Kosmic PACE: Physics-Aware Control Extraction from Egocentric Driving Video

Public Overview

Shaan Vats
Kosmic Worlds
shaan@kosmicworlds.com

October 2026

Abstract

End-to-end (E2E) driving policies and action-conditioned world models need driving video paired with the driver's control inputs. Public driving video is abundant, but video synchronized with CAN bus logs containing steering wheel angle, throttle percentage, and brake percentage remain rare. We introduce Kosmic PACE (*Physics-Aware Control Extraction*), a physics-based algorithm that recovers steering wheel angle, throttle percentage, and brake percentage from egocentric driving data, producing action labels in physical units. Evaluated against ground truth CAN bus logs on eight public datasets spanning three continents, including sedans, SUVs, a van, a coach bus, and a semi-truck, PACE recovers steering wheel angle with a mean absolute error (MAE) of 4.16° and a root mean square error (RMSE) of 7.50° at speeds above 5 m/s (18 km/h), and throttle percentage with an MAE of 6.56 percentage points across the seven datasets with throttle ground truth. PACE is the labeling layer behind Kosmic Worlds' training data for driving world models and policies.

1 Why Action Labels Matter

World models learn how a scene changes in response to a driver's inputs. E2E policies learn to produce those inputs from camera images and other sensors. Both need action labels that are synchronized with the video. Most driving footage, including public datasets, dashcam video, and fleet recordings, contain no CAN bus logs, so they cannot be used for training action-conditioned world models or E2E policies.

Existing ways of recovering actions from video each fall short at scale. Learned inverse dynamics models [2] need action-labeled data from the target domain and do not transfer reliably to new vehicles or cameras. Latent action models [3] need no labels, but the actions they learn are abstract and do not map to physical units(steering wheel angles or pedal percentages). Neither produces labels that a policy could execute on a real vehicle.

PACE takes a different approach: instead of learning a mapping from pixels to actions, it models how the vehicle physically moves and mathematically computes the control inputs that produced that motion. Because it relies on physics rather than training data from a particular vehicle, it transfers across vehicles, cameras, and frame rates without retraining, and every output is in physical units: degrees at the steering wheel and percent of pedal travel.

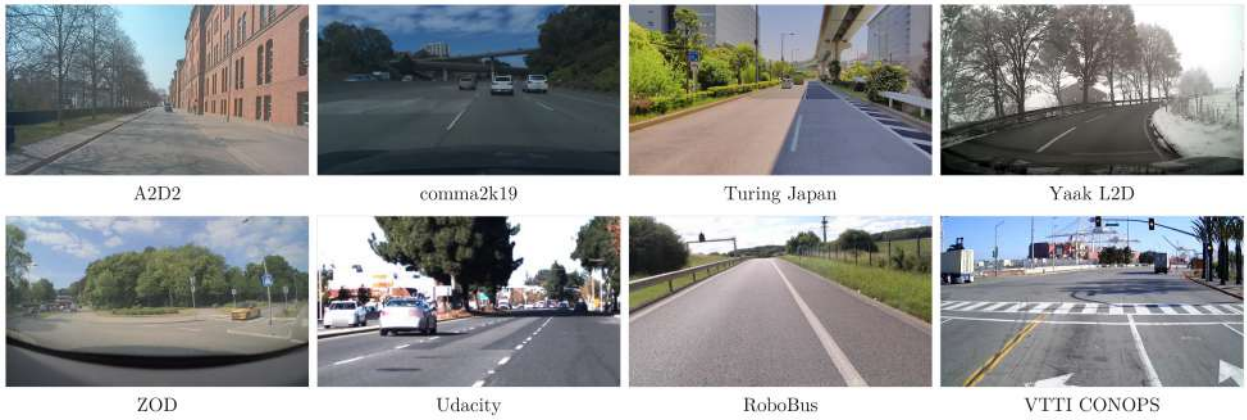


Figure 1: Evaluation datasets. One unmodified front-camera frame from each of the eight datasets, spanning Europe, Japan, and the United States, and covering passenger cars, a van, a coach bus, and a semi-truck. Frames are reproduced with attribution under each dataset's license (see Dataset Attributions).

2 How PACE Works

Figure 2 shows the pipeline at a high level. PACE first recovers the vehicle's motion over time and converts it into vehicle kinematics. A vehicle-dynamics model, configured with the physical properties of the specific vehicle, then splits into two branches: a lateral branch that recovers steering wheel angle, and a longitudinal branch that recovers throttle and brake percentage.

