Model Fusion: Weighted N-Version Programming for Resilient Autonomous Vehicle Steering Control

Ailec Wu, Abu Hasnat Mohammad Rubaiyat, Chris Anton, Homa Alemzadeh
Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904-4259
{aw5fa, ar3fx, cca4qg, ha4d}@virginia.edu

***Abstract*—We present the preliminary results on developing a weighted N-version programming (NVP) scheme for ensuring resilience of machine learning based steering control algorithms. The proposed scheme is designed based on the fusion of outputs from three redundant Deep Neural Network (DNN) models, independently designed using Udacity's self driving car challenge data. The improvement in reliability compared to single DNN models is evaluated by measuring the steering angle prediction accuracy in the presence of simulated perturbations on input image data caused by various environmental conditions.**

***Keywords*—machine learning, N-version programming, resilience, steering angle prediction, image perturbation**

I. Introduction

At the core of autonomous vehicle (AV) technologies are machine learning (ML) models that enable path planning and control based on perception of the surrounding environment. Lane Keep Assistance System (LKAS) is a key AV component utilizing computer vision that processes road images, locates lane markers, and adjusts steering angle to keep the vehicle inside the lanes. Many recent studies demonstrate the vulnerability of Deep Neural Network (DNN) models to adversarial or accidental perturbations [1] [2] [3] that might corrupt AV camera input. However, to the best of our knowledge, no prior work has studied solutions for improving the resilience of AVs to such perturbations.

This paper presents preliminary results on using redundancy to improve the resiliency of DNN-based steering angle prediction algorithms in AVs. We propose an N-version programming (NVP) inspired approach based on fusion of outputs from different DNN models independently designed with different numbers of layers and parameters. In particular, we combine three independently developed community models from Udacity's self driving car project using a weighted voting scheme to improve the performance of the individual DNNs. The system resilience is assessed by injecting increasing levels of image perturbations to the DNN inputs, measured by the structural similarity index (SSIM) [4]. Our proposed NVP scheme addresses asymmetry in the reliability of different community models as well as their continuous outputs (steering angles).

The preliminary results show two major improvements. First, weighted model fusion on average achieves 40% higher accuracy than the individual algorithms in predicting steering angles, as measured by the root mean square error (RMSE) averaged across all different algorithms and types of perturbation. Second, the voter provides robustness to counteract the failure of a single DNN model. The improved accuracy and reliability of the combined system indicates a promising venue to augment the current "sensor fusion" practice in AV technology with "model fusion".

II. Systems Overview and Methodology

We used three independently designed DNN-based steering angle prediction algorithms (Chauffeur, Autumn, and Rambo), from Udacity's challenge [5] in this work. Fig. 1 illustrates the high-level design of our proposed model fusion algorithm that combines these three networks. These DNN models varied in implementation: The Chauffeur model includes one convolutional neural network (CNN) model for extracting features from the image, and one long short-term memory (LSTM)/recurrent neural network (RNN) model for predicting steering angle; the Autumn model consists of 5 CNNs and an LSTM/RNN layer; the Rambo model consists of three CNNs whose outputs are merged using a final layer.

For evaluating the proposed system, we perturb the images by adding real-world environmental effects, feed the faulty images to the networks, record predicted steering angles, apply the weighted voting approach, and measure the final steering angle. Each model (and our combined system) was evaluated based on RMSE steering angle performance compared to a ground truth recording, as shown below:

$$\text{RMSE} := \sqrt{\frac{\sum_{i=1}^{n}(\theta_{\text{i pred}} - \theta_{\text{i ground truth}})^2}{n}} \tag{1}$$

A. Image Perturbation and Quality Assessment

With the images synthesized from one of Udacity's testing sets, we simulated real-world environmental scenarios affecting image quality (rain, fog, snow, contrast change, and brightness change) by creating image perturbations using the technique presented in [6]. The perturbation levels were increased until the image quality level measured by SSIM reached to 0.6. Figs. 2 and 3 show examples of simulated weather conditions and the performance of the individual algorithms.

B. Weighted N-version Programming

Adaptation of the N-version programming approach to DNN models raises two major questions. First, how do we appropriately weigh each neural network's reliability? To account for different model performances, the voting process weighs each model's output inversely proportionally to its RMSE calculated

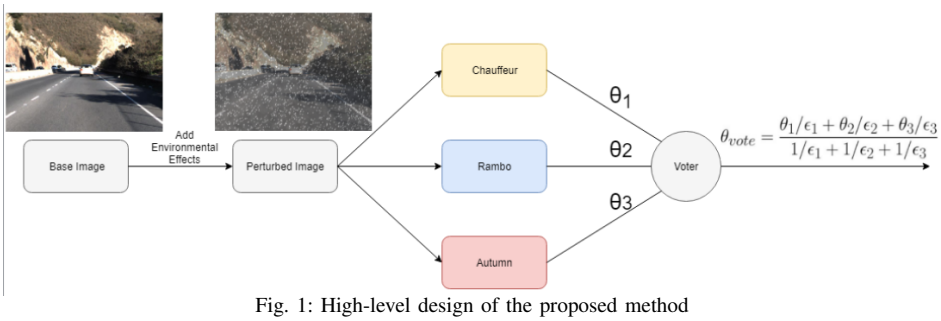


Fig. 1: High-level design of the proposed method

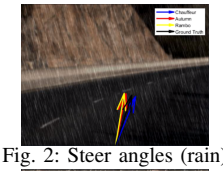


Fig. 2: Steer angles (rain)

