From Aerial Twins to VLA Tuples: A Zero-NRE Data Factory with Caption Drift Evaluation

Anuj Gupta
SimForge
Palo Alto, CA, USA
anuj@simforge.ai

Sourang Sri hari
SimForge
Palo Alto, CA, USA
sourang@simforge.ai

Dibyendusekhar Goswami
SimForge
Palo Alto, CA, USA
dib@simforge.ai

Michael Vu
SimForge
Palo Alto, CA, USA
michael@simforge.ai

Abhishek Shinde
SimForge
Palo Alto, CA, USA
abhishek@simforge.ai

Mayank Gupta
SimForge
Palo Alto, CA, USA
mayank@simforge.ai

***Abstract*—Urban robotic systems, including autonomous vehicles (AVs), UAVs, humanoid robots, and sidewalk delivery robots, share a common data bottleneck: generating perceptually realistic, geographically diverse training environments without expensive ground-vehicle survey fleets or prohibitive non-recurring engineering (NRE) costs. We present a *VLM Data Factory*: a four-stage, zero-NRE pipeline combining aerial digital twins, cloud-based physics simulation, a video-to-video world model for perceptual augmentation, and a vision-language model for automated semantic annotation, all on a pay-as-you-go basis. We further introduce *caption drift*, a geometry-invariant evaluation signal derived from changes in automatically generated scene captions under controlled perceptual variation: because geometry, agent trajectories, and physics are held fixed while only perception changes, any caption shift is attributable solely to perceptual domain gap. We demonstrate caption drift across 20+ structured conditions spanning weather, lighting, and world-model guidance parameters, and show it qualitatively tracks perceptual augmentation intensity. The pipeline generates complete VLA training tuples at $\$0.25\,\text{USD}\,\text{s}^{-1}$ per augmentation condition ($\$5.00\,\text{s}^{-1}$ across all 20 conditions), replacing NRE investments with usage-proportional expenditure.**

***Index Terms*—sim-to-real transfer, Vision-Language-Action models, urban robotics, data factory, domain randomization, evaluation metrics, digital twins, synthetic data generation**

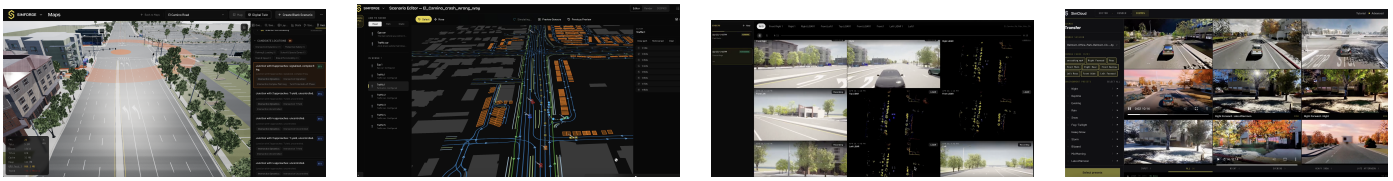
I. Introduction

Vision-Language-Action (VLA) models have emerged as a dominant paradigm for building generalizable robot policies. Systems such as RT-2 [1], OpenVLA [2], π_0 [3], PaLM-E [4], and Gato [5] ground robot control in web-scale vision and language representations, achieving impressive cross-task and cross-embodiment generalization. Large-scale data initiatives such as Open X-Embodiment [6] and DROID [7] have further demonstrated that policy quality scales predictably with data diversity and volume. Yet a persistent gap remains: models trained in simulation, or on narrowly collected teleoperation data, fail to transfer reliably to real sensors due to perceptual distribution mismatch in lighting, atmosphere, and surface reflectance. This challenge spans all urban robotics platforms, including AVs, UAVs, humanoid robots, and sidewalk delivery robots, all of which require training environments with diverse geographies and conditions. Existing approaches rely on LiDAR survey vehicles and GPU-intensive reconstruction pipelines that cost $\$25{,}000\,\text{km}^{-2}$ and impose large NRE commitments before a single training sample is produced [11]. Moreover, evaluating whether a trained policy survives real-world perceptual conditions still requires physical robot experiments that cannot be scaled for ablation studies or hyperparameter search.