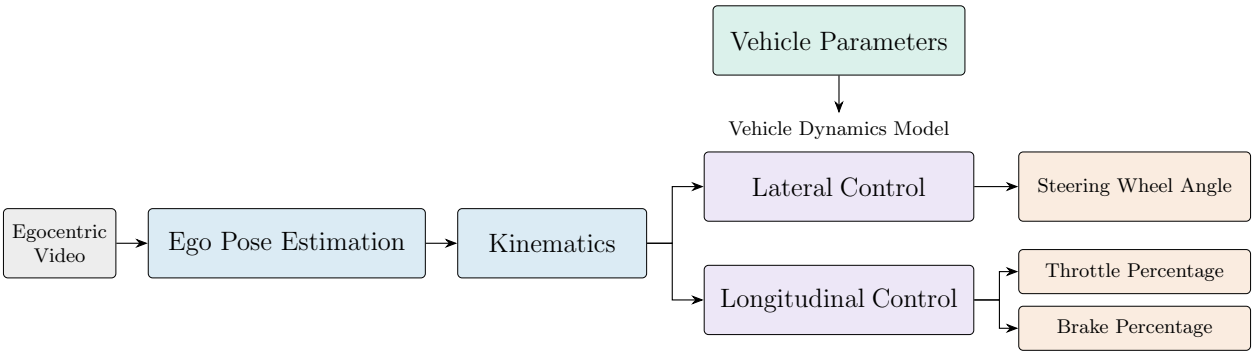


Figure 2: Overview of PACE. The vehicle's motion is converted into kinematics, and a vehicle-dynamics model configured with the vehicle's physical parameters converts those kinematics into steering wheel angle, throttle percentage, and brake percentage.

PACE needs only a basic physical description of the vehicle, such as its size, mass, and steering geometry, which is available from manufacturer specifications for most production vehicles. When the exact model is unknown, defaults for the vehicle class are used. This allows PACE to accurately estimate action labels for a 4.5-meter sedan and a 14-meter coach.

3 Evaluation

Datasets. PACE is evaluated on eight public driving datasets (fig. 1) that include time-synchronized steering wheel angle recorded from the vehicle's CAN bus. Datasets were chosen for diversity in vehicle size, camera hardware and mounting, frame rate, country and side of the road, road type, and driving behavior.

Each dataset is evaluated in full rather than on selected clips, and CAN signals are used only as ground truth for scoring. Results below evaluate PACE's control-extraction stage.

Steering wheel angle. Table 1 reports steering error for each dataset, measured in degrees at the steering wheel. PACE achieves an MAE of 4.16° and an RMSE of 7.50° over nearly 25 million scored frames. Accuracy is highest on highway and arterial driving and lowest on dense urban routes and heavy vehicles, where tight, large-angle maneuvers are more frequent.

Table 1: Steering wheel angle error by dataset, measured at the steering wheel against CAN ground truth for frames with $v > 5\,\mathrm{m/s}$.

Dataset	Scored frames	MAE (°)	RMSE (°)
A2D2 [4]	32,325	1.99	3.03
comma2k19 [5]	2,043,677	2.18	3.11
Turing Japan Open Driving Dataset	2,354,277	1.97	2.83
Yaak L2D	18,873,404	5.70	12.32
Zenseact Open Dataset (ZOD) [1]	284,532	4.42	5.93
Udacity Self-Driving Car Dataset [6]	693,153	1.31	2.24
RoboBus (coach) [7]	690,225	6.67	10.46
VTTI CONOPS (semi-truck)	11,959	9.03	11.02
Equal-dataset average	8 datasets	**4.16**	**7.50**

Large steering angles. PACE handles maneuvers well beyond the small angles of highway driving. On the RoboBus coach, it tracks steering through turns exceeding 850° at the steering wheel, nearly three rotations (figs. 3 and 4).

