







Developing a real-time foot traffic counter in New York City using publicly available camera feeds


Gregory Dobler^{1,2}, Jordan Vani², Trang Dam^{2,3}, Name4 Surname², Name5 Surname², Name6 Surname², Name7 Surname^{1,2,3}, with the Lorem Ipsum Consortium¹


1 Affiliation Dept/Program/Center, Institution Name, City, State, Country


2 Affiliation Dept/Program/Center, Institution Name, City, State, Country


3 Affiliation Dept/Program/Center, Institution Name, City, State, Country

 These authors contributed equally to this work.

 These authors also contributed equally to this work.

 Current Address: Dept/Program/Center, Institution Name, City, State, Country

 Deceased

 Membership list can be found in the Acknowledgments section.

* correspondingauthor@institute.edu

Abstract

Filler text

Introduction

Cities are complex systems characterized by the interaction of people, technology, and the built environment (Patorniti, Stevens, and Salmon, 2017). Correspondingly, the flow of people through the built environment is complex and multi-modal—incorporating public transportation, personal vehicles, bicycles, and pedestrian traffic. Accordingly, quantifying movement across all modalities at a city-wide scale and at a granular temporal scale is vital to the systematic study of cities and the development of efficient and equitable mobility systems (“The City We Need 2.0 Towards a New Urban Paradigm,” 2016). Extensive work has been conducted to measure, model, and forecast public transportation and vehicular traffic (Vlahogianni, Golias, and Karlaftis, 2004; Pelletier, Trépanier, and Morency, 2011; Dou et al., 2015). However, quantifying pedestrian mobility remains difficult (Iacono, Krizek, and El-Geneidy, 2010).

In practice, cities widely rely on pedestrian count surveys, which are limited in temporal range (Cottrell and Pal, 2003). Separately, efforts to measure human mobility have been spurred by the boom of social media data, mobile-phone data, and transactional data (González, Hidalgo, and Barabási, 2008; Hasan, Schneider, Ukkusuri, and González, 2013; Wu, Zhi, Sui, and Liu, 2014). Nonetheless, these efforts do not directly measure pedestrian traffic and are limited by user participation, the lack of real-time data, and/or proprietary data sources. Concurrently, computer vision has been widely used for pedestrian detection in a variety of applications (e.g., autonomous vehicles, surveillance, etc.) (Enzweiler and Gavrilu, 2009). A 2012 review of pedestrian detection algorithms in traffic surveillance imagery underlined the difficulty of detecting pedestrians in real-time with current algorithms (Li, Yao, and Wang, 2012); but, methods have since improved. Specifically, Faster Region-based Convolutional Neural

Networks (Faster R-CNNs) have enabled real-time object detection with a throughput rate of 5fps using the VGG-16 network structure (Ren, He, Girshick, and Sun, 2017).

Accordingly, this work demonstrates the application of a Faster R-CNN for pedestrian detection in real-time over a network of 650 New York City (NYC) traffic cameras. By analyzing the resulting data we illustrate how pedestrian foot traffic is a feature of the city—a feature that can be used to study the city itself. The goal of this work is to provide a public real-time measure of pedestrian foot traffic at a city-wide scale that can ultimately be used to further the study of the city as a complex system.

Materials and methods

Camera network

The New York City Department of Transportation (DOT) maintains a network of real-time traffic cameras (n=650) across the NYC metro area. The recorded camera feeds are publicly available online and are served at approximately one frame per second—with each frame being a 352x240 RGB .jpg. All together, the 650 cameras are distributed across the five NYC boroughs (see Table 1 and Fig 1 (map of cameras across boroughs)) and capture a continuous record of the NYC streetscape including roadways, building facades, and pedestrian facilities. With our focus on pedestrian traffic, each camera was hand-labelled as including pedestrian facilities or not; Fig 2-3 (pedestrian facility images) illustrate the distinction between cameras with and without pedestrian facilities.

Table 1. DOT cameras by New York City borough.

