Applying Knowledge Distillation for Model Size Reduction in Lane Detection Models for Autonomous Driving Vehicles

LeAnn Mendoza
Northeastern University, San Jose
San Jose, CA, USA
mendoza.l@northeastern.edu

ABSTRACT

Deep learning (DL) is a prominent and growing area of machine learning that is rapidly changing how complex data in the world is modeled, interpreted, and utilized. To model these complexities, building accurate and robust DL models can often require heavy computation, memory, and energy costs. Because of this, research efforts in constructing DL neural networks with small model size, light computation cost and high segmentation accuracy have attracted much attention. In this research, we will design efficient and compact DL networks using knowledge distillation[1], a model compression technique in which a large cumbersome model is *distilled* to a smaller student model while preserving accuracy[2], to be applied to lane detection in autonomous driving vehicles. We will do this by developing a student CNN model that reduces model size and minimize accuracy loss as measured by difference in MSE and R^2 between cumbersome model and student models. Our research suggests that knowledge distillation can be applied to deep learning models trained for lane detection with improved model compactness and moderate accuracy preservation.

1 PROBLEM AND MOTIVATION

Deep Learning (DL), an area of machine learning interested in modeling data through neural networks[3], has been applied to computer vision and image classification across many fields. However, the application of DL has proven to require heavy computation, memory and energy demands hindering DLs application into low-power edge devices [4]. Without high computing power afforded by either costly cloud-powered systems or high-performance computing (HPC) clusters, a DL models pipeline suffers from size constraints, latency, and lack of runtime memory [2]. DL neural networks with small model size, light computation cost and high segmentation accuracy, have attracted much attention because of the growing need of applications on edge devices[5].

In this research, we design efficient and compact DL networks using knowledge distillation[1], a model compression technique in which a large cumbersome model is *distilled* to a smaller student model without a considerable performance loss[2], to be applied to lane detection in autonomous driving vehicles.

2 BACKGROUND AND RELATED WORK

2.1 Lane Detection Approaches

Current approaches to lane detection methods in intelligent transport systems can be classified into two categories; feature-based or model-based [6]. Hough lane detection is one of the more popular lane detection feature-based approaches that utilizes an algorithm that automatically emphasizes the lane marks and recognizes them from digital images[7]. Model based approaches to lane detection often involve the deployment of convolutional neural networks (CNNs), a type of DL network where the network "learns" features to maximize its ability to correctly classify images through training on data with known labels[8]. In this research, we will utilize both approaches, feature-based and model-based, coupled with knowledge distillation (described below) to develop more efficient CNNs for lane detection.

2.2 Knowledge Distillation

Proposed in 2015 by Hinton et. Al, knowledge distillation is the process of transferring ("distilling") the knowledge a cumbersome model, describing a trained ensemble of models or a single large model, into a smaller model more suitable for deployment[1]. Knowledge distillation utilizes the "softmax" output layer of class probabilities produced by the cumbersome model to train the distilled model in producing the correct labels, as described below:

> Neural networks typically produce class probabilities by using a "softmax" output layer that converts the logit, z_i, computed for each class into a probability, q_i, by comparing z_i with the other logits. where T is a temperature that is normally set to 1. Using a higher value for T produces a softer probability distribution over classes [1].

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)} \quad (1)$$

By tweaking the temperature (T) of the distillation of class probabilities produced by larger trained lane detection models, we will develop smaller student models that reduce model size and minimize accuracy loss as measured by difference in mean squared error (MSE) and r-squared coefficient (R^2) between cumbersome model and student models.

3 APPROACH AND UNIQUENESS

3.1 Data Collection

To build the autonomous driving vehicle used to capture the image data, we built a SunFounder Smart Video Car Kit V2.0 for Raspberry Pi 4. Following the software set up instructions outlined in Atol Lab: Autonomous Car using Deep Learning and Computer Vision (PiCar Ver.)[9], we collected images and saved their respective driving angle, calculated using Hough line detection, in a driver-controlled run.

3.2 Cumbersome Model Development and Training

The cumbersome model was built with TensorFlow[10] Keras[11] libraries and used an NVIDIA CNN Model architecture, a sequential model that utilizes an Exponential Linear Unit (ELU) activation function, optimized with Adam, with a total of 252, 219 parameters. Data augmentation strategies were applied to increase variation in the training set. 567 images collected from the driver-controlled run were used to create the training (n = 432 images), validation (n = 49 images), and testing datasets (n = 86 images). Images were augmented by resizing image to 200 x 66 pixels, normalized, and applying randomized zooming, panning, brightness adjustment, blurring, and/or random flipping (Figure 1).

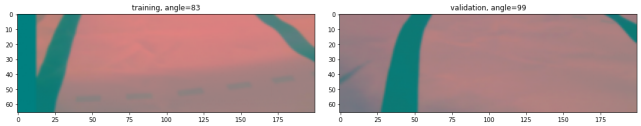

Figure 1. Images are collected from driver-controlled run of autonomous vehicle, and steering angles are calculated using Hough line detection. Images are then randomly augmented to create the training data sets.

3.3 Experimental Design

In these experiments, we seek to develop a student CNN model that reduces model size and minimize accuracy loss as measured by difference in MSE and R^2 between cumbersome model and student models.

We defined 10 student model architectures. All student models are sequential models that utilize an Exponential Linear Unit (ELU) activation function, optimized with Adam, but differ in filter size between their top two 2D Convolution layer filter size which alters the model's number of parameters (See Table 1).