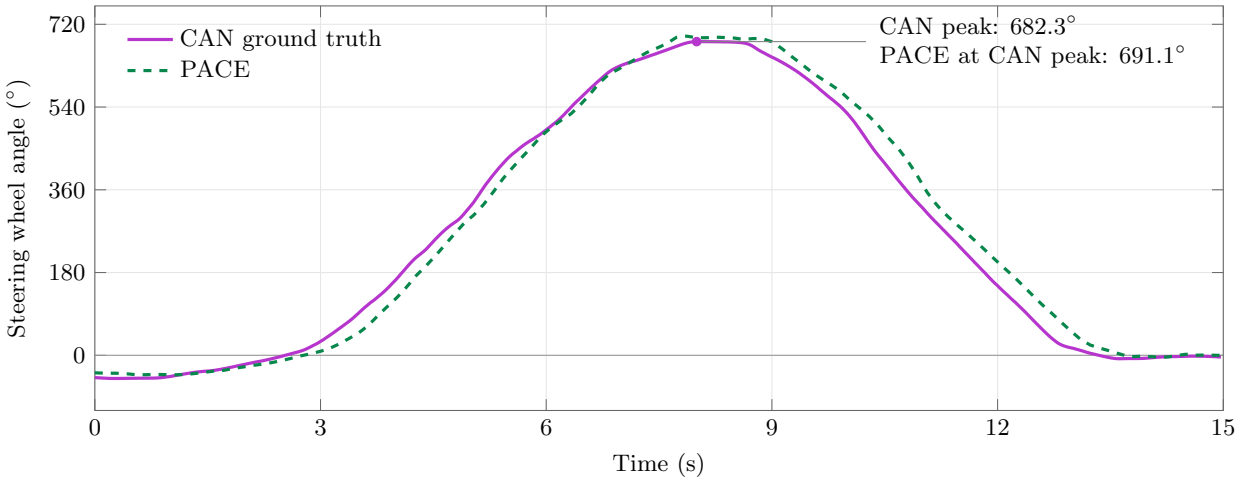


Figure 3: Steering wheel angle through a roundabout on the RoboBus coach (Setra S 431 DT), CAN ground truth vs PACE over a 15 s excerpt (450 frames, all speeds, no additional smoothing). At the recorded peak of 682.3°, PACE reads 691.1°, an error of 8.8°. The peak occurs at 4.6 m/s, below the speed range used for the benchmark results in table 1.

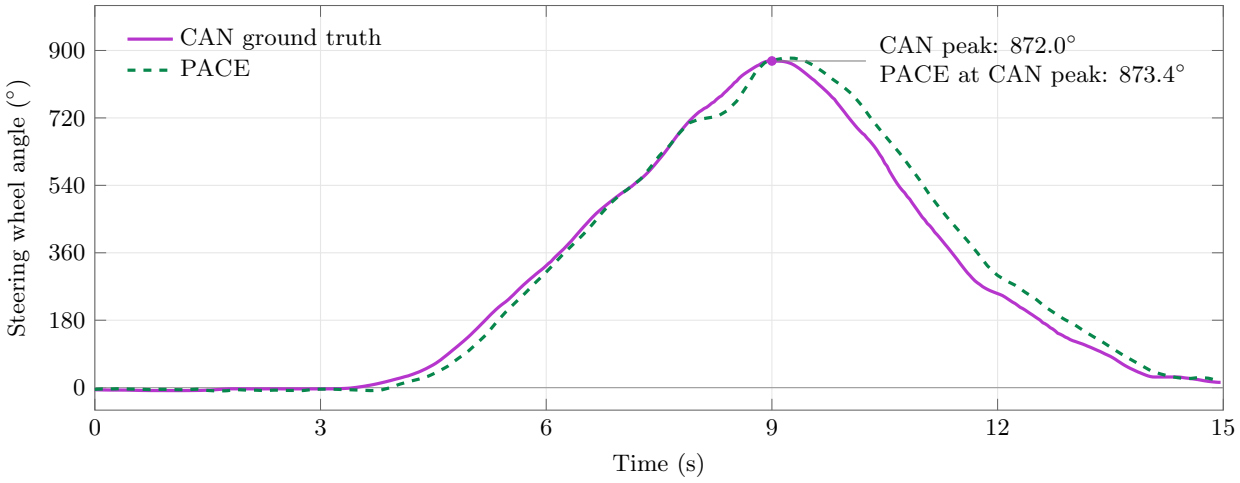


Figure 4: Steering wheel angle on a second RoboBus excerpt (Setra S 431 DT), CAN ground truth vs PACE over a 15 s excerpt (450 frames, all speeds, no additional smoothing). At CAN's recorded peak of 872.0°, nearly three rotations of the steering wheel, PACE estimates an angle of 873.4°, an error of only 1.4°.

