International Journal of Applied Operational Research
Vol. 14, No. 4, pp. 63-83, Autumn 2026

Journal homepage: ijorlu.liau.ac.ir

Mathematical optimization and configuration analysis of multimodal end-to-end deep learning for joint steering and speed control

M. Aghamohammadi, K. Khoshhal Roudposhti*

Received: 5 July 2026; **Accepted:** 25 August 2026

Abstract End-to-end autonomous driving can be interpreted as a high-dimensional nonlinear optimization problem in which heterogeneous sensory observations are mapped directly to continuous vehicle-control commands. This paper provides an optimization-oriented mathematical interpretation and configuration-sensitivity analysis of joint steering-angle and vehicle-speed prediction using real-world camera and LiDAR observations across two successive phases of model development. In Phase I, Xception and ResNeXt are evaluated under six RGB/LiDAR and temporal configurations to quantify the effects of depth representation and Long Short-Term Memory (LSTM). In Phase II, complementary backbones are combined through stacking, followed by LSTM and attention, allowing the incremental contributions of ensemble learning, temporal modeling, and attention to be analyzed. The reported experiments use a fixed-weight scalarized steering-speed loss optimized with Adam; adaptive weighting, gradient-conflict mitigation, and bilevel optimization are discussed as mathematical extensions and future research directions rather than experimentally implemented methods. A post-hoc Balanced Accuracy Index is introduced to summarize the two control outputs without altering the original training objective, and horizon-wise RMSE is used for contextual comparison with a related camera-LiDAR E2E baseline. Phase I shows that adding LSTM to RGB-only models increases steering accuracy by 9.2 percentage points for Xception and 10.5 percentage points for ResNeXt. Phase II improves performance substantially: the ResNeXt-Xception-Attention-LSTM configuration reaches 94.91% steering accuracy and 94.09% speed accuracy. Its RMSE values across 1-5 s horizons are 0.6116, 0.8626, 1.0866, 1.3005, and 1.5055, respectively, with reductions of up to 19.8% for M6 relative to the selected published baseline under non-identical experimental protocols. The results show that the attained solution is sensitive to architecture, input representation, temporal context, and fusion strategy.

Keyword: Multi-Objective Optimization, End-To-End Autonomous Driving, Multimodal Fusion, LSTM, Attention, Steering And Speed Control.

1 Introduction

Autonomous driving combines perception, temporal reasoning, decision making, and vehicle control under uncertain and rapidly changing environmental conditions. Deep learning has accelerated the development of data-driven driving systems by enabling nonlinear mappings

* **Corresponding Author. (✉)**
E-mail: Ka.khoshhal@iau.ac.ir (K. Khoshhal Roudposhti)

M. Aghamohammadi
Department of Mechanical Engineering, BaA.C., Islamic Azad University, Bandar Anzali, Iran

K. Khoshhal Roudposhti
Department of Computer Engineering, La.C., Islamic Azad University, Lahijan, Iran

between high-dimensional sensory observations and control actions to be learned from demonstrations [1,2]. Within this context, end-to-end (E2E) learning attempts to replace part or all of the conventional hand-engineered driving stack with a trainable mapping from sensory input to control output [3–5].

Early E2E systems demonstrated that steering commands could be predicted directly from road images [5]. Later studies introduced temporal dependencies, conditional imitation learning, attention, and multimodal sensor fusion [6–9]. Camera and LiDAR data are particularly complementary: RGB images provide texture, lane, object, and semantic context, while LiDAR provides geometric and depth information. The LiDAR-Video driving dataset of Chen et al. provides synchronized real-world imagery, point clouds, and driver-control labels suitable for behavioral learning [10]. Related camera–LiDAR E2E studies have confirmed that the representation and fusion of these modalities can strongly affect the learned policy [11,12].

Temporal context is equally important. Steering and vehicle speed at time t are not functions of a single isolated image; they depend on recent scene evolution and vehicle behavior. Recurrent networks, particularly LSTM, provide a mechanism for preserving relevant information across consecutive observations [13]. This motivates explicit comparison between purely convolutional models and CNN–LSTM models under otherwise similar sensory configurations.

From a mathematical standpoint, E2E learning is an optimization problem over a large and non-convex parameter space. Adam is widely used because its adaptive moment estimates provide parameter-specific update scaling and work well with stochastic gradients [14]. Nevertheless, optimizer selection alone does not determine model quality. The geometry of the loss surface depends on the architecture, the representation of sensory inputs, the temporal window, and the relative importance assigned to multiple control objectives.

Joint steering and speed prediction is especially suited to a multi-objective interpretation. Steering and speed have different physical units, different error distributions, and may improve at different rates during training. A simple weighted loss is a scalarization of two objectives, while more advanced multi-task optimization methods explicitly consider gradient conflicts, Pareto stationarity, or adaptive task weighting [15–22]. These methods motivate a richer interpretation of the driving-control problem even when the underlying experiments were trained with a standard optimizer.

Recent E2E literature has moved toward unified planning-oriented networks, multimodal transformers, and robust fusion [12,23–28]. UniAD jointly optimizes a full-stack driving formulation around planning [25], while TransFuser and BEVFusion demonstrate the importance of complementary camera and LiDAR representations [12,27]. The comprehensive survey of Zhao et al. [23] and the end-to-end multimodal EMMA framework of Hwang et al. [24] further illustrate the rapid evolution toward integrated, multimodal autonomous-driving systems. Contemporary surveys identify multimodality, robustness, interpretability, and closed-loop evaluation as persistent challenges for E2E driving [26]. The present study remains grounded in real-world camera-LiDAR behavioral data and focuses specifically on how the attained optimization outcome changes as model configuration becomes progressively richer.

The principal contribution of this paper is therefore not the introduction or experimental validation of a new numerical optimizer. Instead, the paper (i) formulates steering and speed prediction as a coupled optimization problem; (ii) analyzes Phase-I sensitivity to architecture, depth representation, and temporal modeling; (iii) incorporates the doctoral dissertation's Phase-II ensemble, LSTM, and attention results; (iv) derives post-hoc multi-output summary measures and horizon-wise RMSE comparisons from the reported results; and (v) positions these experiments within contemporary multi-objective optimization literature. The advanced

adaptive-weighting, gradient-balancing, and bilevel optimization concepts discussed later are not claimed to have been used in the original training and are presented as future extensions.

This study builds upon the experimental framework developed in the first author's doctoral dissertation [29], from which the core model architectures, training configurations, and associated experimental results were derived. The present manuscript extends that experimental foundation through an optimization-oriented mathematical interpretation, systematic configuration-sensitivity analysis, a post-hoc Balanced Accuracy Index (BAI) for joint assessment of steering and speed, and horizon-wise RMSE analysis. This distinction is stated explicitly to separate the provenance of the experimental material from the additional analytical contributions of the present study.

2 Related work

2.1 End-to-end driving and behavioral learning

Direct-perception and E2E driving research progressively reduced dependence on manually engineered intermediate representations. DeepDriving linked visual input to driving affordances [3], while large-scale video and direct steering models demonstrated that neural networks can learn useful control mappings from visual demonstrations [4,5]. Conditional imitation learning later introduced high-level commands to disambiguate behavior at intersections and route choices [7].

More recent E2E systems are broader. UniAD organizes perception, prediction, and planning around a common planning-oriented objective [25]. Surveys of E2E driving show that modern systems increasingly rely on unified representations, richer temporal reasoning, and multi-task optimization, while robustness and interpretability remain unresolved [26].

