Enhancing the Performance of the Neural-Network for the Self-Driving Car

Alexander Hadik (Alexander_Hadik@brown.edu)
Undergraduate, Brown University

Sukhyun Kevin Jung (Kevin_Jung@brown.edu)
Undergraduate, Brown University

Evan Lester (Evan_Lester@brown.edu)
Undergraduate, Brown University

Keywords: Neural Network; Hidden Layers; Retinal Coverage;

Abstract

To improve the performance of the neural network, a variety of image preprocessing methods were implemented. By reducing the dimensionality of the image by removing unnecessary portions, it was found to improve the performance of a neural-network enabled RC car navigating a course.

Introduction

Mathematical modeling techniques of human cognition offer many representations of human perception and decision making. These techniques have value in modeling human perception of the environment. Additionally, these techniques have the potential to simulate real-time decision making in real world applications. To test the viability of neural networks in real-time applications, we implemented a system capable of controlling an RC car through a course, given accurate decision making. Using a live wireless video stream as input, we applied several methods of image preprocessing to determine how we could best implement an autonomous vehicle with a multi-dimensional neural network. The three image preprocessing methods we used were: Cropping, Pixel Value Averaging, Contrast-Rounding.

The techniques we employed, as well as the network's performance using these image preprocessing techniques, are presented in the following report.

Methods

Training the Neural-Network

The output was constructed by "hand-driving" the car. Taking in live picture feed from the iPhone™ at every iteration, the car would wait for a keyboard input from MATLAB™ and then proceed down the track according to the user input. This way, each picture (input) would have direction association with it (output) that could be used to train the Neural-Network.

Cropping the Input Images

To test whether "focusing" the images by cropping out the unnecessary portions of the original image improved test error, a cropping algorithm was implemented. The algorithm was used to take the reduced image (size 18x22 pixels) and remove a certain number of rows. The upper rows contained horizon objects such as walls, cabinets, doors, and road irrelevant to the current decision making process, and the bottom rows contained images of the car and road already driven. Due to the fact that a small number of pixels generalized better to the relatively small number of training examples (n = 1000), a smaller image that emphasized the data most crucial to the decision making process was sought after.

Pixel Value Averaging

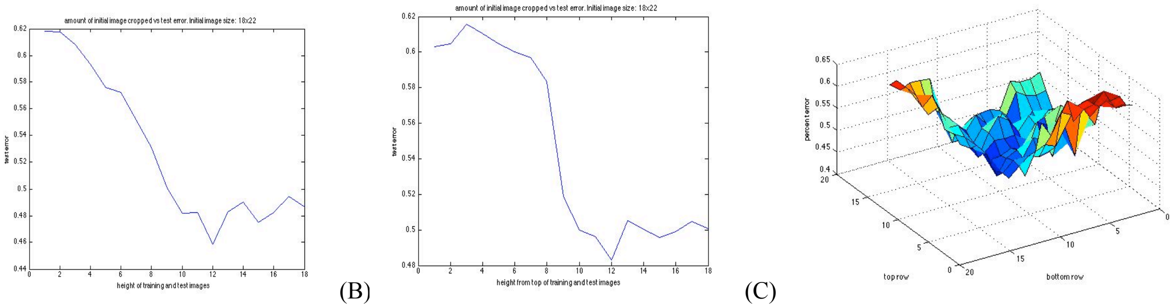
To determine which side of the image had more "white," and thus the path to avoid, an averaging method was implemented. The mean pixel values along each column was taken for each image and that single remaining row was divided into three relatively equal parts. The mean of those three equal parts was taken to produce a three pixel output. These three pixels represented the three portions of the image - left, center, and right.

Contrast-Rounding

To create a more stark difference between the path outlined by printer paper and the rest of the room, a pixel color value-rounding algorithm (contrast rounding) was implemented to turn any pixel with color value > 200 to 255 and any pixel with color value 200 or less to 0. In this way, the path stood out from the rest of the picture, eliminating potential false positives from other white spaces in the frame, like the off-white from the walls. A processed picture is attached in **Figure 1**.

(A) (B)

Figure 1: (A) Original full-sized image. (B) Contrast-rounded image of (A).



(A) (B) (C)

Figure 2: (A) From the top, test error reaches its minimum when 6 rows of pixels are remove. This makes sense as the upper rows are further in the horizon providing roads that do not effect the direction decision currently or is background noise. The increase in error after 6 rows indicates that important data is being removed and the decision approaches random (0.66). (B) From the bottom, test error reaches its minimum when 6 rows of pixels are removed. (C) Combining the two variables, the test error for each permutation was found.

Results

From cropping the images we found cropping certain sections of the input picture could reduce the test error. This analysis was obtained from training data of size 397. To test whether removing pixels improved performance of the neural network, we first removed rows of the picture from the top. From this we found that removing six rows reduced the error from 48.5% to 46.0% **Figure 2 (A)**. Then, we tested removing from the bottom where we found removing six from the bottom reduced test error from 50.1% to 48.3% **Figure 2 (B)**. Having shown that removing certain data from the picture improves results, we then proceeded to run permutations of top and bottom removal. From this analysis we found that by removing eight pixel rows from the bottom six pixel rows from top the lowest possible test error of 40.96% was achieved **Figure 2 (C)**. A representation of a cropped image is shown in **Figure 3**.