Throttle and brake. Unlike steering wheel angle, pedal signals on the CAN bus differ widely between manufacturers: some vehicles log calibrated pedal positions, others log brake pressure, raw values, or only brake-light status. PACE outputs throttle and brake as a percentage of pedal travel, and each pedal is scored only on datasets whose CAN logs provide a comparable signal. Table 2 reports pedal error in percentage points of pedal travel. Throttle is scored on seven datasets, with an equal-dataset MAE of 6.56 and RMSE of 11.48 over more than 29 million frames. Brake is scored on Yaak L2D, the only dataset in the evaluation verified to contain usable brake percentages, with an MAE of 1.95 and RMSE of 5.15 over nearly 25 million frames. The Turing Japan Open Driving Dataset logs no usable pedal signal and is not included. As with steering wheel angle estimation, error is highest on the coach and semi-truck.

Table 2: Throttle and brake error by dataset, in percentage points of pedal travel against CAN ground truth. Dashes mark datasets without a comparable ground-truth signal.

		Throttle (%)		Brake (%)	
Dataset	Scored frames	MAE	RMSE	MAE	RMSE
A2D2 [4]	65,371	2.51	4.51	–	–
comma2k19 [5]	2,360,839	7.21	10.17	–	–
Yaak L2D	24,885,560	5.05	8.37	1.95	5.15
Zenseact Open Dataset (ZOD) [1]	351,055	7.71	11.54	–	–
Udacity Self-Driving Car Dataset [6]	976,609	4.36	6.78	–	–
RoboBus (coach) [7]	847,852	9.96	17.01	–	–
VTTI CONOPS (semi-truck)	39,765	9.10	16.15	–	–
Equal-dataset average	7 datasets	**6.56**	**11.48**	**1.95**†	**5.15**†

† Dataset-wide average is equivalent to Yaak L2D's brake averages as it is the only dataset verified to contain usable brake percentages.

Qualitative examples. Figure 5 shows PACE estimates alongside CAN ground truth over four clips, covering a passenger car on rural and suburban roads, a coach on wet roads, a semi-truck at a terminal gate, and an urban intersection turn.