2.2 Multimodal camera–LiDAR fusion

Multimodal sensing is motivated by complementary failure modes and information content. Chen et al. introduced synchronized LiDAR-video driving data with behavior labels [10]. Xiao et al. examined multimodal E2E driving and showed that the way image and depth information are represented affects performance [9]. Park et al. evaluated CNN-based E2E driving with camera–LiDAR fusion in a real-world setting [11].

Transformer-era studies further strengthened this direction. TransFuser uses attention to exchange global context between camera and LiDAR features for E2E driving [12]. BEVFusion develops a simple fusion framework with improved robustness to LiDAR malfunction [27]. These studies support the general conclusion that sensor fusion should be designed together with model structure rather than treated as a neutral preprocessing step.

Navarro et al. [30] also addressed joint prediction of vehicle speed and steering-wheel angle using six end-to-end DNN architectures and combinations of front-camera imagery with vehicle-dynamics signals. The present study shares the joint-output objective but differs in its sensory and architectural design: it emphasizes camera-LiDAR multimodality, compares Xception and ResNeXt backbones under alternative depth representations, and evaluates a staged progression from stacking to LSTM and attention. The comparison is therefore methodological rather than a direct numerical benchmark, because the datasets, input modalities, architectures, and evaluation protocols differ.

2.3 Temporal modeling and attention

Driving actions are sequential. Eraqi et al. explicitly considered temporal dependencies for steering prediction [6], while the classical LSTM formulation provides gated memory to preserve useful information over time [13]. Attention mechanisms can further prioritize informative features or temporal states. Ishihara et al. demonstrated attention-aware multi-task learning in E2E driving [8].

The dissertation-derived model progression considered here follows a similar conceptual path: single-frame convolutional baselines are first tested, LSTM is then introduced, and the final phase combines ensemble learning with LSTM and attention. This staged design provides a useful empirical basis for optimization-oriented ablation analysis.

2.4 Multi-task and multi-objective optimization

Multi-output learning naturally raises the question of how task losses should be balanced. Kendall et al. proposed uncertainty-based weighting [15], Groenendijk et al. proposed coefficient-of-variation weighting [16], and Chennupati et al. examined geometric aggregation for multi-task learning [17].

Sener and Koltun formalized multi-task learning as multi-objective optimization and connected training to Pareto-optimal solutions [18]. PCGrad modifies task gradients to reduce destructive interference [19], CAGrad seeks conflict-averse directions [20], and FAMO performs fast adaptive task weighting [22]. At the same time, large-scale empirical analysis has shown that sophisticated multi-task optimizers do not automatically outperform well-tuned weighted-loss baselines [21]. This is important for the present paper: The mathematical formulation motivates future Pareto-aware or adaptive methods, but the reported experimental results are deliberately interpreted using the training procedure actually performed.

3 Materials and methods

3.1 Research framework and two-phase experimental program

As shown in Fig. 1, the study is organized as a two-phase experimental program. Phase I evaluates individual Xception and ResNeXt backbones under six input/temporal configurations. Phase II uses the lessons from Phase I to construct ensemble models that combine complementary backbones and progressively introduce LSTM and attention. This structure allows the optimization outcome to be analyzed at both configuration level and model-composition level.

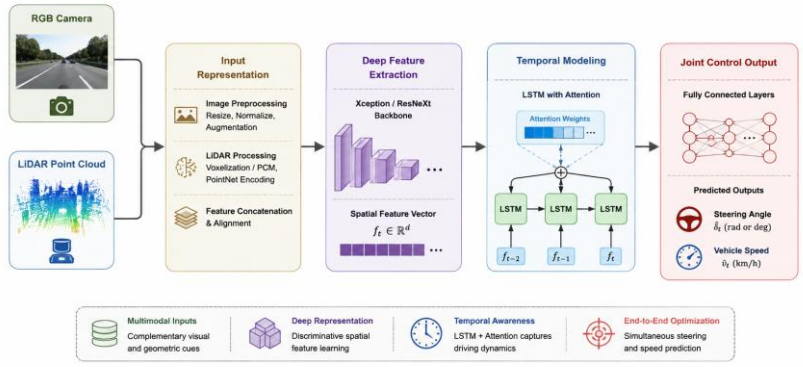

Fig. 1 Optimization-oriented E2E framework for multimodal steering and speed prediction

3.2 Real-world multimodal dataset

The experiments use a synchronized real-world LiDAR-video driving dataset [10]. The processed research dataset contains 38,880 image frames with corresponding LiDAR files and driver-behavior labels. The recorded environments include local roads, main roads, intersections, bridges, school zones, mountainous roads, and scenes with varying pedestrian density. This diversity is valuable because the optimization process is exposed to a wider distribution of visual and geometric conditions than would be available from a narrow single-scenario dataset.

For sample i, the target and predicted control vectors are

$$y_i = [\delta_i, v_i]^T \quad (1)$$

$$\hat{y}_i = [\hat{\delta}_i, \hat{v}_i]^T \quad (2)$$

where δ denotes steering angle and v denotes vehicle speed. Both outputs are continuous, so the learning problem is treated as multi-output regression.

3.3 Input representation and depth processing

The multimodal observation contains RGB information and a LiDAR-derived representation:

$$x_t = [x_t^{RGB}, x_t^L] \quad (3)$$

Phase I examines RGB alone, RGB with PointNet-processed LiDAR, and RGB with point-cloud mapping (PCM). These alternatives are important because the same physical LiDAR measurements can generate different optimization behavior after representation changes. PointNet preserves point-set structure, while PCM converts geometric information into an image-like representation that can be processed by convolutional backbones.

3.4 Spatial and temporal representation

Xception and ResNeXt are the principal convolutional backbones. Xception uses depthwise separable convolution [31], whereas ResNeXt uses aggregated residual transformations and cardinality [32]. The spatial feature mapping is represented as

$$f_t = F\theta c(x_t) \quad (4)$$

When temporal modeling is enabled, the spatial features are passed through an LSTM:

$$h_t = F\theta L(f_t, h_{t-1}) \tag{5}$$

The final Phase-II architecture combines two feature extractors through stacking, then applies temporal modeling and attention. Figure 2 shows the representative best-performing ordering used in the final comparison.

