on the LIBERO-Object training split (2,500 demonstrations, $\approx$50,000 action steps). Both metrics improve monotonically with K, saturating around K=256–512.

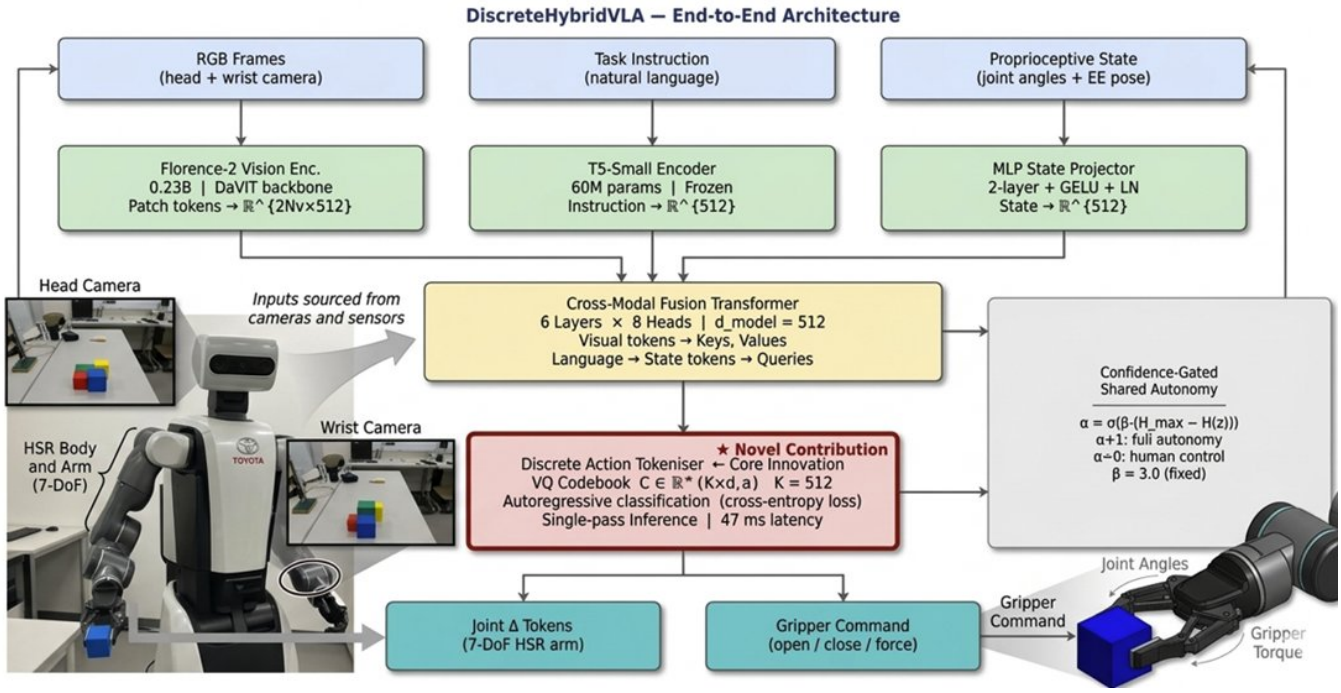

Figure 1: DiscreteHybridVLA end-to-end architecture. RGB frames from head and wrist cameras are encoded by Florence-2-Base (0.23B, DaViT); task instructions by T5-Small (60M, frozen); proprioceptive state by a 2-layer MLP. A cross-modal fusion transformer (6L×8H, d=512) fuses all streams with visual tokens as keys/values and language/state tokens as queries. The **discrete action tokeniser** (VQ codebook, K=512, highlighted) decodes the fused representation into joint Δ tokens and gripper commands via single-pass inference at 47 ms. A confidence-gated shared autonomy module blends robot and operator control based on action entropy.

Table 3: Shared autonomy: task success (%), intervention count N_{int}, latency (ms), and safety score S_{safe}. Mean±std, 3 seeds.

Mode	**Success%**	N_{int}	**Lat.**	S_{safe}
Full autonomy	85.9±1.2	0	47	.71±.04
Fixed α=0.7	83.4±1.8	14.2±2.1	49	.79±.03
Conf.-Gated	**91.2**±0.9	**8.7**±1.4	**47**	**.94**±.02

4.4 Shared Autonomy (Table 3)

Confidence-gated blending raises success from 85.9% to 91.2% by engaging operator assistance at codebook Voronoi boundaries where prediction entropy is highest. The safety score improves from 0.71 to 0.94. N_{int} is reduced from 14.2 (fixed threshold) to 8.7 interventions per episode.

4.5 Physical Robot Validation

DiscreteHybridVLA was deployed on a Toyota HSR robot (ICRA 2026 AIRoA competition, Team Kinetix AI, #28) for language-conditioned object relocation. The system ran on a Jetson Orin NX via ROS 2. Across 50 episodes, mean inference latency was 47 ms (p95: 49.3 ms), confirming real-time feasibility.