Model Number	Model Size (# parameters)	Conv2D Layer 1 Filter Size	Conv2D Layer 1 Filter Size
0	70345	12	24
1	109861	24	36
2	151969	36	48
3	209537	48	64
4	105061	12	36
5	139777	12	48
6	186065	12	64
7	145873	24	48
8	193889	24	64
9	201713	36	64

Table 1. Ten sequential student models with varying filter size withing the top two Convolution layers, effecting parameter number and model size.

Let a trial be defined as applying knowledge distillation from cumbersome model to student model using temperature (T). For each student model architecture, we ran 10 trials for each T where T ∈ [12 2.5]). To evaluate accuracy of the model the difference in MSE and R^2 cumbersome model and student models was collected, and the averaged per trial for analysis.

4 RESULTS AND CONTRIBUTIONS

Results from experiments are outlined in Table 2 and visualized in Figure 2. The smallest MSE difference was observed in model 6 and temperature 5 at 10.15 degrees (±3.19 degrees error), with an R^2 difference of 0.14, and a 26.28% model size reduction. Model size was reduced the most at 72.11% in Model 0. Model 0 at temperature 10 produced its smallest MSE difference occurred with 18.31 degrees (±4.28 degrees error), with an R^2 difference of 0.25. The overall worst performing student model was Model 0 distilled at temperature 2.5, with an MSE of 26.97 (±5.19 degrees error), degrees and R^2 difference of 0.37.

		Temperature							
		20		10		5		2.5	
Model Number	Model Size (% parameter reduction)	Difference in MSE (degrees)	Difference in R^2	Difference in MSE (degrees)	Difference in R^2	Difference in MSE (degrees)	Difference in R^2	Difference in MSE (degrees)	Difference in R^2
0	72.11	23.50	0.32	18.31	0.25	23.39	0.32	26.97	0.37
4	58.35	13.84	0.13	11.73	0.11	11.89	0.11	13.08	0.12
1	56.44	12.96	0.15	11.43	0.13	12.77	0.15	14.71	0.17
5	44.58	14.21	0.19	17.63	0.24	15.92	0.22	16.03	0.22
7	42.16	15.75	0.24	21.75	0.33	22.58	0.35	14.93	0.23
2	39.74	15.70	0.21	17.53	0.24	19.47	0.26	18.17	0.25
6	26.28	12.13	0.17	13.19	0.18	10.15	0.14	13.34	0.15
8	23.13	17.09	0.16	14.72	0.14	14.98	0.14	11.89	0.11
9	20.02	14.09	0.17	15.00	0.17	18.23	0.21	14.90	0.17
3	16.92	10.25	0.14	11.41	0.16	12.33	0.17	12.19	0.17

Table 2: Table of resulting data mapping model size ordered by % parameter reduction between the cumbersome model and student model, difference in MSE in degrees between the cumbersome model and student model, and difference R^2 between the cumbersome model and student model.

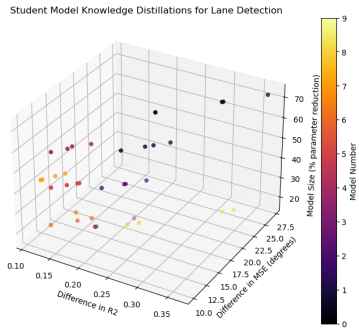

Figure 2: 3D Scatterplot of resulting data mapping the difference in R^2 and difference in MSE (degrees) from the cumbersome model and student model, and the student model size in % parameter reduction. Color of datapoints, as represented by the color bar, correlates to the model number (x ∈ {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}).

Our research suggests that knowledge distillation can be applied to deep learning models trained for lane detection with improved model compactness and moderate accuracy preservation.

REFERENCES

[1] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531,* vol. 2, no. 7, 2015.

[2] A. Banitalebi-Dehkordi, "Knowledge distillation for low-power object detection: A simple technique and its extensions for training compact models using unlabeled data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 769-778.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," vol. 521, ed: *Nature*, 2015, pp. 436–444.

[4] N. D. Lane *et al.*, "Deepx: A software accelerator for low-power deep learning inference on mobile devices," in *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2016: IEEE, pp. 1-12.

[5] Y. Liu, K. Chen, C. Liu, Z. Qin, Z. Luo, and J. Wang, "Structured knowledge distillation for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2604-2613.

[6] G. Kaur and D. Kumar, "Lane detection techniques: A review," *International Journal of Computer Applications,* vol. 112, no. 10, 2015.

[7] F. Mariut, C. Fosalau, and D. Petrisor, "Lane mark detection using Hough transform," *2012 International Conference and Exposition on Electrical and Power Engineering,* pp. 871-875, 2012.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 770-778, 2016.

[9] J. Lee. "AtoI Lab : Autonomous Car using Deep Learning and Computer Vision (PiCar Ver.)." https://github.com/jaykay0408/dmcar-student (accessed.

[10] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467,* 2016.

[11] F. a. o. Chollet, "Keras," ed, 2015.

[12] T. Kobayashi, T. Shiba, A. Kinoshita, T. Matsumoto, and Y. Hori, "The influences of gender and aging on optic nerve head microcirculation in healthy adults," *Scientific Reports,* vol. 9, no. 1, p. 15636, 2019/10/30 2019, doi: 10.1038/s41598-019-52145-1.