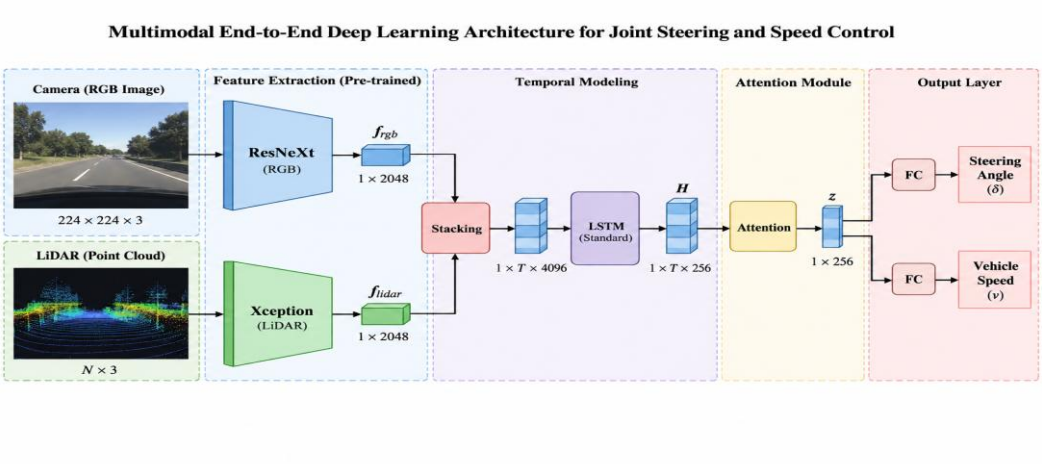

Fig. 2 Representative Phase-II architecture: ResNeXt–Xception stacking followed by LSTM and attention

3.5 Coupled steering–speed optimization

Let the complete model be $F(\cdot;\theta)$, where θ includes the trainable convolutional, recurrent, attention, and output-layer parameters:

$$\hat{y}_i = F(x_i; \theta) \tag{6}$$

$$\theta^* = \arg\min_\theta J(\theta) \tag{7}$$

Separate squared-error objectives are defined for the two physical outputs:

$$L\delta(\theta) = (1/N) \sum_{i=1}^{N} (\delta_i - \hat{\delta}_i)^2 \tag{8}$$

$$Lv(\theta) = (1/N) \sum_{i=1}^{N} (v_i - \hat{v}_i)^2 \tag{9}$$

A regularized scalarized objective can be written as

$$J(\theta) = \lambda\delta\, L\delta(\theta) + \lambda v\, Lv(\theta) + \lambda r\, R(\theta) \tag{10}$$

$$\lambda\delta \geq 0,\ \lambda v \geq 0,\ \lambda r \geq 0,\ \lambda\delta + \lambda v = 1 \tag{11}$$

The scalarized form is a mathematically explicit representation of the coupled prediction problem. Here, $R(\theta)$ denotes a generic regularization term and $\lambda r \geq 0$ is its corresponding weighting coefficient in the general formulation. No separate regularization term $R(\theta)$ or numerical λr was independently implemented or tuned in the reported experimental pipeline; the experiments used the fixed-weight steering-speed scalarization optimized with Adam. Accordingly, the formulation is used to clarify the broader optimization structure rather than to imply an unreported regularization experiment.

3.6 Scale normalization and objective comparability

Steering and speed are measured in different units. When two losses with different numerical scales are combined, the larger-magnitude loss can dominate gradient updates even if it is not more important physically. A generic normalized loss can be written as

$$\tilde{L}k = Lk / (sk + \varepsilon), \quad k \in \{\delta, v\} \tag{12}$$

Here, k is an index rather than a calculated numerical variable, with $k \in \{\delta, v\}$: $k = \delta$ denotes the steering objective and $k = v$ denotes the speed objective. Accordingly, Lk is the task-specific loss and sk is its associated positive reference scale used for normalization. This normalization was not part of the original experimental pipeline; it is included as a rigorous extension for future adaptive multi-objective work.

3.7 Gradient Interaction Between Objectives

$$\nabla\theta J = \lambda\delta \nabla\theta L\delta + \lambda v \nabla\theta Lv + \lambda r \nabla\theta R \tag{13}$$

$$\cos \varphi = (\nabla L\delta^{T} \nabla Lv) / (\|\nabla L\delta\|_2 \|\nabla Lv\|_2) \tag{14}$$

A positive cosine indicates local alignment, a value near zero indicates weak interaction, and a negative value indicates gradient conflict. This diagnostic connects the scale-comparability discussion above to gradient-aware methods such as PCGrad and CAGrad [19,20]. Per-task gradients were not archived in the original experiments, so no retrospective gradient-conflict values are inferred here.

3.8 Adam-Based Parameter Optimization

$$g_t = \nabla\theta J_t(\theta_{t-1}) \tag{15}$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{16}$$

$$s_t = \beta_2 s_{t-1} + (1 - \beta_2)(g_t \odot g_t) \tag{17}$$

$$\hat{m}_t = m_t/(1 - \beta_1^t), \quad \hat{s}_t = s_t/(1 - \beta_2^t) \tag{18}$$

$$\theta_t = \theta_{t-1} - \eta\, \hat{m}_t/(\sqrt{\hat{s}_t} + \varepsilon) \tag{19}$$

Adam adapts effective step magnitudes using first- and second-moment information [14]. In the reported experiments, Adam was the optimizer used with the fixed-weight scalarized loss. It is therefore treated here as one component of the overall optimization system rather than as the sole explanation for performance differences across configurations.

3.9 Evaluation Metrics

$$\mathrm{RMSE}(y, \hat{y}) = \sqrt{[(1/N) \sum_{i=1}^{N} (y_i - \hat{y}_i)^2]} \tag{20}$$

$$\mathrm{Acc}\delta = (1/N) \sum_{i=1}^{N} \mathbb{1}(|\delta_i - \hat{\delta}_i| \le 6^\circ) \tag{21}$$

$$\mathrm{RMSE}(y, \hat{y}) = \sqrt{[(1/N) \sum_{i=1}^{N} (y_i - \hat{y}_i)^2]} \tag{20}$$

$$\mathrm{Accv} = (1/N) \sum_{i=1}^{N} \mathbb{1}(|v_i - \hat{v}_i| \leq 5 \text{ km/h}) \tag{22}$$

RMSE measures continuous deviation, while tolerance accuracy reports the fraction of predictions within the adopted practical bounds.

3.10 Post-Hoc balanced accuracy and relative gains

$$\mathrm{BAI} = 2\, \mathrm{Acc}\delta\, \mathrm{Accv} / (\mathrm{Acc}\delta + \mathrm{Accv}) \tag{23}$$

$$\mathrm{Grel} = [(\mathrm{Mnew} - \mathrm{Mref})/\mathrm{Mref}] \times 100\% \tag{24}$$

$$\mathrm{RRMSE}(h) = [\mathrm{RMSEref}(h) - \mathrm{RMSEmodel}(h)] / \mathrm{RMSEref}(h) \times 100\% \tag{25}$$

BAI is the harmonic mean of steering and speed accuracy. It is used only for post-hoc analysis and is not a training objective.

3.11 Pareto interpretation

$$L(\theta) = [L\delta(\theta), Lv(\theta)]^T \tag{26}$$

$$\theta a \prec \theta b \text{ iff } Lk(\theta a) \leq Lk(\theta b)\ \forall k, \text{ with strict inequality for at least one objective} \tag{27}$$

As illustrated in Fig. 3, weighted scalarization selects one compromise, while Pareto analysis characterizes non-dominated alternatives. Figure 3 is schematic and is not presented as experimental data.

