Friendly Streets: A Street View Classifier for Cautious Cyclists

Josh Sennett
University of Massachusetts, Amherst
jsennett@umass.edu

Evan Rourke
University of Massachusetts, Amherst
erourke@umass.edu

Abstract

The goal of this project is to classify street-view images as "bike-designated" or "not". Existing research has shown that convolutional neural networks (CNNs) are well-suited for scene classification and object detection, but have not been applied to classify bike-designation. The three contributions of this work are 1) to create a tool to generate high-quality datasets of street-view images labeled as bike-designated or not; 2) to develop a neural network to predict this label; and, 3) to establish a human baseline with which to compare the accuracy of our neural network. By fine-tuning a pre-trained ResNet neural network, we achieve 68.0% overall accuracy on a test set of 3,454 unseen images, outperforming a human benchmark of 59.7% on 1,296 images.

1. Introduction

Cyclists depend on routing applications to find safe and enjoyable bike routes. In these applications, roads may be "bike-designated" based on data from public records, privately-owned data sources, or open-source communities. The classification of bike-designation is often outdated, inaccurate, or unavailable, presenting a challenge for mapping software and cyclists to plan safe and optimal routes.

To approach this problem, we develop a convolutional neural network to classify street-view images of roads as bike-designated or not. We train our network on a dataset of street-view images labeled "bike-designated" or "not", which we generated by integrating OpenStreetMap [5] road classifications with Google's Street View API [1].

"Bike-designation" refers to whether a road contains infrastructure to make a road safer or more enjoyable for cyclists. OpenStreetMap, which we use as our source of road classification data, defines bike-designation as whether a roadway (for motor-vehicles) contains either a bike lane (a lane within a roadway) or a bike path (a lane alongside a roadway, but separated by a barrier). Bike-designated roads may also have shared-road arrows ("sharrows"), colored bike lanes, physical barriers, and cautionary ("share

Figure 1: Shared road markings ("sharrows"), colored bike lanes, and barriers are common features of bike-designated roads.

the road") street signs.

Developing a bike-designation street-view image classifier faces several challenges. First, there is no existing public, large-scale database of street-view images with the bike-designation label. Second, bike-designated roads contain a variety of different types of cycling infrastructure, and each type of infrastructure may look different depending on which city and country the road pertains to. Third, street-view images may not be positioned or oriented correctly to capture the relevant features. Last, street-view images are taken in real-world settings, and face several factors that decrease image quality including feature occlusion, background clutter, differences in illumination, and blur.

To overcome the first challenge, we generate a labeled dataset of street-view images by integrating OpenStreetMap road classification data with Google's Street View API. To overcome the remaining challenges, we develop a convolutional neural network to classify street-view images as bike-designated or not. Previous research has demonstrated the

(a) Bike-designated

(b) Not bike-designated

Figure 2: Example street-view images of bike-designated and not bike-designated roads.

effectiveness of CNNs for scene understanding and object detection in real-world settings, where viewpoint variation, occlusion, background clutter, illumination, and intra-class variation are expected [cite 231].

The paper is structured as follows. Section 2 presents related research. Section 3 explains our approach to developing a labeled dataset, neural network, and human baseline. Section 4 contains the results of our experiments to improve our network's performance. Section 5 concludes with lessons learned and directions for future research.

2. Background and Related Work

2.1. Scene Classification

Convolutional neural networks have been widely successful in object classification tasks [4]. Trained on massive datasets, these networks have been able to nearly achieve or even exceed human performance benchmarks.

Scene classification, however, is a more challenging, generalized task than object classification: a scene class may be correlated with the presence of objects, but such objects may appear in other scene classes. Correctly classifying a scene depends not only on objects but also attributes (relating to spatial envelope, materials, lighting, and textures).

Xiao et al. developed the SUN Database, the first large-scale scene understanding database with 899 scene categories and over 130,000 labeled images[11]. Shortly thereafter, Patternson and Hays developed the first large-scale scene attributes dataset and demonstrate the predictive power of attributes for classifying scenes [6]. Most recently, Zhou et al. develop the Places database, a database of 434 scene categories and over 10 million scene images [14]. Using the Places database, Zhou et al. surpassed previous scene classification benchmarks using VGG, GoogLeNet, AlexNet, and ResNet convolutional neural networks.

2.2. Street View Images

Several researchers have used Google Street View images as a source of natural images; Google Street View cars take panoramic photos along roads, providing a rich source of images of outdoor scenes. Netzer et al use Street View images to generate the Street View House Number dataset for digit classification. Similarly, Slavkovikj et al.

use Street View images to develop a road-texture dataset for classification [cite]. Gebru et al. develop a car brand classifier [cite], and Hershey and Wulfe use deep learning to recognize geo-location from city Street View images.

3. Approach

3.1. Data

To generate a labeled dataset of bike-designated street-view images, we created a tool to integrate OpenStreetMap road classifications with Google's Street View API. This tool does the following:

1. Download OpenStreetMap data for a particular region

2. Extract road segments (way elements) for this region using particular criteria

3. Extract road points (node elements) for these ways using particular criteria

4. Calculate road orientation for each for these nodes

5. Query Google's Street View API using node coordinates and road orientation

6. Save the street-view image with the road classifications

Specifically, to create a dataset of bike-designated images, we did the following.

In step 1, we used OpenStreetMap data for four cities: Boulder, Pittsburgh, Portland, and Seattle. We chose these regions because of the quality and availability of bike-designation for roads in these cities, and because these cities have near-balance between bike-designated and non-designated roads (in total, 57% and 43%).

