Exploring transfer learning for self-driving car dataset

Paula Moraes and Felipe Salvatore

Department of Computer Science - University of São Paulo, Brazil

{paulaksm,felsal}@ime.usp.br

IME - Instituto de Matemática e Estatística

1. Introduction

A core subject in the self-driving cars domain is how to use images taken in real-time to best steer a vehicle in a road. In this context, how to encode an image as a feature vector is a crucial step. To obtain a good performance with an end-to-end learning approach, it is mandatory to have a large dataset of this domain. **However, domain-related datasets are not always available in a good amount nor easily collected. Learning from small datasets can be achieved with transfer learning.**

Transfer learning is a framework that allows knowledge transfer in a way that the training domain and feature space for a given task in a model may be different from the testing domain or task for this same model. One approach to transfer learning is to take pretrained CNNs and remove the last fully-connected layer, working as a **fixed feature extractor**. The resulting image embedding can be used as input of different machine learning algorithms.

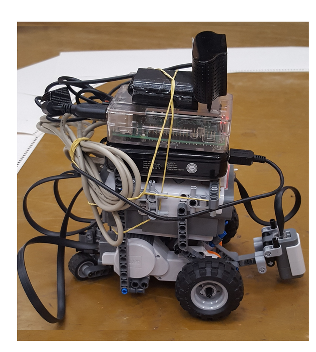
Research Questions:

- *How powerful can pretrained CNNs be in a complete new domain of which they were trained?*
- *Is it possible to achieve similar accuracy of end-to-end models using transfer learning to generate image embeddings?*

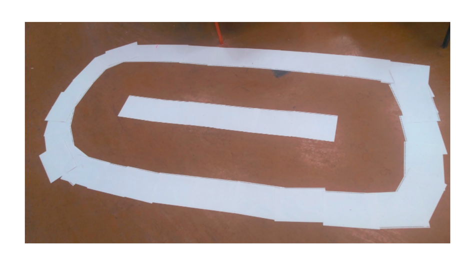
2. Learning Task and Dataset

Self-driving car dataset:

- Raspberry Pi 3 with a camera mounted on a Lego Mindstorms NXT robot car
- images recorded while taking laps at an oval track
- 70,000 data points
- 3 categories ('left', 'forward', 'right')
- $45 \times 80 \times 3$



RC car



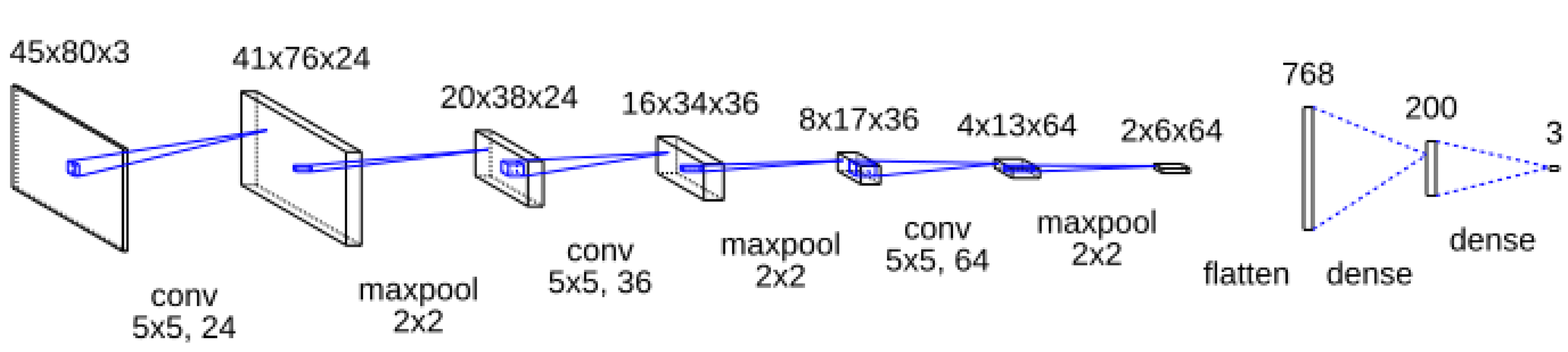
Track

left

forward

right

Task: ***Road Following*** - Given an image, in real-time, send a discrete control command to the vehicle.



Baseline - Proposed CNN architecture

3. VGG-16

- Input size 224×224 RGB images
- 16 standard convolutional layers
- 3×3 filters

4. Xception

- Input size 299×299 RGB images
- Depthwise separable convolutions instead of standard convolutions
- 36 convolutional layers
- Smaller memory footprint

5. MobileNet

- Input size 224×224 RGB images
- Built on top of Xception network
- Possible to shrink model through width and resolution multipliers hyperparameters

Contact & References

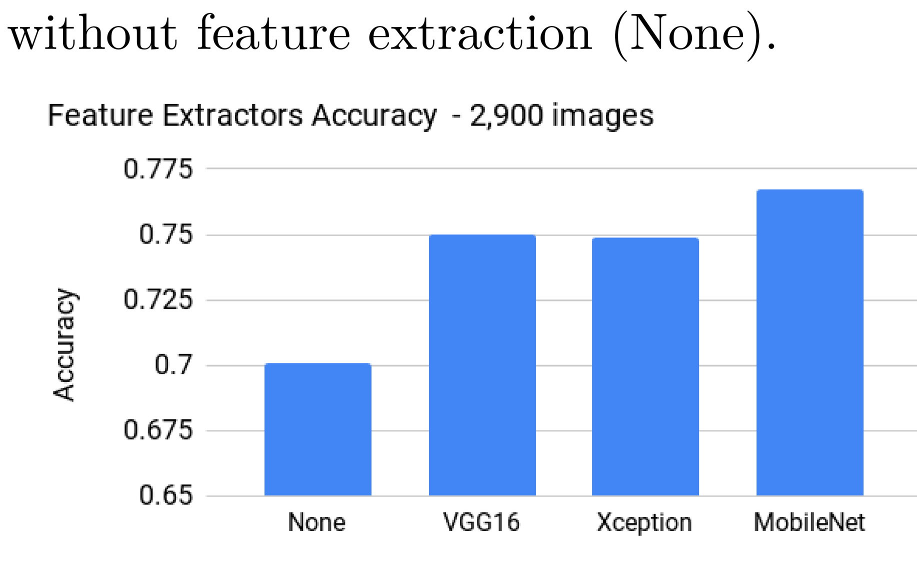
6. Results

Results from training simple multilayer perceptrons on different image embbedings compared to models trained in an end-to-end fashion (highlighted in red).

Embedding	Architecture	Input features	Training instances	Accuracy
VGG-16	[200, 3]	4,096	2,900	0.797
MobileNet	[200, 3]	1,024	2,900	0.784
Xception	[200, 3]	2,048	2,900	0.785
Flattening	[200, 3]	10,800	56,000	0.773
Raw image	[(24, 5), (36, 5), (64, 5), 200, 3]	768	56,000	0.800

7. Results II

Accuracy evaluation of feature extraction methods and without feature extraction (None).



8. Discussion & Future Work

Conclusion

- Empirically, transfer learning yields good results on feature extraction in a unique domain.
- It was possible to achieve similar results of deep models trained in a classic fashion using only 5% of the training dataset in a transfer learning framework.

Future Work

- Implement layer visualization techniques on the pretrained models to gain better insights on their hidden layers.
- Explore different transfer learning approaches based in CNNs to self-driving car tasks.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.