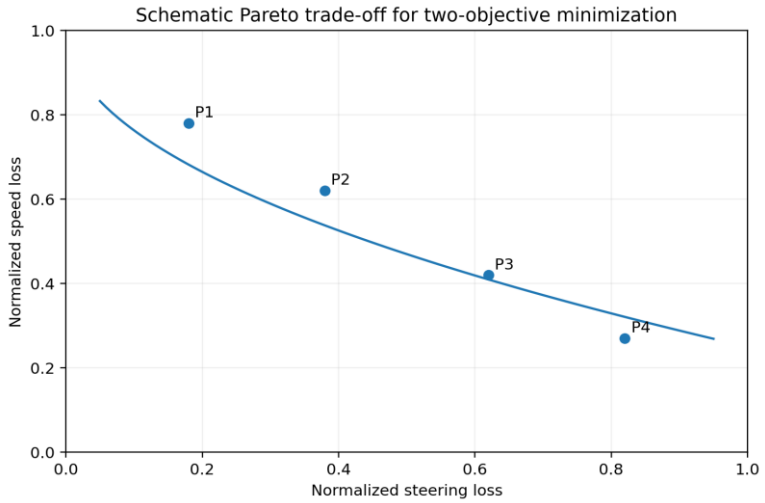

Fig. 3 Schematic Pareto trade-off for a two-objective steering–speed minimization problem

3.12 Hyperparameter-level optimization

$$\xi = [\eta, \beta_1, \beta_2, B, \lambda\delta, \lambda v, \lambda r]^T \quad (28)$$

$$\xi^* = \arg\min_{\xi \in \Omega} \Psi(\theta^*(\xi), \xi) \quad (29)$$

This bilevel notation separates trainable network parameters from training hyperparameters. It provides a formal basis for the dissertation's proposed future combination of global hyperparameter search with Adam. Tables 1 and 2 summarize the principal notation and Phase-I experimental configurations, respectively.

Table 1 Principal mathematical notation

Symbol	Definition
N	Number of labeled samples
δ, $\hat{\delta}$	Ground-truth and predicted steering angle
v, $\hat{v}$	Ground-truth and predicted speed
θ	Trainable network parameters
ξ	Hyperparameter vector
$L\delta$, Lv	Steering and speed losses
$\lambda\delta$, λv	Objective weights
η	Learning rate
β_1, β_2	Adam decay parameters
BAI	Balanced Accuracy Index
$R(\theta)$	Generic regularization term in the general formulation
λr	Non-negative regularization weight in the general formulation
P	Observed model performance
O	Optimizer and loss-setting
A	Network architecture
X	Input representation
T	Temporal-modeling choice
F	Fusion strategy

Table 2 Phase-I experimental configurations

Config.	Input	Depth representation	LSTM
A1	RGB	None	No
B1	RGB + LiDAR	PointNet	No
C1	RGB + LiDAR	PCM	No
A2	RGB	None	Yes
B2	RGB + LiDAR	PointNet	Yes
C2	RGB + LiDAR	PCM	Yes

4 Results and discussion

4.1 Phase-I Xception results

Table 3 Phase-I Xception tolerance accuracy

Config.	Speed (%)	Steering (%)
A1	69.7	70.1
B1	79.4	81.2
C1	68.8	76.4
A2	74.2	79.3
B2	77.6	80.8
C2	79.1	83.4

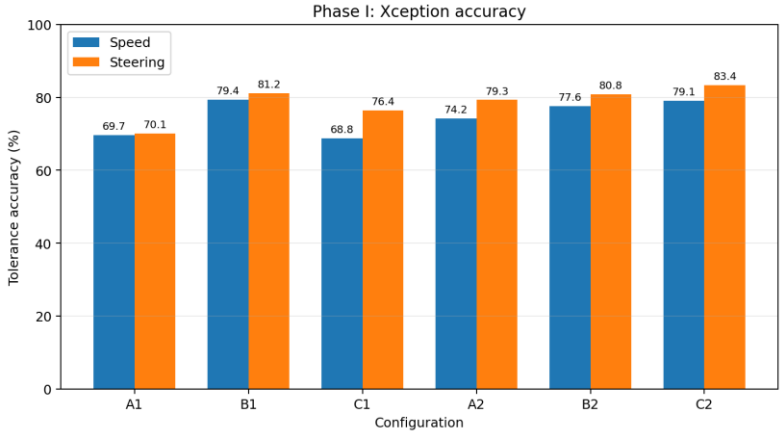

Fig. 4 Phase-I Xception performance across input and temporal configurations.

As summarized in Table 3 and illustrated in Fig. 4, Xception A1, which uses RGB without temporal modeling, achieves 69.7% speed accuracy and 70.1% steering accuracy. A2 isolates the effect of LSTM on the same RGB-only input and raises the corresponding values to 74.2% and 79.3%. B1 shows that PointNet-derived depth can improve the convolutional-only configuration, reaching 79.4% speed and 81.2% steering. C2 yields the strongest Xception steering result in Phase I at 83.4%.

4.2 Phase-I ResNeXt results

Table 4 Phase-I ResNeXt tolerance accuracy

Config.	Speed (%)	Steering (%)
A1	72.6	68.2
B1	78.5	79.9
C1	71.7	69.3
A2	75.8	78.7
B2	75.9	79.2
C2	81.6	84.3

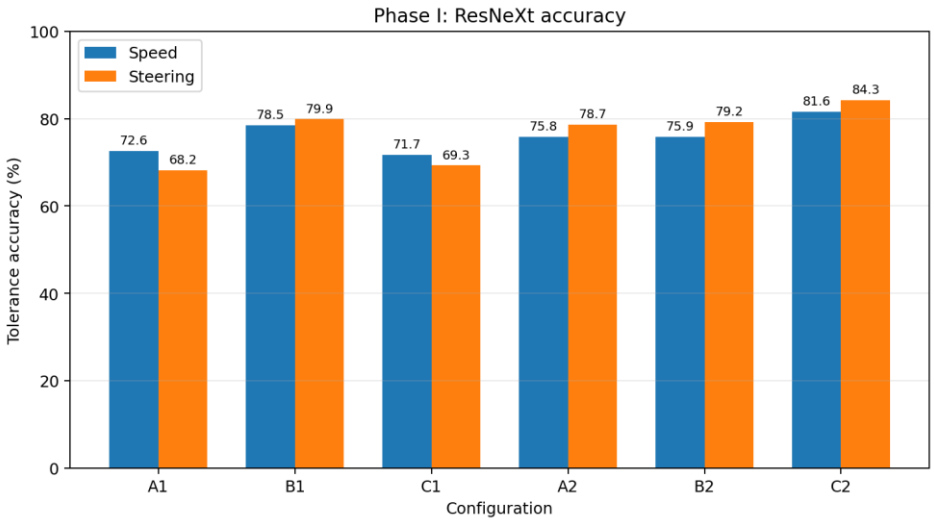

Fig. 5 Phase-I ResNeXt performance across input and temporal configurations.

As summarized in Table 4 and illustrated in Fig. 5, ResNeXt shows the same broad sensitivity to data representation and temporal context. C2 produces the best simultaneous Phase-I result, reaching 81.6% speed accuracy and 84.3% steering accuracy. The fact that neither backbone uniformly dominates every configuration supports the later decision to preserve both architectures through ensemble learning rather than selecting a single winner.

4.3 Quantifying the effect of temporal modeling

The A1-to-A2 comparison provides the cleanest estimate of temporal-model contribution because input modality is held fixed. Xception gains 4.5 percentage points in speed and 9.2 points in steering. ResNeXt gains 3.2 points in speed and 10.5 points in steering. The larger steering gains suggest that the lateral-control output is particularly sensitive to temporal context in these experiments.