In step 2, we filtered road segments to only include road types "motorway", "primary highway", "secondary highway", and "tertiary highway". Other segment types included non-road segments not relevant to our analysis such as footpaths and bike-paths, as well as residential roads, which were often unlabeled with road classifications.

In step 3, we filtered nodes to only include the midpoint of each road. Given the cost of the Google Street View API ($7 per 1000 images), we could not capture Street View images at all millions of nodes of all relevant highways. Nodes

are typically several meters apart; so, capturing Street View images at adjacent nodes will result in similar images with a slightly shifted camera perspective. Extracting a single image for each road segment (rather than extracting multiple images at a subset of road segments) increases image heterogeneity while keeping our API costs reasonable.

In step 4, we determine the required camera orientation such that each image is pointed in the direction of the road. To do so, we calculate the geodesic azimuth (angular distance from north) between a node and the subsequent node. For one-way roads, we orient the camera to face in the forward-direction of the road.

In step 5, we query the API to determine if a Street View image is available for the given coordinate in a radius of 10 meters. We found that reducing the search radius to 10 meters decreased the chance that images of nearby (but different) roads were not accidentally downloaded in the case that a road did not have street-view images. Then, if an image is available, we download a 640x640 image using the node coordinate and the calculated camera orientation.

3.2. Neural Network

why we chose pre-trained places365, why we chose to fine-tune, how we came up with the final model pictures of training/val loss accuracy

3.3. Human Benchmark

NOTE: INCLUDE A SCREENSHOT OF THE HUMAN BENCHMARK INTERFACE if you have per-person scores, maybe include the standard deviation of scores and average images classified per person

4. Experiments

different models: fc1, fc2, fc3, w/wout dropout, leakyrelu different optimizers: adam, SGD with momentum + L2 reg hyperparameter search: learning rate, reg strength, hidden layer size how we chose: trained on 10% of dataset, looked at loss/acc curves

show images, CAM, attributes, with our prediction top accuracy for our models (68%)

final model: fc3 (400 units?), 10% dropout and leakyrelu, SGD + momentum, L2 reg

This section begins with what kind of experiments you're doing, what kind of dataset(s) you're using, and what is the way you measure or evaluate your results. It then shows in details the results of your experiments. By details, we mean both quantitative evaluations (show numbers, figures, tables, etc) as well as qualitative results (show images, example results, etc).
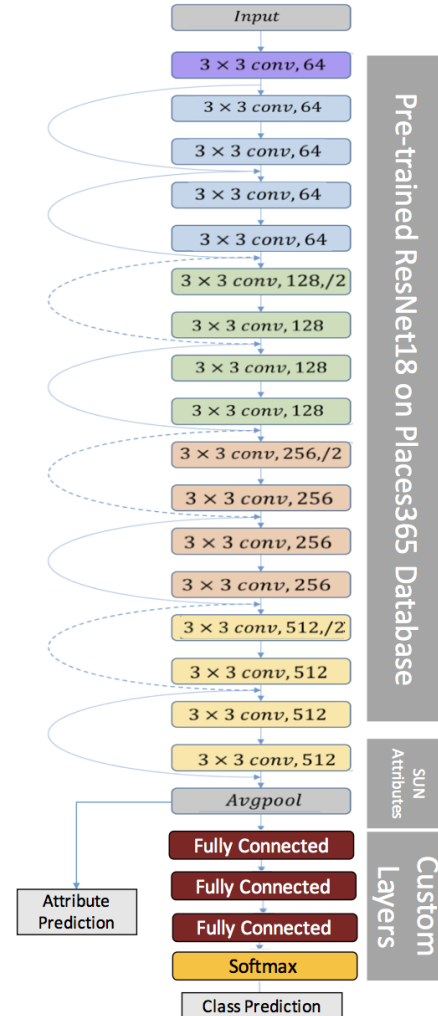
5. Conclusion

What have you learned?

Figure 3: Final model: Pre-trained ResNet18 with three fine-tuned fully-connected layers.

Suggest future ideas.

List and number all bibliographical references at the end of your paper. When referenced in the text, enclose the citation number in square brackets, for example.

TODO: add the rest of the citations here from cs682_final_report.bib

[9] [7] [8] [15] [13] [10] [2] [12] [14] [3]

References

[1] Google. Google street view image api, 2018.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[3] W. Koehrsen. Transfer learning with convolutional neural networks in pytorch, 2018.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[5] OpenStreetMap. Openstreetmap, 2018.

[6] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2751–2758, June 2012.

[7] G. Patterson, C. Xu, H. Su, and J. Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81, 2014.

[8] A. G. Rundle, M. D. Bader, C. A. Richards, K. M. Neckerman, and J. O. Teitler. Using google street view to audit neighborhood environments. *American journal of preventive medicine*, 40(1):94–100, 2011.

[9] V. Slavkovikj, S. Verstockt, W. De Neve, S. Van Hoecke, and R. Van de Walle. Image-based road type classification. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 2359–2364. IEEE, 2014.

[10] M. Wang. Multi-path convolutional neural networks for complex image classification. *CoRR*, abs/1506.04701, 2015.

[11] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, pages 3485–3492. IEEE, 2010.

[12] S. Zagoruyko and N. Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016.

[13] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *CoRR*, abs/1412.6856, 2014.

[14] B. Zhou, A. Khosla, À. Lapedriza, A. Torralba, and A. Oliva. Places: An image database for deep scene understanding. *CoRR*, abs/1610.02055, 2016.

[15] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2017.