Prior work addresses sim-to-real via domain randomization [10] or photo-realistic rendering [12], but proposes no scalable *evaluation metric* decoupled from real robot deployments. Language-conditioned robot policies such as SayCan [14] and CLIPort [15] further highlight the importance of tight vision-language grounding, yet their evaluation remains tied to physical setups. VLA data pipelines such as RT-2 rely on teleoperation; simulation-based alternatives [8] suffer from the perceptual reality gap. Prior appearance-transfer approaches using game-engine imagery [18] showed promise but lacked the semantic annotation layer needed for VLA fine-tuning. Video world models such as NVIDIA Cosmos [9] now enable controllable appearance transfer while preserving geometry, a property we exploit to isolate perceptual variation as an experimental variable. We address both the data bottleneck and the evaluation gap with a *VLM Data Factory*: a four-stage pipeline operating at zero NRE on a pay-as-you-go model, paired with *caption drift* as a robot-free evaluation signal for sim-to-real perceptual fidelity.

II. The Four-Stage VLA Data Pipeline

Fig. 1 illustrates the pipeline; each stage is independently scalable and cost-transparent. Together, the four stages form a closed loop from raw geography to annotated $(v_t, \ell, \mathbf{a}_t)$ training tuples without any ground-vehicle operation or on-premise GPU infrastructure.



(a) Candidate locations (b) Scenario creation (c) Sensor output (d) Cosmos augmentation

Fig. 1. VLM Data Factory pipeline on El Camino Real, Palo Alto. **(a)** SimForge Maps Digital Twin: 36 candidate scenario locations auto-detected and ranked by intersection complexity (95% for the highlighted 8-leg signalized junction). **(b)** Scenario Editor: ego vehicle and five traffic actors scripted with time-stamped waypoints for the `El_Camino_crash_wrong_way` scenario. **(c)** Synchronized multi-sensor output: six RGB cameras and three LiDAR streams, forming visual input v_t of VLA training tuples. **(d)** Cosmos Transfer 2.5B augmentation: the same CARLA scene across nine environment presets (Night, Evening, Heavy Snow, Late Afternoon, and others) with geometry and agent trajectories held fixed.

Stage 1: Aerial Digital Twin Reconstruction. Traditional reconstruction pipelines rely on NeRF [12] or photogrammetry driven by ground-level LiDAR vehicles, incurring large NRE costs before any simulation asset is usable. Our aerial-first approach processes licensed aerial photogrammetry into *SimReady* assets (geometry, HD semantic maps, and lane networks) at $\sim$10 cm relative accuracy and \$2,500 USD km^{-2}, a 10$\times$ cost reduction over ground-vehicle reconstruction [11], with no additional GPU-intensive processing step. The resulting assets are immediately compatible with CARLA and cover full urban road networks including intersections, parking zones, and pedestrian paths. All experiments in this paper use a 1.2 km corridor on El Camino Real, Palo Alto, CA, with 36 candidate scenario locations auto-detected by the SimForge Maps platform.

Stage 2: Cloud-Based Synthetic Data Generation (CARLA). Digital twins are imported into CARLA 0.9.16 [8], an open-source, researcher-first simulator deployed as a cloud service with no on-premise GPU required. Multi-camera sensor rigs and traffic agent spawners produce *action-labeled* recordings: waypoints, steering, throttle, and brake signals logged at every step, forming the action component $\mathbf{a}_t$ of VLA training tuples. Traffic flow is scripted using time-stamped waypoints, enabling reproducible edge-case scenarios such as wrong-way conflicts and unprotected left turns, analogous in spirit to multi-agent simulation frameworks used in large-scale AV research [19]. CARLA's open API supports adaptation to AV, UAV, and ground-robot sensor configurations, and the cloud deployment model means experiment throughput scales horizontally without hardware procurement.

Stage 3: Perceptual Augmentation (Cosmos Transfer 2.5B). CARLA renders are passed to NVIDIA Cosmos Transfer 2.5B [9], conditioned on edge maps and text prompts. This approach extends earlier work on using game-engine and synthetic imagery for perception training [18] by adding a learned world model that introduces physically plausible sensor characteristics rather than simple texture overlays. Control weight $\omega_c \in [0.5, 1.0]$ governs geometric fidelity; transfer guidance $\omega_g \in [3, 7]$ controls prompt adherence. We evaluate 20+ combinations across five weather conditions and three time-of-day settings; lighting, atmospheric diffusion, surface reflectance, and sensor noise all change while geometry and motion are preserved, isolating perceptual variation as the sole experimental variable.

