Simple Scenes Classifer

Alex Shroyer

2022-05-04

Abstract

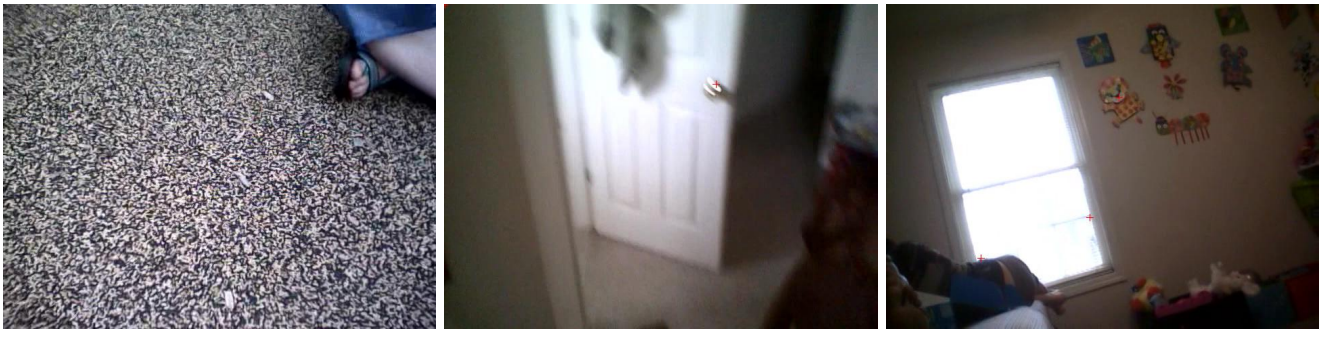
The work described in this paper is meant to assist in automatically classifying images for a psychological and brain sciences research task. The research task aims to show what types of image scenes are preferred by infants at different stages in their cognitive development. The evidence used is a collection of images taken by head-worn cameras, worn by infants of different ages in their day-to-day activities. Until the work of this paper, these images were labeled by hand according to two categories: "simple" scenes with high contrast and relatively few sharp corners and lines, or "regular" scenes with varying contrast and larger numbers of corners or lines. This work describes an image classifier based on a Convolutional Neural Network which is able to classify images with 82.60 - 92.22 percent accuracy. This system should reduce the amount of hand-labeling required and allow the researchers to focus on the more interesting tasks of their work.

Introduction

This project was a collaboration between myself and researchers with Psychological and Brain Sciences at Indiana University who study cognitive development. Specifically, their area of study by uses head-worn, or *egocentric* cameras to capture what infants see from their perspective. The researchers describe the images infants tend to focus on as "simple edges with high contrast - in real-world vision, these show up as doorframes and ceiling corners. We want to know if there are more of these in younger infants' vision". Therefore, this project aims to classify these images as either **simple** or **regular**.

Background and Related Work

Based on the description of a simple scene, one approach is to construct a set of filters or algorithms that attempt to find images with the stated features. I tried this using Harris corner detection(2) but the results were unsatisfactory.



Left to right: simple (0 corners); regular (1 corner on doorknob); simple (2 corners in lower part of window).

Labeled images in each category can have high, low, or no detectable corners, so counting corners is unreliable. Also, because the images vary in contrast, blur, hue, saturation, focus, and other parameters, it is difficult to choose a corner detection threshold for the Harris algorithm that works on the whole data set.

Other possible options which I researched but did not implement include Gray Level Grouping(1) and Performance Metrics for Image Contrast(4) for obtaining contrast data from an image. Related work in image classification can be found in satellite image analysis (3).

Ultimately, heuristic-based or filtering-based approaches rely on how the problem is stated, rather than how the data was actually labeled. Therefore, a neural network approach may be more robust.

Method

Based on the success of Convolutional Neural Networks (CNNs) on similar image classification problems, this project uses a CNN to classify images as either simple or regular.

Before training, the color jpg input images were resized to 32x32 pixels.

This model uses the following layers, with a total of 61370 parameters:

- 2d convolution with 5x5 kernel
- 2d batch normalization
- ReLU activation function
- 2d max-pooling

The next 4 layers are similar to the first 4, with minor differences in in- and out-channel sizes:

- 2d convolution with 5x5 kernel
- 2d batch normalization
- ReLU activation function
- 2d max-pooling

The convolutional layers are followed by linear stages after flattening the output into a 1d vector:

- linear with output size 120
- ReLU
- linear with output size 84
- ReLU
- linear with output size 2
- sigmoid activation function

This model was trained on the enire labeled training data (20% witheld for validation), for 100 epochs, using Adam optimizer and a learning rate of 0.001. During training, the best parameters of the model are kept, based on predicting the correct label on the validation set. These "best" model parameters are then saved to a file for later use in making predictions on unseen data.

Training versus validation accuracy for different data sizes

Even though the model overfits the training data, parameters are only saved when predictions on the validation set improve compared to previous epochs. It is also evident from these loss curves that larger data sizes reduce overfitting, and that regularization of the input data would provide further benefits.

Training was done on an Apple M1 CPU (no GPU was used for this project) with training time shown by Table 1. A large portion of training is spent loading and resizing images. Image loading and model training use multiple CPU cores when possible, which is reported as cpu usage percentages greater than 100.

Table 1: Training time

dataset	N	user(s)	system(s)	cpu	total
small	48	15.50	4.21	499%	3.95
large	11905	203.62	77.42	451%	1:02.24
full	15765	267.80	102.94	453%	1:21.81

Results

After training, the prediction accuracy on subsets of the test set is shown in table 2.

Table 2: Accuracy on validation data

image class	accuracy (%)
simple_older	82.60
simple_younger	86.94
regular	92.22

Correct Predictions

Because the data is already labeled, it is trivial to find correct predictions among the training and/or validation data.

Incorrect Predictions

Similarly, incorrect predictions can be found within the training and/or validation data.

Conclusions

Overall, the CNN model is a good choice for this type of problem, where the specification is sufficiently vague and the amount of labeled training data is sufficiently large. Because the model learns implicit features from the data, it approximates the same implicit and explicit heuristics used by the humans to label the training data.

For example, many **simple** images in the data are of indoor scenes, which may imply label outdoor scenes are usually **regular**.

This feature learning is what makes neural networks well suited to classification and regression type tasks. CNNs offer a further improvement by leveraging spatial locality in the images.

Potential Mispredictions

Figure 1: Predicted regular, seem simple

Figure 2: Predicted simple, seem regular

Given the ambiguity in the labeled data, there are bound to be some predictions that do not match an intuitive explanation. It is possible that other factors which happen to occur in one class of images, but are not explicitly distinguishing features, have crept into the model. For example, most simple scenes are indoors, so the presence of trees, grass, or sky (or colors and textures like these things) might cause the model to predict "regular" even if the scene otherwise fits the "simple" definition.

Ultimately, researchers must decide whether these images are classified correctly.

References

[1] Chen, ZhiYu and Abidi, Besma R and Page, David L and Abidi, Mongi A, *Gray-level grouping (GLG): an automatic method for optimized image contrast Enhancement-part I: the basic method*, IEEE, 2006.

[2] Derpanis, Konstantinos G, *The harris corner detector*, Citeseer, 2004.

[3] Liu, Qingshan and Hang, Renlong and Song, Huihui and Li, Zhi, *Learning multiscale deep features for high-resolution satellite image scene classification*, IEEE, 2017.

[4] Tripathi, Abhishek Kumar and Mukhopadhyay, Sudipta and Dhara, Ashis Kumar, *Performance metrics for image contrast*, 2011.