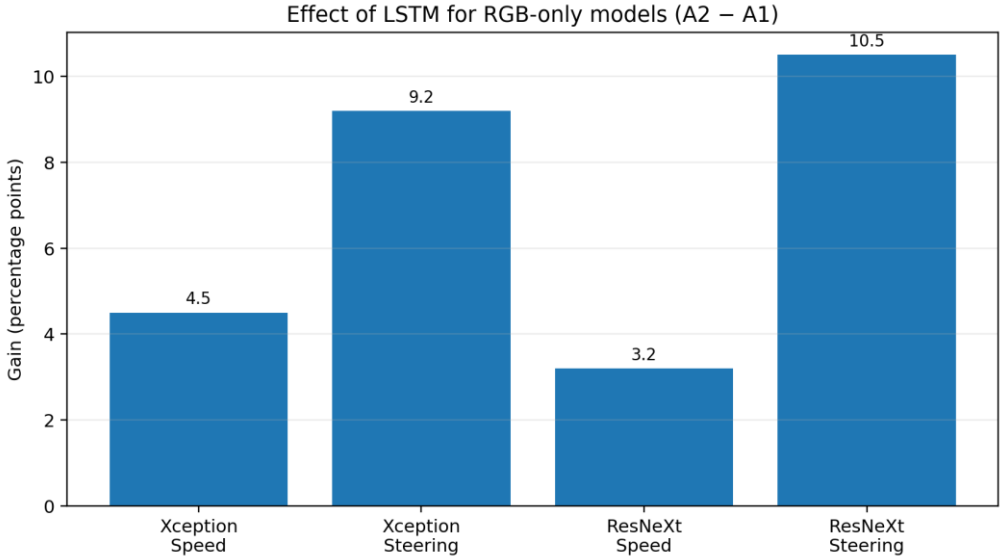

Fig. 6 Accuracy gain from adding LSTM to RGB-only configurations.

As shown in Fig. 6, these values should be interpreted as configuration-specific empirical gains rather than universal causal effects. Nevertheless, they provide a strong experimental rationale for retaining temporal modeling in Phase II.

4.4 Phase-II model progression

Table 5 Phase-II accuracy of the six final models

ID	Model	Speed (%)	Steering (%)
M1	Xception–ResNeXt	83.12	90.01
M2	ResNeXt–Xception	85.69	86.49
M3	Xception–ResNeXt–LSTM	91.63	92.33
M4	ResNeXt–Xception–LSTM	93.16	93.58
M5	Xception–ResNeXt–Attention–LSTM	93.51	94.75
M6	ResNeXt–Xception–Attention–LSTM	94.09	94.91

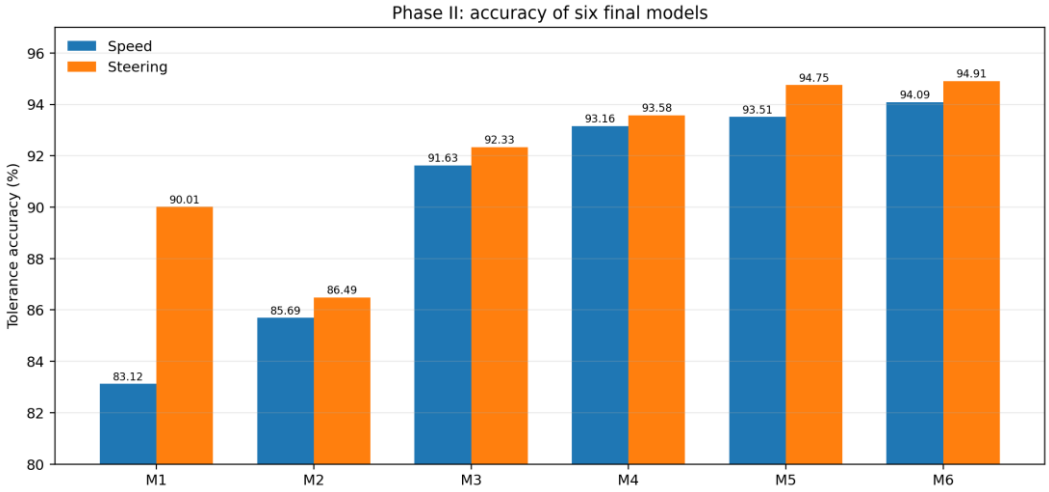

Fig. 7 Phase-II speed and steering accuracy for the six final models

As summarized in Table 5 and illustrated in Fig. 7, Phase II begins with two stacked Xception–ResNeXt orderings, adds LSTM in the next two models, and finally adds attention in the last two models. The best model, M6 (ResNeXt–Xception–Attention–LSTM), reaches 94.09% speed accuracy and 94.91% steering accuracy. M5 is close, at 93.51% and 94.75%, respectively. The results indicate that both final orderings benefit from temporal modeling and attention, while the ResNeXt-first ordering provides the strongest balanced result.

4.5 Stage-wise contribution of LSTM and attention

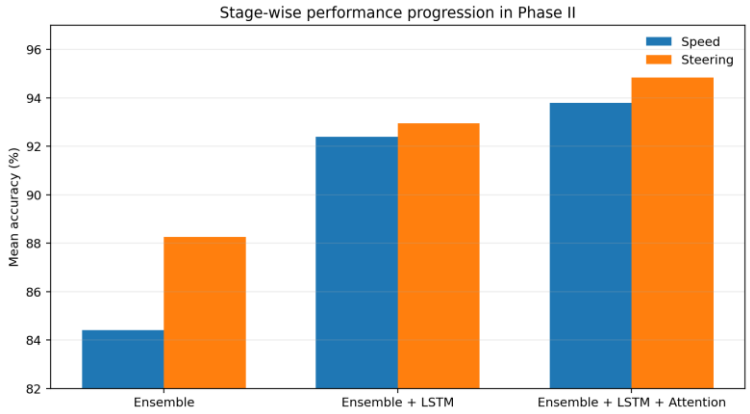

Fig. 8 Mean Phase-II accuracy after stacking only, stacking + LSTM, and stacking + LSTM + attention.

As shown in Fig. 8, averaging the two model orderings within each stage gives speed accuracies of 84.41%, 92.39%, and 93.80% for ensemble-only, ensemble + LSTM, and ensemble + LSTM + attention, respectively. The corresponding steering averages are 88.25%, 92.96%, and 94.83%. This stage-wise view reduces sensitivity to one particular backbone ordering and shows a clear performance progression.

4.6 Balanced accuracy across outputs

Table 6 Post-hoc Balanced Accuracy Index

Model	BAI (%)
M1	86.43
M2	86.09
M3	91.98
M4	93.37
M5	94.13
M6	94.50

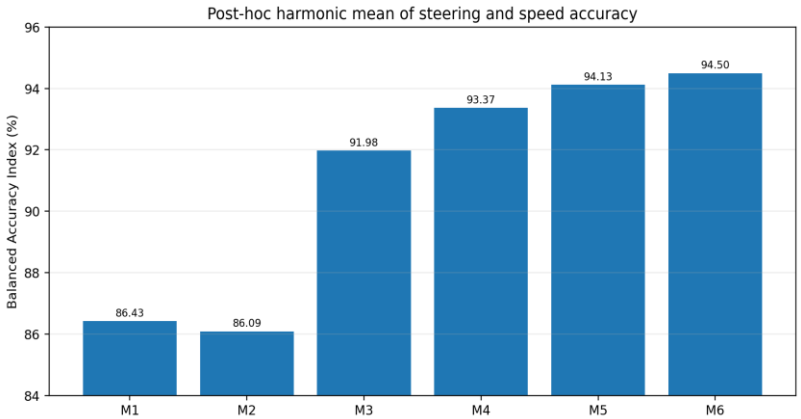

Fig. 9 Balanced accuracy index across Phase-II models

As summarized in Table 6 and illustrated in Fig. 9, BAI increases from approximately 86.43% for M1 to 94.50% for M6. Because the harmonic mean penalizes imbalance between the two outputs, the high M6 score confirms that its performance is not driven by steering accuracy alone.