NVIDIA Cosmos Transfer 2.5B operates as a video-to-video diffusion model: given a structured control signal derived from CARLA edge maps and a natural-language environment prompt (e.g., "heavy snow, night, urban lighting"), it synthesizes a perceptually realistic video clip in which all physics-level properties of the original simulation remain intact. The edge-map conditioning acts as a geometric anchor, preserving lane markings, vehicle silhouettes, pedestrian positions, and building outlines, while the diffusion process fills in surface materials, atmospheric scattering, precipitation, and lens artifacts consistent with the target environment description. This mechanism transforms a single simulated scenario into a family of perceptually distinct variants without re-running any physics simulation or re-scripting any agent behavior. For scenario creation, one carefully scripted edge case (e.g., an unprotected left turn or a wrong-way conflict) can be instantiated across rain, fog, night, and heavy snow at the cost of a single Cosmos Transfer inference pass per condition, producing a structured grid of training samples that would otherwise require separate real-world data collection in each weather regime.

Stage 4: Semantic Annotation (Cosmos Reason-2). Each clip (baseline and augmented) is annotated with NVIDIA Cosmos Reason-2, a video-language model that extracts structured fields (*actor*, *action*, *weather*, *time-of-day*, *risk indicators*) in both freeform and structured output modes. This automated annotation replaces human labelers for the language component ℓ of training tuples, yielding complete $(v_t, \ell, \mathbf{a}_t)$ triplets for VLA fine-tuning. The structured output mode enables downstream scenario search by keyword (e.g., "low visibility right-turn conflict"), analogous to dense captioning pipelines used in vision-language pre-training [17].

III. Caption Drift as an Evaluation Metric

Existing sim-to-real metrics require either a physical robot or pixel-level statistics (FID, LPIPS) that correlate poorly with downstream policy performance [13]. Semantic similarity scores such as CLIP [16] offer a step forward but are not designed to measure the perceptual domain gap that a VLA policy's language backbone would experience at deployment. Because VLA language backbones are trained on real-world images and captions, they are inherently sensitive to perceptual

TABLE I
CAPTION SHIFT VS. CONTROL WEIGHT ω_c ("WET ROAD, DAY")

ω_c	Representative caption shift
1.0	No weather language; geometry and trajectory focus
0.7	"wet asphalt", "reduced traction", "light rain streaks"
0.5	"heavy precipitation", "severely reduced visibility"

TABLE II
CAPTION DRIFT MAGNITUDE BY WEATHER CONDITION

Condition	Drift	Example emergent keyword
Clear, noon (baseline)	n/a	n/a
Morning, early fall	Low	"golden light", "long shadows"
Evening, summer	Low to medium	"dusk", "reduced contrast"
Fog	Medium	"reduced visibility", "diffuse light"
Rain	Medium to high	"wet surface", "reduced traction"
Night, urban	High	"headlight glare", "low visibility"
Snow, heavy	High	"snow accumulation", "blizzard"

cues: the descriptions they produce should shift when the perceptual distribution of the input departs from training conditions. We formalize this intuition as *caption drift*. Let c_0 be the caption of a baseline CARLA render of scene s, and c_i the caption of its Cosmos-augmented variant under condition i. We define:

$$D_i = 1 - \frac{\mathbf{e}(c_0) \cdot \mathbf{e}(c_i)}{\|\mathbf{e}(c_0)\|\|\mathbf{e}(c_i)\|} \tag{1}$$

where $\mathbf{e}(\cdot)$ is a sentence embedding produced by a pre-trained text encoder (e.g., a CLIP text tower [16] or a dedicated sentence transformer). $D_i \in [0, 1]$; because geometry, physics, and agent state are fixed across c_0 and c_i, any $D_i > 0$ is attributable solely to perceptual variation introduced by Cosmos Transfer. The metric is cheap to compute (no robot required, no pixel rendering beyond what Stage 3 already produces) and is architecture-agnostic: it can be applied to any VLA pipeline that uses a language model for action conditioning. We hypothesize that D_i proxies the perceptual domain gap a VLA policy faces when deployed under condition i; high-D_i conditions such as fog and night should represent harder transfer scenarios than low-D_i conditions such as morning or evening. Empirical validation against real-hardware rollout performance is left for future work, as is exploration of fine-tuning on the CALVIN benchmark [20] to assess generalization from high-D_i training data.

IV. EXPERIMENTS AND RESULTS

All experiments use the El Camino Real corridor in Palo Alto, CA (northbound and southbound segments) and a scripted wrong-way collision scenario. Clips are 15 to 20 seconds at 30 fps. We report qualitative caption shift as the primary observable, with approximate cosine-distance D values drawn from one representative clip per condition to illustrate the trend.