Borough	Bronx	Brooklyn	Manhattan	Queens	Staten Island	Bridges	Total
Total cameras	44	105	226	195	41	39	650
Cams w ped. facilities	13	30	157	65	11	5	281

Training and testing data

3918 daytime images were scraped from 17 cameras on April 30th, 2016 and June 19th, 2016. The 3918 images were then labelled by hand for positive pedestrian examples and negative examples using bounding boxes with a constant aspect ratio of 3:4. These labels were not exhaustive (i.e., not all pedestrians were labelled). Across the 3918 images, 16022 positive examples and 41449 negative examples were labelled—approximately a 2:5 (pos:neg) ratio. XML files were subsequently written for each image, following the PASCAL Visual Object Classes format, and was then split using a 70:30 train:test split.

Faster RCNN training

Using the training set described above a Faster RCNN was trained using Tensorflow following the VGG16 network structure. Training parameters included: (1) a learning rate of 0.0005; (2) a Region Proposal Network (RPN) batch size of 256; (3) an RPN positive overlap of 0.7; and, (4) a minimum RPN size of 2x2. Using this setup, the network was trained for a total of 90,000 iterations on a GeForce GTX 1080 Ti GPU.

Model performance

(SEE: [hadive/src/assorted/confmatrix-prec-recall.py](#) for reference).

Model performance was first assessed using the test set from the original labelled data. For all labels in the test set, the centroid of the bounding boxes were extracted. Then, each corresponding image was passed through the Faster RCNN and the resulting bounding boxes for all detections were written to file. Accordingly, for each image the count of true negatives, false positives, true positives, and false negatives were found through a pairwise elimination process of labels (points) to detections (boxes). Each negative label centroid found outside all detection bounding boxes was counted as a true negative. If a negative label centroid was found within a detection, the detection was eliminated from future matches and counted as a false positive. For a given positive label centroid, if it was found within a detection, the detection was eliminated from future matches and counted as a true positive. If a positive label did not have a corresponding detection, this was counted as a false positive.

Secondly, a separate validation set was created using randomly selected daytime and nighttime images from 3 cameras (n=286). This validation set was then exhaustively labeled for pedestrians. Similarly, a process of pairwise elimination was used to receive the count of true negatives, false positives, true positives, and false negatives. In this case, if a positive label centroid was found within a detection, the detection was eliminated from future matches and counted as a true positive. If the positive label centroid was found to not fall within any detections, this was counted as a false negative. The remaining detections were counted as false positives. Accordingly, both the precision and recall of our detections were calculated. Lastly, this validation set was bootstrapped into 14 subsamples with a sample size of 20, to extrapolate the population precision and recall of our model and corresponding 95% confidence intervals.

Counting pedestrians

Counting pedestrians in all cameras was conducted in a continuous loop. Looping over each camera, an image was downloaded (if available) and the time at download was recorded. The downloaded image was parsed for available metadata overlain on the image (camera direction and timestamp). Then, the image was passed to the Faster RCNN which output the number of pedestrian detections. Lastly, the resulting data (camera id, time at download, camera direction, overlain timestamp, and the number of pedestrian detections) was saved to a local database. In addition, for each loop over all 650 cameras, 1 image was saved to file at random. Ultimately, looping over all 650 cameras took approximately 70 seconds, enabling the detection of pedestrian foot traffic at a temporal resolution of just over 1 minute.

Filler text

Fig 1. Bold the figure title. Figure caption text here, please use this space for the figure panel descriptions instead of using subfigure commands. A: Lorem ipsum dolor sit amet. B: Consectetur adipisicing elit.

Results

Filler text

Discussion

Filler text

Conclusion

101

Filler text

102

Supporting information

103

S1 Fig. **Bold the title sentence.** Descriptive text (optional).

104

Acknowledgments

105

Filler text

106

References

1. Conant GC, Wolfe KH. Turning a hobby into a job: how duplicated genes find new functions. *Nat Rev Genet.* 2008 Dec;9(12):938–950.