4.7 Steering–speed joint improvement geometry

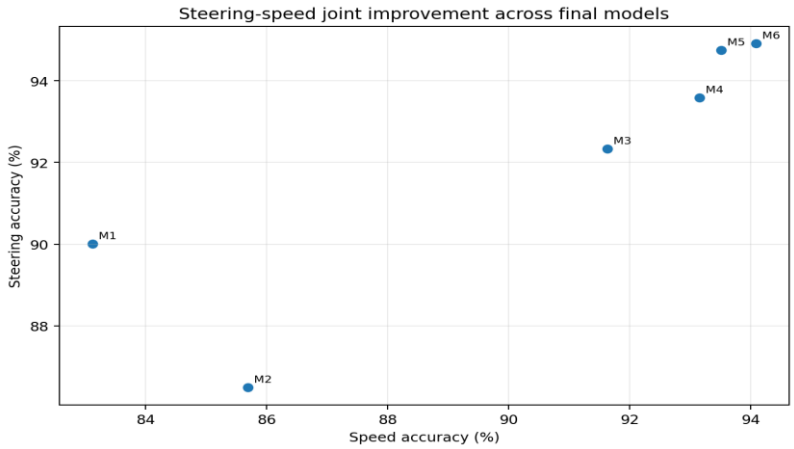

Fig. 10 Steering–speed joint improvement across the six Phase-II models.

As shown in Fig. 10, the final models move generally upward and to the right in the two-dimensional accuracy plane. M6 occupies the strongest observed region. This joint-improvement pattern indicates that steering and speed accuracy generally co-move across the evaluated architecture progression rather than exhibiting a clear empirical trade-off. This evaluation-time behavior does not prove that steering and speed gradients were always aligned during training, but it indicates that the selected configurations can improve both outputs simultaneously.

4.8 Improvement relative to the best phase-I model

Using the best simultaneous Phase-I ResNeXt C2 result as a reference, M6 improves speed accuracy from 81.6% to 94.09% and steering accuracy from 84.3% to 94.91%. These correspond to relative gains of 15.31% and 12.59%, respectively. In absolute terms, the gains are 12.49 percentage points for speed and 10.61 percentage points for steering.

4.9 Horizon-wise RMSE

Table 7 RMSE across 1–5 s prediction horizons

Model	1 s	2 s	3 s	4 s	5 s
Park et al. ResNet50–InceptionV3	0.6240	0.9113	1.2170	1.6210	1.6774
Xception–ResNeXt–Attention–LSTM	0.6330	0.9044	1.2219	1.5621	1.8848
ResNeXt–Xception–Attention–LSTM	0.6116	0.8626	1.0866	1.3005	1.5055

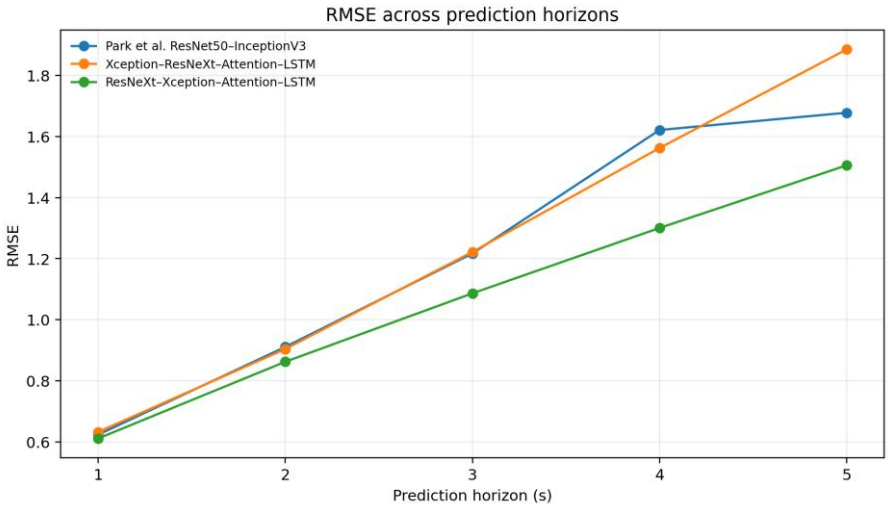

Fig. 11 Horizon-wise RMSE comparison for the two attention-based final models and the Park et al. baseline.

As summarized in Table 7 and illustrated in Fig. 11, the mean RMSE across the five horizons is 1.2101 for the selected Park baseline, 1.2412 for Xception-ResNeXt-Attention-LSTM, and 1.0734 for ResNeXt-Xception-Attention-LSTM. Thus, M6 has the lowest mean horizon-wise RMSE among these three reported curves. However, the Xception-first M5 configuration reaches an RMSE of 1.8848 at 5 s, which is higher than the Park et al. value of 1.6774 at the same reported horizon. This crossover suggests greater long-horizon error accumulation for that backbone ordering and shows that the RMSE advantage should not be generalized to both final configurations. The RMSE-reduction claim in the following section is therefore restricted to M6.

These cross-study values are provided for contextual comparison only. The Park et al. study and the present experiments were not necessarily conducted with identical train/test partitions, tolerance definitions, preprocessing procedures, or prediction-horizon protocols. Consequently, the numerical differences should not be interpreted as a controlled head-to-head benchmark under identical experimental conditions.

4.10 Relative RMSE reduction versus Park et al.

Table 8 Relative RMSE reduction of M6 versus Park et al.

Horizon	RMSE reduction
1s	1.99%
2s	5.34%
3s	10.71%
4s	19.77%
5s	10.25%

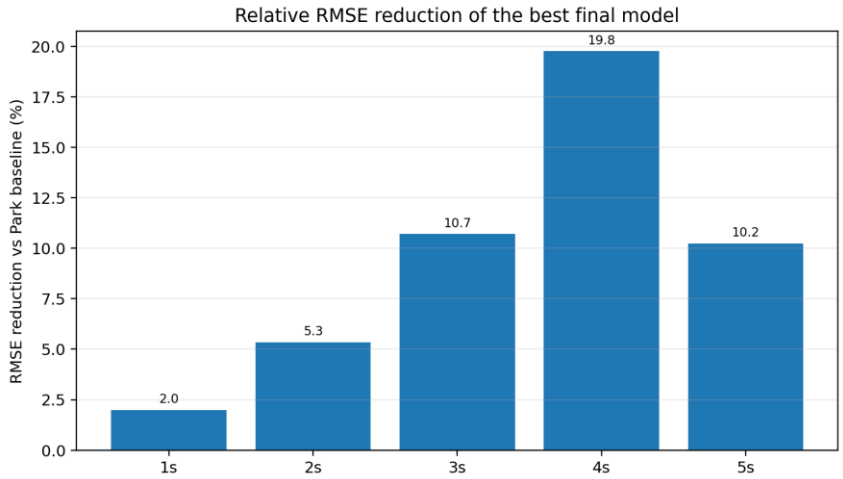

Fig. 12 Relative RMSE reduction of the best final model versus the Park et al. baseline

As summarized in Table 8 and illustrated in Fig. 12, for M6 (ResNeXt-Xception-Attention-LSTM), the RMSE reduction relative to the selected Park et al. published baseline is positive at all five reported horizons: approximately 2.0%, 5.3%, 10.7%, 19.8%, and 10.2% from 1 to 5 s. The largest contextual relative difference occurs at the 4 s horizon. These percentages apply specifically to M6 and remain subject to the cross-study protocol limitations stated above.