Control weight. Table I shows caption shifts as ω_c decreases (guidance fixed at ω_g=3, prompt: "wet road, day"). At ω_c=1.0, captions are nearly identical to baseline ($D\approx0.04$); at 0.7, weather and traction language emerges ($D\approx0.21$); at 0.5, extreme precipitation language dominates ($D\approx0.38$) at some cost to temporal smoothness. This monotonic relationship confirms that ω_c directly controls the magnitude of perceptual injection and, consequently, the magnitude of caption drift.

Transfer guidance. Increasing ω_g from 3 to 7 (at ω_c=1.0) produces monotonically stronger stylistic descriptors ("intense glare", "high-contrast shadows") absent in baseline captions, confirming D_i captures domain shift even when geometric adherence is strict. This is particularly relevant for deployment in regions with strong directional sunlight or urban canyon lighting, where stylistic drift without geometric drift may be the primary domain gap.

Weather conditions. Table II summarizes drift across six conditions (ω_c=1.0, ω_g=3). Rain and Night produce the largest drift, consistent with their strong perceptual departure from the clear-noon baseline. Morning and Evening produce low to medium drift, suggesting that policies trained on clear-noon data would face only moderate perceptual shift in those conditions, while fog, rain, and night represent high-risk deployment scenarios requiring targeted augmentation.

Collision scenario. In the `El_Camino_crash_wrong_way` scenario (Fig. 2), baseline captions focus on trajectory conflict and right-of-way violation. Augmented variants introduce weather-specific risk language: wet-road versions reference reduced stopping distance; night versions cite limited visibility as a contributing factor; snow versions describe loss-of-traction risk. The structured output mode of Cosmos Reason-2 surfaces these as discrete, queryable fields, enabling scenario-searchable annotation without manual labeling. This mirrors the utility of dense video captioning systems [17] applied to safety-critical robotics data.

V. PIPELINE ECONOMICS

A persistent barrier to large-scale VLA data generation is cost unpredictability. Traditional pipelines bundle environment capture, reconstruction, and simulation into large upfront NRE commitments, making it difficult for academic labs or small robotics teams to access city-scale training data. Table III summarizes the marginal costs of our pipeline. Ground-vehicle teleoperation and LiDAR reconstruction require six-figure NRE before a single training sample is produced; our pipeline replaces this with *pay-as-you-go* pricing: environment coverage at \$2,500 km^{-2} and Cosmos Transfer augmentation at \$0.25 s^{-1} per condition (\$5.00 s^{-1} across all 20 conditions). The same twin reuses across unlimited perceptual variants at no extra environment cost, providing free domain random-

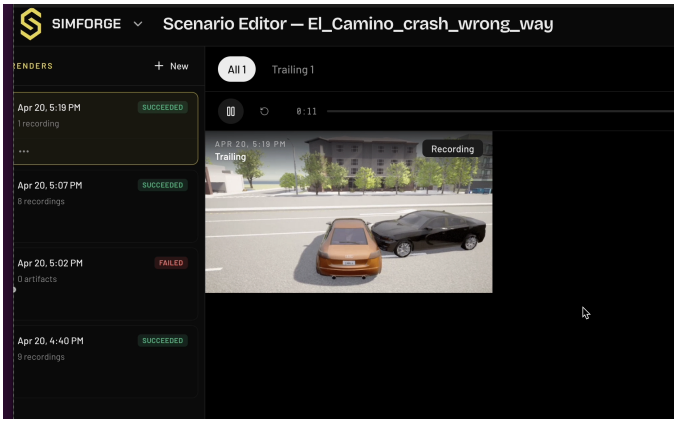


Fig. 2. The `El_Camino_crash_wrong_way` scenario (trailing camera). An orange sedan and black vehicle interact in the scripted wrong-way conflict; multiple render sessions (left panel) generate caption drift measurements across perceptual conditions.

TABLE III
PAY-AS-YOU-GO PIPELINE COST BREAKDOWN

Component	Unit	Cost (USD)
Aerial twin + HD maps	per km^2	\$2,500
Cloud CARLA simulation	per second of video	\$0.05
Cosmos Transfer (per condition)	per second of video	\$0.25
Cosmos Transfer (all 20 conditions)	per second of video	\$5.00
Cosmos Reason-2 annotation	per clip	$<$\$0.05
VLA tuple, 1 condition (15 s)	per clip	$\approx$\$3.80
VLA tuple set, all 20 conditions (15 s)	per clip set	$\approx$\$75.05

ization post-licensing. This access model removes barriers for research groups working on open-vocabulary manipulation [15], mobile manipulation [14], and multi-embodiment generalization [6], all of which benefit from large, diverse, and perceptually varied training corpora.