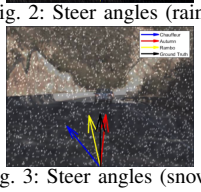
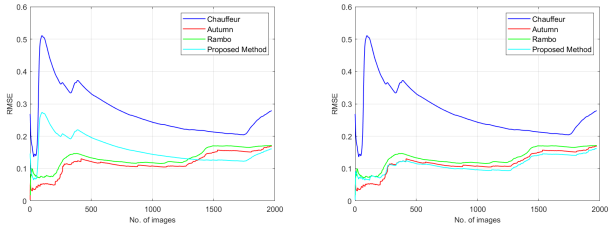
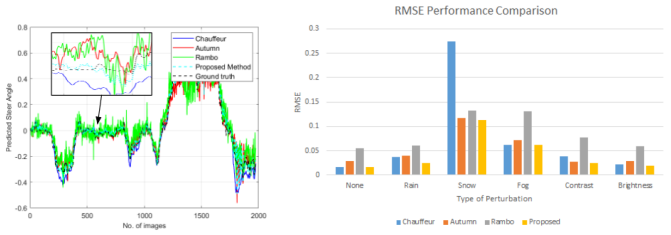


Fig. 3: Steer angles (snow)



(a) (b)

Fig. 4: Effects of NVP on the faulty images (Simulated snow scenario), (a) RMSE (weighted voting), (b) RMSE (NVP+weighted voting)



(a) (b)

Fig. 5: (a) Comparison of the proposed method with three networks in case of rainy images, (b) Overall RMSE of the three networks and proposed method

from prior training data (this simple scheme is referred to as weighted voting in Figure 4a):

$$\theta_{vote} = \frac{\theta_1/\epsilon_1 + \theta_2/\epsilon_2 + \theta_3/\epsilon_3}{1/\epsilon_1 + 1/\epsilon_2 + 1/\epsilon_3} \tag{2}$$

Second, how will the voter identify agreement and detect faults? To deal with the steering angle outputs from the DNN models, we use each model's maximum steering angle deviation from the ground truth, calculated from prior training data, as a maximum deviation threshold for voter acceptance.

In the NVP+weighted voting scheme, the steering angle differences between each pair of DNN models is calculated, and compared against the maximum deviation thresholds. If a particular model is faulty, it will affect the differences between the two corresponding steering angles as shown below.

$$\Delta\theta_{12} \approx \Delta\theta_{13} \not\approx \Delta\theta_{23} \implies \theta_1 \not\approx \theta_2 \approx \theta_3 \tag{3}$$

In the case of this single error detection, the voter can effectively correct faults by excluding the value in its voting algorithm (Eq. 2). Furthermore, if two steering angle predictors are faulty, then all three steering angle differences will not be equal within the deviation thresholds and, thus, a double error is identified:

$$\Delta\theta_{12} \not\approx \Delta\theta_{13} \not\approx \Delta\theta_{23} \implies \text{double error} \tag{4}$$

III. Preliminary Results

Fig. 4 (a) and (b) represent, respectively, the RMSE of the weighted model fusion without (just Eq. 2) and with Eq. 3's outlier rejection compared to baseline algorithms (bottom legend line). The NVP plus weighted voting performed substantially better in this scenario than using just Eq. 2.

Fig. 5a compares the predicted steering angles and RMSEs of the three networks versus the proposed method when running on rainy images. Fig. 5b illustrates the overall RMSE measured for the perturbed images and original images. In case of perturbed images our proposed method shows substantially improved RMSE when compared to the three baseline networks. Notably, while the Chaffuer model struggled with the snowy images, the Rambo model struggled with the foggy images. On the other hand, the Autumn model performed well consistently across different scenarios. However, our three version model fusion scheme consistently discarded faulty models and achieved better performance than the Autumn model. On average our model achieves 40% lower RMSE score compared to the average RMSE of different models across different perturbation scenarios. These results indicate the potential of the proposed NVP-inspired model fusion technique in improving the resilience of DNN-based steering control in AVs.

However, the cost of developing an NVP model is potentially high, requiring training individual DNN models with different structures and parameters. The degree of independence of the individual DNN models and their failure rates is affected by the diversity of datasets used for training. A more in-depth analysis of the cost associated with the proposed NVP model and the improvement in reliability compared to individual models, given different DNN structures, training datasets, and failure rates are the subject of future work.

References

[1] Y. Tian *et al.*, "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," *arXiv preprint arXiv:1708.08559*, 2017.

[2] K. Pei *et al.*, "Deepxplore: Automated whitebox testing of deep learning systems," in *Proceedings of the 26th Symposium on Operating Systems Principles*. ACM, 2017, pp. 1–18.

[3] S. Jha *et al.*, "Avfi: Fault injection for autonomous vehicles," in *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE, 2018.

[4] K. G. Larkin, "Structural similarity index ssimplified: Is there really a simpler concept at the heart of image quality measurement?" *arXiv preprint arXiv:1503.06680*, 2015.

[5] Udacity, "Self driving car," https://github.com/udacity/self-driving-car/tree/master/steering-models/community-models, 2018.

[6] A. H. M. Rubaiyat *et al.*, "Experimental Resilience Assessment of An Open-Source Driving Agent," *ArXiv e-prints*, Jul. 2018.