4.11 Comparison with Chen et al.

Table 9 Accuracy comparison with models reported by Chen et al.

Model	Speed (%)	Steering (%)
Chen et al. ResNet-152	78.3	84.2
Chen et al. Inception-V4	77.3	84.8
Proposed M5	93.51	94.75
Proposed M6	94.09	94.91

As summarized in Table 9, the dissertation comparison reports higher tolerance accuracy for the two attention-based final models than for the cited Chen et al. ResNet-152 and Inception-V4 baselines. These values are contextual rather than strictly controlled head-to-head comparisons: dataset partitioning, preprocessing, tolerance thresholds (including the present study's ±6° steering and ±5 km/h speed criteria), and evaluation procedures were not necessarily identical across studies. More recent multimodal systems such as UniAD, TransFuser, BEVFusion, EMMA, and the 2025-2026 studies discussed in the future-work section are therefore used primarily as contemporary methodological context rather than inserted into Tables 7-9 as numerically equivalent baselines.

4.12 Optimization interpretation

The complete result sequence supports a layered interpretation of optimization. In Phase I, changing sensory representation while holding the optimizer conceptually fixed changes the

attainable solution. Adding temporal state produces further gains. Phase II then changes the architecture itself through stacking and attention. The observed performance can therefore be summarized conceptually as $P = \Phi(O,A,X,T,F)$, where P denotes observed performance, O the optimizer and loss-setting, A the network architecture, X the input representation, T the temporal-modeling choice, and F the fusion strategy. This is a descriptive relation, not a fitted statistical model.

Future experiments should save per-task gradients and evaluate the cosine criterion in Equation (14), enabling direct comparison of standard weighted Adam with PCGrad, CAGrad, FAMO, or related methods [19,20,22].

4.13 Sensitivity and reproducibility

The present analysis is configuration-level rather than a full hyperparameter sweep. A stronger reproducibility protocol should report random seeds, repeated runs, confidence intervals, exact normalization constants, optimizer β-values, per-output loss scales, training time, and hardware configuration. These additions would separate architecture effects from stochastic training variance.

Tolerance-based accuracy is intuitive but can hide changes within the accepted interval. For that reason, the article reports RMSE alongside accuracy and recommends future inclusion of MAE, R^2, calibration measures, and closed-loop driving metrics when the necessary predictions or simulator evaluations are available. The consolidated empirical findings are summarized in Table 10.

4.14 Consolidated findings

Table 10 Consolidated empirical findings

Theme	Reference result	Value
Temporal context	RGB-only LSTM gain	Steering: +9.2 pp (Xception), +10.5 pp (ResNeXt)
Best Phase I	ResNeXt C2	81.6% speed; 84.3% steering
Best Phase II	M6	94.09% speed; 94.91% steering
Balanced performance	M6 BAI	94.50%
Mean 1–5 s RMSE	M6	1.0734
Largest RMSE reduction vs Park	4 s horizon	19.8%

5 Extended discussion

5.1 Why configuration matters to optimization

Optimization algorithms operate on the loss surface induced by a particular model and data representation. Changing the backbone changes parameterization and feature geometry; changing LiDAR preprocessing changes the statistics of the input; adding LSTM changes both the dimensionality of the trainable state and the information available to the loss; attention

changes how feature relevance is distributed. Therefore, identical Adam hyperparameters can lead to different effective optimization problems even when the task labels are unchanged.

The Phase-I results provide a concrete example. PointNet improves Xception without LSTM, while PCM becomes particularly effective in the C2 recurrent configuration. This interaction argues against treating preprocessing as a fixed front-end decision. In multimodal E2E learning, preprocessing and optimization are coupled through representation.

5.2 Scalarization versus true multi-objective optimization

A weighted sum is the simplest mechanism for combining steering and speed losses. Its main advantages are implementation simplicity and compatibility with standard optimizers. However, fixed scalarization can be sensitive to scale and may miss non-convex portions of a Pareto frontier. Sener and Koltun [18] therefore formulate multi-task learning explicitly as multi-objective optimization, while later work manipulates or balances gradients [19,20,22].

For the present two-output control problem, a practical next experiment would compare fixed scalarization, normalized scalarization, and adaptive or gradient-aware balancing under identical data splits. Evaluation should remain two-dimensional so that an improvement in one output at the expense of the other is not hidden by a single aggregate number.

5.3 Implications for multimodal robustness

Recent multimodal research emphasizes that fusion performance should be evaluated under sensor degradation, not only under clean inputs. BEVFusion explicitly studies LiDAR malfunction [27], and current E2E surveys identify robustness as a central unresolved challenge [26]. This is directly relevant because Phase I already demonstrates sensitivity to LiDAR representation.

A natural continuation is a controlled robustness matrix in which camera noise, LiDAR sparsification, modality dropout, synchronization error, and weather-related image degradation are introduced independently. The same steering–speed metrics can then quantify whether degradation affects lateral and longitudinal control symmetrically or produces task-specific failure modes.

5.4 Practical meaning of the phase-II progression

The Phase-II progression is important because it shows that the final result is not an isolated high-accuracy number. The six models form an ordered sequence of design decisions: stacking, temporal modeling, and attention. Averaging the two backbone orderings at each stage reduces sensitivity to one particular ordering and supports the interpretation that the final architecture benefits from multiple complementary mechanisms.

6 Conclusion and future research

This paper presented an expanded mathematical and experimental analysis of multimodal E2E steering and speed prediction derived from the underlying doctoral research. The control

problem was expressed as coupled multi-output regression, with explicit steering and speed objectives, Adam-based updates, Pareto interpretation, and a separation between trainable parameters and higher-level hyperparameters.

Phase I demonstrated substantial sensitivity to input representation and temporal modeling. Phase II then showed a clear progression from stacked backbones to LSTM-augmented and attention-augmented models. The best final model, ResNeXt–Xception–Attention–LSTM, achieved 94.91% steering accuracy and 94.09% speed accuracy and the strongest Balanced Accuracy Index.

Horizon-wise RMSE analysis further supported the M6 configuration. Across 1–5 s horizons, M6 produced RMSE values of 0.6116, 0.8626, 1.0866, 1.3005, and 1.5055, respectively, with positive contextual reductions relative to the selected Park et al. published baseline at every reported horizon. These differences should not be interpreted as a controlled head-to-head benchmark, because the compared results were obtained under non-identical experimental protocols.

Future work should move from post-hoc multi-objective interpretation to directly tested adaptive optimization. Priority directions are per-task gradient logging and conflict analysis; comparison of fixed weighting with uncertainty weighting, PCGrad, CAGrad, and FAMO; outer-loop hyperparameter search around Adam; repeated-run statistical analysis; and robustness experiments under camera/LiDAR degradation and partial modality loss. Recent studies further motivate this direction: Guo et al. [33] combine multimodal spatial-temporal fusion for multi-task steering and speed prediction, Yu et al. [34] investigate bilateral modality interaction for multimodal end-to-end driving, and Zhao et al. [35] employ transformer-based multimodal fusion with adaptive trajectory-control balancing. These developments provide relevant directions for extending the present configuration-sensitivity framework while maintaining protocol-consistent evaluation.