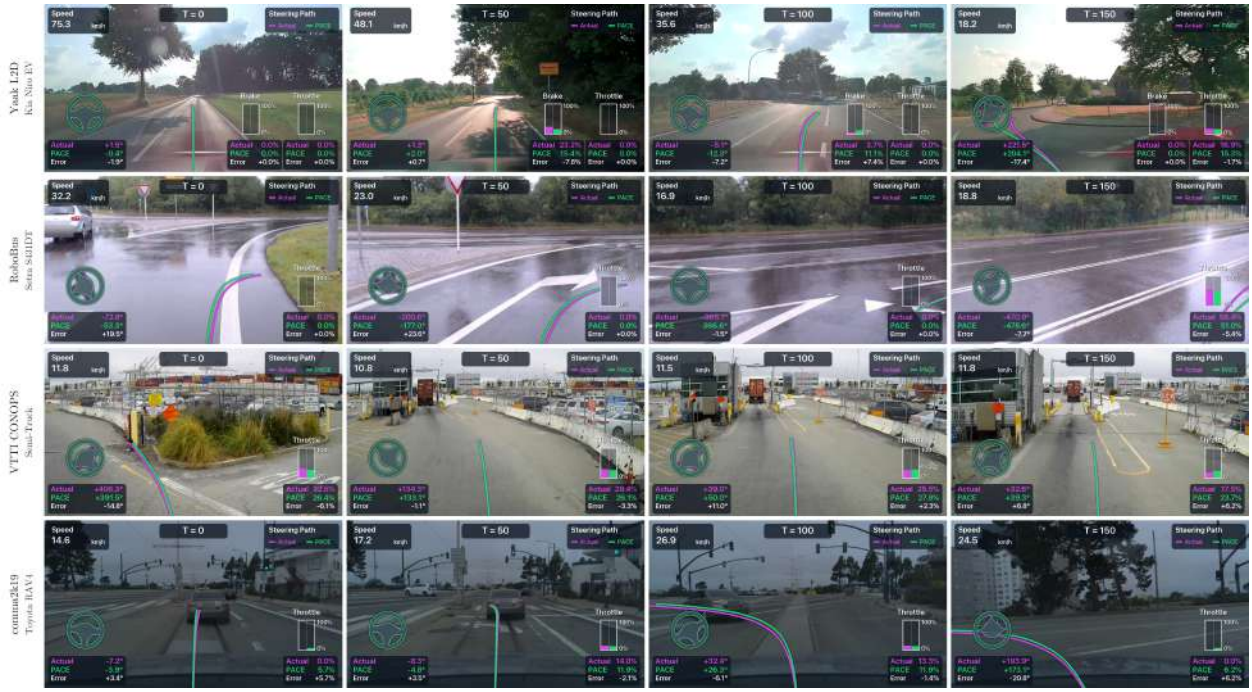


Figure 5: Qualitative examples of PACE control extraction across four driving datasets. Magenta ("Actual") denotes CAN ground truth; green denotes PACE estimates. Columns show camera-frame offsets $T \in \{0, 50, 100, 150\}$ within each clip. Speed is shown in km/h and pedal error in percentage points. Steering paths use an approximate camera projection. Overlays added by Kosmic Worlds.

4 Scope and Limitations

Low speeds. PACE V1.0 produces steering labels above 5 m/s (18 km/h). At very low speeds, such as parking and walking-pace turns, small errors in the recovered motion are strongly amplified, and steering cannot be determined reliably from motion alone. Reversing maneuvers fall almost entirely in this range. Results at lower speeds, where accuracy degrades, are included in the full technical report.

Vehicle description. Because PACE is physics-based, its accuracy depends on how well the vehicle is described. Footage from an unknown vehicle falls back to class defaults, which reduces accuracy.

Articulated vehicles. The current model treats the vehicle as a single rigid body. Tractor-trailers and other articulated vehicles require a multi-body model and are not yet supported.

5 Roadmap

- **Low-speed steering (V1.1).** A learned model that operates alongside the physics pipeline below 5 m/s, covering parking, tight turns, and reversing.
- **Learned throttle and brake (V1.1).** A learned pedal model to complement the physics-based longitudinal branch, trained on CAN-labeled data across vehicle classes.
- **Heavy and articulated vehicles.** Broader validation on buses and trucks, and a multi-body model for tractor-trailers.

- **PACE-R for robotics.** The same principle of recovering control inputs from a physical model of the embodiment, applied to robotic manipulators and mobile robots.

Further Information

A full technical report, including the complete algorithm and mathematical formulation, the vehicle-dynamics and tire modeling, vehicle parameterization, per-dataset analysis, pedal-active results, speed breakdowns, and per-dataset pedal ground-truth handling, is available to partners under NDA. Contact `shaan@kosmicworlds.com`.

Dataset Attributions

- **A2D2: Audi Autonomous Driving Dataset**, © Audi AG. Licensed under CC BY-ND 4.0.
- **comma2k19**, © comma.ai. Licensed under the MIT License.
- **Yaak L2D (Learning to Drive)**, © Yaak. Licensed under the Apache License 2.0.
- **Zenseact Open Dataset (ZOD)**, © 2022 Zenseact AB. Licensed under CC BY-SA 4.0. For this dataset, Zenseact AB has taken all reasonable measures to remove all personally identifiable information, including faces and license plates. To the extent that you like to request the removal of specific images from the dataset, please contact `privacy@zenseact.com`.
- **Udacity Self-Driving Car Dataset**, © 2016 Udacity. Licensed under the MIT License.
- **RoboBus**, University of Luxembourg. Licensed under the MIT License.
- **VTTI CONOPS**, Virginia Tech Transportation Institute. Released under CC0 1.0.
- **Japan Open Driving Dataset**, © Turing Inc. This work uses a dataset produced through the "Support Program for the Development and Demonstration of Autonomous Driving Services to Address Local Mobility Challenges," funded under Japan's Ministry of Economy, Trade and Industry (METI) FY2024 supplementary budget.

References

[1] Mina Alibeigi, William Ljungbergh, Adam Tonderski, Georg Hess, Adam Lilja, Carl Lindström, Daria Motorniuk, Junsheng Fu, Jenny Widahl, and Christoffer Petersson. Zenseact open dataset: A large-scale and diverse multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[2] Bowen Baker, Ilge Akkaya, Peter Zhokhov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (VPT): Learning to act by watching unlabeled online videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[3] Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

[4] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320*, 2020.

[5] Harald Schafer, Eder Santana, Andrew Haden, and Riccardo Biasini. A commute in data: The comma2k19 dataset. *arXiv preprint arXiv:1812.05752*, 2018.

[6] Udacity. Udacity self-driving car dataset. `https://github.com/udacity/self-driving-car`, 2016.

[7] Georgios Varisteas, Raphaël Frank, and François Robinet. RoboBus: A diverse and cross-border public transport dataset. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2021.