VI. Discussion and Conclusion

Caption drift serves as a *pre-deployment screening tool*: given a target deployment environment, one can generate Cosmos Transfer augmentations mimicking its perceptual conditions and compute D_i to flag distribution mismatch before any robot is dispatched; conditions where $D_i > \tau$ trigger targeted data collection or fine-tuning. This is analogous in spirit to the sim-to-real predictivity framework of Kadian et al. [13], but operates entirely in the language domain without requiring any hardware rollouts. Caption drift is complementary to pixel-level metrics (FID, LPIPS), which quantify image quality; D_i quantifies *semantic* departure from the training distribution using the same embedding space that VLA language backbones operate in [16]. It is also complementary to behavioral metrics (task success rate, intervention frequency), which require physical robots and cannot be run during dataset construction. Current limitations include sensitivity of absolute D_i values to the choice of sentence embedding model and the absence of end-to-end validation correlating D_i with real-hardware policy rollout performance; the latter is the primary planned extension. We also plan to fine-tune OpenVLA [2] and evaluate on the CALVIN manipulation benchmark [20] to assess whether high-D_i augmented training data improves real-world transfer relative to clean-simulation baselines.

We have presented a *VLM Data Factory* (aerial digital twin, cloud-based SDG, Cosmos Transfer augmentation, and Cosmos Reason-2 annotation) that generates complete $(v_t, \ell, \mathbf{a}_t)$ VLA training tuples at city scale, zero NRE, and pay-as-you-go marginal cost, applicable across AVs, UAVs, humanoids, and sidewalk robots. The central contribution is *caption drift* (Eq. (1)): a robot-free, geometry-invariant evaluation signal that quantifies perceptual sim-to-real gap through automated semantic annotation, with qualitative sensitivity demonstrated across 20+ weather and lighting conditions on a real urban corridor.

Acknowledgment

The authors thank Runpod.io for H100 compute credits used for Cosmos Transfer experiments.

References

[1] A. Brohan et al., "RT-2: Vision-language-action models transfer web knowledge to robotic control," in *Proc. CoRL*, 2023.
[2] M. J. Kim et al., "OpenVLA: An open-source vision-language-action model," arXiv:2406.09246, 2024.
[3] K. Black et al., "π_0: A vision-language-action flow model for general robot control," arXiv:2410.24164, 2024.
[4] D. Driess et al., "PaLM-E: An embodied multimodal language model," in *Proc. ICML*, 2023.
[5] S. Reed et al., "A generalist agent," *Trans. Mach. Learn. Res.*, 2022.
[6] A. Padalkar et al., "Open X-Embodiment: Robotic learning datasets and RT-X models," in *Proc. ICRA*, 2024.
[7] A. Khazatsky et al., "DROID: A large-scale in-the-wild robot manipulation dataset," arXiv:2403.12945, 2024.
[8] A. Dosovitskiy et al., "CARLA: An open urban driving simulator," in *Proc. CoRL*, 2017.
[9] NVIDIA, "Cosmos: World foundation model platform for physical AI," Tech. Rep., 2025. [Online]. Available: https://developer.nvidia.com/cosmos
[10] J. Tobin et al., "Domain randomization for transferring deep neural networks from simulation to the real world," in *Proc. IROS*, 2017.
[11] H. Caesar et al., "nuScenes: A multimodal dataset for autonomous driving," in *Proc. CVPR*, 2020.
[12] T. Müller et al., "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, 2022.
[13] A. Kadian et al., "Sim2Real predictivity: Does evaluation in simulation predict real-world performance?" *IEEE RA-L*, vol. 5, no. 4, 2020.
[14] M. Ahn et al., "Do as I can, not as I say: Grounding language in robotic affordances," in *Proc. CoRL*, 2022.
[15] M. Shridhar, L. Manuelli, and D. Fox, "CLIPort: What and where pathways for robotic manipulation," in *Proc. CoRL*, 2022.
[16] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. ICML*, 2021.
[17] J. Li et al., "BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *Proc. ICML*, 2023.
[18] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. ECCV*, 2016.
[19] R. Krajewski et al., "The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways," in *Proc. ITSC*, 2018.
[20] O. Mees et al., "CALVIN: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks," *IEEE RA-L*, vol. 7, no. 3, 2022.