Acknowledgments

The authors would like to express their sincere gratitude to the late Dr. Ali Jamali, whose valuable guidance, scientific insight, and continued support played an important role in the development and advancement of this research. His contributions to the conceptual and methodological foundations of this work are gratefully acknowledged. His invaluable contributions to this research are deeply appreciated, and his memory will remain an enduring part of this work.

References

1. Kuutti, S., Bowden, R., Jin, Y., Barber, P., & Fallah, S. (2021). A survey of deep learning applications to autonomous vehicle control. IEEE Transactions on Intelligent Transportation Systems, 22(2), 712–733, doi:10.1109/TITS.2019.2962338.
2. Kocić, J., Jovičić, N., & Drndarević, V. (2019). An end-to-end deep neural network for autonomous driving designed for embedded automotive platforms. Sensors, 19(9), 2064. doi:10.3390/s19092064.
3. Chen, C., Seff, A., Kornhauser, A., & Xiao, J. (2015). DeepDriving: Learning affordance for direct perception in autonomous driving. Proceedings of ICCV, 2722–2730. doi:10.1109/ICCV.2015.312.
4. Xu, H., Gao, Y., Yu, F., & Darrell, T. End-to-end learning of driving models from large-scale video datasets. Proceedings of CVPR, 2174–2182, 2017.
5. Bojarski, M., et al. End to end learning for self-driving cars. arXiv:1604.07316, 2016.

6. Eraqi, H. M., Moustafa, M. N., & Honer, J. End-to-end deep learning for steering autonomous vehicles considering temporal dependencies. arXiv:1710.03804, 2017.
7. Codevilla, F., Müller, M., López, A., Koltun, V., & Dosovitskiy, A. (2018). End-to-end driving via conditional imitation learning. Proceedings of ICRA, 4693–4700, doi:10.1109/ICRA.2018.8460487.
8. Ishihara, K., Kanervisto, A., Miura, J., & Hautamäki, V. (2021). Multi-task learning with attention for end-to-end autonomous driving. Proceedings of CVPR Workshops, 2902–2911.
9. Xiao, Y., Codevilla, F., Gurram, A., Urfalioglu, O., & López, A. M. (2022). Multimodal end-to-end autonomous driving. IEEE Transactions on Intelligent Transportation Systems, 23(1), 537–547, doi:10.1109/TITS.2020.3013234.
10. Chen, Y., Wang, J., Li, J., Lu, C., Luo, Z., Xue, H., & Wang, C. (2018). LiDAR-Video driving dataset: Learning driving policies effectively. Proceedings of CVPR, 5870–5878.
11. Park, M., Kim, S., & Park, S. (2021). A convolutional neural network-based end-to-end self-driving using LiDAR and camera fusion: Analysis perspectives in a real-world environment. Electronics, 10(21), 2608. doi:10.3390/electronics10212608.
12. Prakash, A., Chitta, K., & Geiger, A. (2021). Multi-modal fusion transformer for end-to-end autonomous driving. Proceedings of CVPR, 7077–7087.
13. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735.
14. Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR), arXiv:1412.6980.
15. Kendall, A., Gal, Y., & Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. Proceedings of CVPR, 7482–7491.
16. Groenendijk, R., Karaoglu, S., Gevers, T., & Mensink, T. (2021). Multi-loss weighting with coefficient of variations. Proceedings of WACV, 1469–1478, 2021. doi:10.1109/WACV48630.2021.00151.
17. Chennupati, S., Sistu, G., Yogamani, S., & Rawashdeh, S. (2019). MultiNet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. Proceedings of CVPR Workshops.
18. Sener, O., & Koltun, V. (2018). Multi-task learning as multi-objective optimization. Advances in Neural Information Processing Systems 31 (NeurIPS).
19. Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., & Finn, C. (2020). Gradient surgery for multi-task learning. Advances in Neural Information Processing Systems 33 (NeurIPS).
20. Liu, B., Liu, X., Jin, X., Stone, P., & Liu, Q. (2021). Conflict-averse gradient descent for multi-task learning. Advances in Neural Information Processing Systems 34 (NeurIPS).
21. Xin, D., Ghorbani, B., Gilmer, J., Garg, A., & Firat, O. (2022). Do current multi-task optimization methods in deep learning even help? Advances in Neural Information Processing Systems 35 (NeurIPS).
22. Liu, B., Feng, Y., Stone, P., & Liu, Q. (2023). FAMO: Fast adaptive multitask optimization. Advances in Neural Information Processing Systems 36 (NeurIPS), doi:10.52202/075280-2500.
23. Zhao, J., Zhao, W., Deng, B., et al. (2024). Autonomous driving system: A comprehensive survey. Expert Systems with Applications, 242, 122836, doi:10.1016/j.eswa.2023.122836.
24. Hwang, J.-J., Xu, R., Lin, H., et al. (2024). EMMA: End-to-End Multimodal Model for Autonomous Driving. arXiv:2410.23262.
25. Hu, Y., Yang, J., Chen, L., et al. (2023). Planning-oriented autonomous driving. Proceedings of CVPR, 17853–17862, 2023.
26. Chen, L., Wu, P., Chitta, K., Jaeger, B., Geiger, A., & Li, H. (2023). End-to-end autonomous driving: Challenges and frontiers. arXiv:2306.16927.
27. Liang, T., Xie, H., Yu, K., et al. (2022). BEVFusion: A simple and robust LiDAR-camera fusion framework. Advances in Neural Information Processing Systems 35 (NeurIPS). doi:10.52202/068431-0757.
28. Xu, D., Li, H., Wang, Q., Song, Z., Chen, L., & Deng, H. (2024). M2DA: Multi-modal fusion transformer incorporating driver attention for autonomous driving. arXiv:2403.12552.
29. Aghamohammadi, M. (2026). Optimal Training of Driver Behavior Based on Deep Learning for Autonomous Vehicles. Doctoral dissertation, Islamic Azad University.
30. Navarro, P. J., Miller, L., Rosique, F., Fernández-Isla, C., & Gila-Navarro, A. (2021). End-to-End Deep Neural Network Architectures for Speed and Steering Wheel Angle Prediction in Autonomous Driving. Electronics, 10(11), 1266, doi:10.3390/electronics10111266.
31. Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. Proceedings of CVPR, 1251–1258.
32. Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. Proceedings of CVPR, 1492–1500.

33. Guo, B., Liu, H., Yang, X., Cao, Y., Jin, L., & Wang, Y. Multi-modal information fusion for multi-task end-to-end behavior prediction in autonomous driving. Neurocomputing, 634, 129857, 2025. doi:10.1016/j.neucom.2025.129857.
34. Yu, Z., Li, J., Chen, Z., Wei, Y., Zhang, X., & Tan, X. Multimodal end-to-end autonomous driving via bilateral modality interaction. Expert Systems with Applications, 2025, 128458. doi:10.1016/j.eswa.2025.128458.
35. Zhao, X., Ke, H., & Yang, Y. Transformer-based multi-modal feature fusion for end-to-end autonomous driving. International Journal of Transportation Science and Technology, 2026. doi:10.1016/j.ijtst.2025.10.017.