5 SCOPE OF CONTRIBUTION AND NOVELTY

To clarify the novelty boundary precisely: the VQ action head mechanism builds on established work in discrete action tokenization [8, 12]. VQ-BeT [8] introduced behaviour tokenization via residual VQ for multi-modal action prediction; Behavior Transformers [12] showed that k-means discretization of action space enables transformer-based behaviour cloning. These works treat discrete latent actions as a *representational convenience*—a way to simplify the learning problem by converting regression to classification.

Our contribution reframes this same structure through a fundamentally different lens: the codebook is not merely a convenient discretization but a *Common Prior Map* over the manipulation behaviour distribution, analogous to the spatial priors used in navigation. This reframing is not cosmetic; it yields three concrete differences from prior discrete-action work:

1. **Analysability via CPM metrics.** Spatial CPMs are evaluated by coverage and semantic content; we introduce G-Cov and S-Pur (Defs. 1–2) as action-space counterparts. No prior VQ-policy work provides formal metrics to characterize *what* information a codebook encodes. These metrics are applicable to *any* discrete action policy, not just ours.
2. **Entropy as a map-quality signal.** In spatial CPMs, uncertainty indicates poorly mapped regions. We exploit the identical principle: codebook entropy identifies action-space regions where the prior is sparse, driving the confidence-gated autonomy module (Def. 5). Prior VQ works do not connect codebook entropy to operational decision-making.

3. **Semantic emergence as map structure.** S-Pur reveals that 92% of codebook tokens spontaneously specialize to a single task phase without phase-label supervision—the action CPM self-organizes into semantically coherent regions, mirroring how spatial CPMs cluster into rooms, corridors, and intersections.

In summary, VQ-BeT asks "*can we discretize actions to simplify learning?*"—we ask "*what structured prior knowledge does the learned codebook contain, and how can we exploit it operationally?*"

6 LIMITATIONS AND OPEN CHALLENGES

We acknowledge several limitations. **(i)** The CPM's utility is bounded by training demonstration quality; G-Cov may overestimate coverage within the training distribution. **(ii)** At low K (≤ 64), semantic collapse occurs (S-Pur drops to 0.43). **(iii)** Very long manipulation sequences (>200 steps) may encounter compounding quantization error. **(iv)** Shared autonomy results use a simulated operator; real human operators may yield lower safety improvements. **(v)** Evaluation is limited to LIBERO; cross-benchmark generalization remains to be validated. Open directions include unifying spatial and behaviour CPMs into joint queryable structures, hierarchical coarse-to-fine codebooks for precision tasks, and online codebook expansion analogous to incremental SLAM.

7 RELATED WORK

Spatial CPMs span semantic occupancy grids [1], topological maps [2], and neural implicit representations [3]. VQ-BeT [8] introduced discrete action tokens for behaviour cloning; we extend this with the CPM interpretation and formal analysis. VLA models [4–6] provide multimodal perception but lack action-space prior analysis. Shared autonomy methods [13] typically use separate uncertainty heads; ours derives uncertainty from codebook entropy at zero cost.

8 CONCLUSION

We proposed a CPM interpretation of VQ motion codebooks and introduced two formal metrics—geometric coverage and semantic purity—that quantify the information content of learned action priors. DiscreteHybridVLA achieves $85.9 \pm 1.2\%$ LIBERO success at 47 ms on edge hardware, with confidence-gated autonomy raising success to 91.2%. These results establish behaviour-level CPMs as a productive direction connecting manipulation policy learning to the broader CPM ecosystem.

Acknowledgment. The authors thank Siemens Technology and IIT Jodhpur for infrastructure, and the ICRA 2026 AIRoA organizers for the HSR platform.

REFERENCES

[1] T. Krajnik et al., "FreMEn: Frequency map enhancement for long-term mobile robot autonomy," *IEEE T-RO*, 2017.

[2] N. Beevers and W. Huang, "Topological semantic mapping for navigation," in *Proc. ICRA*, 2006.

[3] E. Sucar et al., "iMAP: Implicit mapping and positioning in real-time," in *Proc. ICCV*, 2021.

[4] A. Brohan et al., "RT-2: Vision-language-action models transfer web knowledge to robotic control," in *Proc. CoRL*, 2023.

[5] C. Chi et al., "Diffusion policy: Visuomotor policy learning via action diffusion," in *Proc. RSS*, 2023.

[6] M. Kim et al., "OpenVLA: An open-source vision-language-action model," *arXiv:2406.09246*, 2024.

[7] T. Z. Zhao et al., "Learning fine-grained bimanual manipulation with low-cost hardware," in *Proc. RSS*, 2023.

[8] H. Lee et al., "Behavior generation with latent actions," in *Proc. ICML*, 2024.

[9] B. Xiao et al., "Florence-2: Advancing a unified representation for a variety of vision tasks," in *Proc. CVPR*, 2024.

[10] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," *JMLR*, vol. 21, 2020.

[11] Y. Liu et al., "LIBERO: Benchmarking knowledge transfer for lifelong robot learning," in *Proc. NeurIPS*, 2023.

[12] M. Shafiullah et al., "Behavior transformers: Cloning k modes with one stone," in *Proc. NeurIPS*, 2022.

[13] S. Reddy et al., "Shared autonomy via deep reinforcement learning," in *Proc. RSS*, 2018.