Figure 4, shows a thorough investigation of the various preprocessing techniques that were attempted. We see from the figure that the best results were obtained when cropping was executed alone. Surprisingly, exemplifying the path through contrast-rounding actually increased test error. Unlike the original prediction, a combination of contrast-rounding and cropping actually increased test error in comparison to cropping alone.

This phenomenon can be explained by looking at the cropping data. We saw that the ideal cropping was to remain with a 4x22 image. Thus, we can safely say the total shape of the road is not what is important in the decision making process. In fact, making the image dual-colored is essentially data loss, and it may be that it

(A) (B)

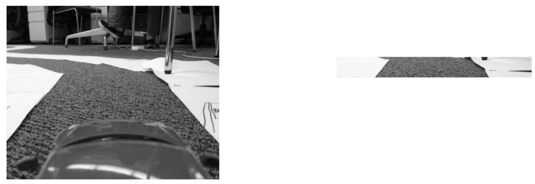


Figure 3: (A) Original full-sized image (not used for testing). (B) Ideally cropped image of (A).

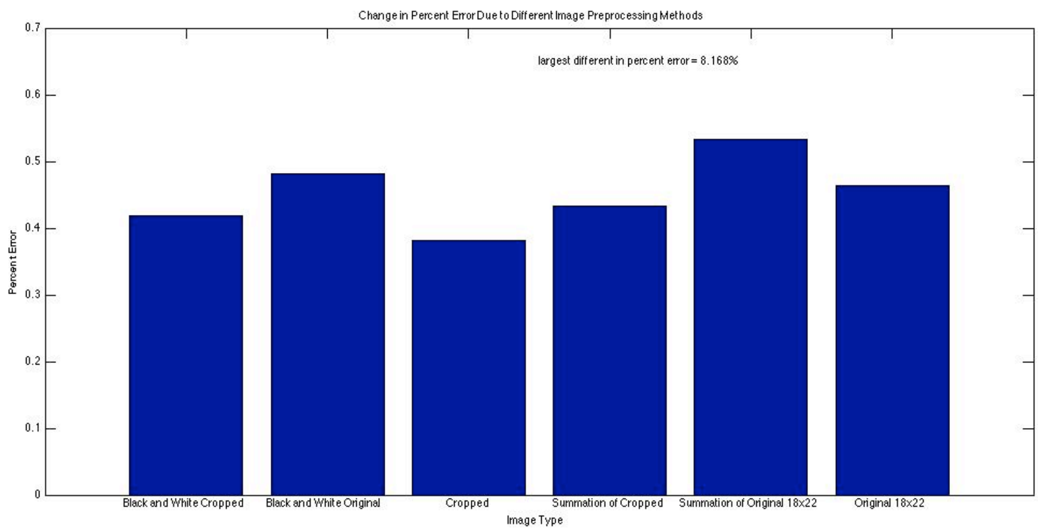


Figure 4: Test error for the various pre-processing techniques averaged from 100 iterations. Cropping was shown to be the only effective preprocessing technique.

causes data loss in the important middle 4x22 image section.

As expected the pixel value summation is not a very good technique to improve performance. Simplifying the entire pixel data to three points is simply too much data loss. Also, averaging may not be the best method as it likely simplifies everything to a medium grey value.

Conclusion

The cropping function is similar to discounting peripheral vision. Coverage, the spacing of retinal ganglion cells on the retina, is not consistent through the human retina. There is a greater density of retinal ganglion cells in the fovea, which keys into the most important regions of the visual field. Our cropping method simulates coverage and gives all weight to the middle portions of the image, where most of the important data was stored. This gives us computational insights into how the visual system works and how it works best when given concentrated information that carries the most important data.

The cropping function is similar to discounting peripheral vision. Coverage, the spacing of retinal ganglion cells on the retina, is not consistent through the human retina. There is a greater density of retinal ganglion cells in the fovea, which emphasizes the most important regions of the visual field. Our cropping method simulates coverage and gives all weight to the middle portions of the image, where most of the important data was stored. This gives us computational insights into how the visual system works and how it works best when given concentrated information that carries the most important data.

By making the images only represent black or white, we are doing a very extreme rendition of monochromatic vision. As seen in **Figure 4**, contrast-rounding was detrimental to the performance of the neural network. This is most likely due to the fact that we were feeding the neural network less information without reducing the dimensionality of the data, which led to a decrease in generalization.

Attempting to simplify the data to three pixels was ineffective. However, there is the potential that if an appropriate weighting algorithm was used to correctly weigh which portions of the picture should correspond to the three pixels, it may be possible to simplify the results to be more representative of the original full size image with greatly reduced dimensionality.

The RC car was ran with the cropped image inputs and ran much more successfully than ever before. This improved model navigated all various realistic courses perfectly. A video of its performance can be found at http://youtu.be/VedN4CGoYoo.

Acknowledgments

Special thanks to Brown University's Thomas Serre Ph.D., Nicholas Franklin, and Robin Martens for their insights into cognitive science and the neural network.

Contributions

Alex Hadik - Engineering, programming, generating training data, presentation and paper presentation

Sukhyun Jung - Engineering, programming, generating training data, presentation and paper presentation

Evan Lester - Engineering, programming, generating training data, presentation and paper presentation