Tasty City: Photo Geolocation using Street and Restaurant Data

Tianyu Su (6.869)
Department of Urban Studies and Planning
Massachusetts Institute of Technology
sutianyu@mit.edu

Zhuangyuan Fan (6.869)
Department of Urban Studies and Planning
Massachusetts Institute of Technology
yuanzf@mit.edu

Abstract

In computer vision, the photo geolocation problem has been usually approached at a global scale or regional scale. In this project, we derive knowledge from urban studies and present a classification based method looking into neighborhood scale photo geotagging. We subdivide Boston Chinatown into multiple geographic cells and train a deep network using 20K google street view images labeled with local restaurant density and street speed limit. We show that this multitask model could output a probability distribution over a couple of cells in the neighborhood.

1. Introduction

1.1 Motivations

Satellite positioning technologies like GPS introduce convenience in navigation and expand human's capacity to explore the urban and natural environment. However, there are occasions when we lose our GPS signals in dense urban areas, which cause difficulties in navigation and create unpleasant experience. Recently, image-based geotagging and geolocation has become an emerging field as the great progress in computer vision. While a few models like PlaNet [11] and IM2GPS [4] achieve a level of sound accuracy on global and regional scale geotagging, little has been done focusing on street-level image geotagging.

In the context of the urban environment, built environment attributes such as vegetation, colors, building height, building density usually reflect the characters of the context and influence human perceptions of surroundings [2]. But are they able to provide enough information to geotag a photo at the street level? To answer this question, we explore the street-level photo geolocation in Boston Chinatown with Google Street View images.

1.2 Research questions

Given an image in Boston Chinatown, how to identify the fine-grained location information based on the features of the built environment in the image? The fine-grained location information here includes the "restaurant zone" information and "street speed limit" information, which could give us a better sense of where the image is located.

2. Related work (Tianyu Su)

In this project, we have reviewed papers covering image geotagging methods and the applications of computer vision methods in urban studies.

With the potential applications in navigations and photo geotagging, image geolocation has become an emerging field in the computer vision field. Classification models based on the neural network has been widely applied and reach a sound geotagging accuracy on large geographical scales like continent-scale and country-scale. IM2GPS [4]

takes a data-driven scene matching and nearest neighbor approach to estimate a distribution over geographic locations globally. PlaNet [11] uses Convolutional Neural Networks (CNN), augmented by a long short-term memory (LSTM) architecture, to classify images in photo albums into a global grid and reach an accuracy level of 71.3% on continent-scale and 53.6% on country-scale, overperforming that of IM2GPS. [9] takes a similar path including deep CNN and metric learning to classify landmarks on a noisy and diverse Dataset. However, both IM2GPS and PlaNet perform poorly on street-level, with an accuracy of, respectively, 2.5% and 8.4%. In this project, we want to explore some possibilities for street-level image geolocation using our knowledge of the urban environment.

The urban study area also sees a rising trend of utilizing methods and models of computer vision to understand the built environment. [8] takes a semantic segmentation approach to indicate physical urban changes by comparing Google Street View images taking on the same urban areas now and then. [3] uses an approach of the same kind to map previously unmeasurable elements in the urban environment, for example, sky and trees. Most of the urban studies using computer vision follow the same paradigm: given a location, collecting the street view images, indicate urban environment elements using semantic segmentation or similar approaches.

In our project, we would like to flip this paradigm by inspirations we get from geotagging works, which is: given an image, how we could infer some location indicating information for an approximation of the location the image is taken. We use deep convolutional neural networks to deal with this multi-task classification problem.

3. Approach and algorithms

3.1 Data (Tianyu Su)

3.1.1 Google Street View images

To construct the collection of street view images for our multi-task classification problem, we first obtain the street vector data of Boston from OpenStreetMap, an open-source map platform [5]. We then generate point data with GPS coordinates by equally sampling the street vectors every 30 feet. Finally, 20,000 street view images are retrieved via the Google Maps API. This project uses eight street view images (i.e., 0, 45, 90, 135, 180, 225, 270, 315) from each sample point.

Fig 1. sample points and eight street view images of a selected point

3.1.2 Boston street segments

The street segment data is collected from Analyze Boston (https://data.boston.gov/dataset?q=street), which includes street names, street types, and street speed limits. We select street speed limits as an indicator for the street environment and then use it as a geolocation label.

3.1.3 Boston restaurant data

We retrieve the location data of all the active restaurants in Boston from referenceUSA (http://www.referenceusa.com/Home/Home), which provides open-source social and economic data. Restaurant data is processed as a density map and acts as an indicator of urban vibrancy.

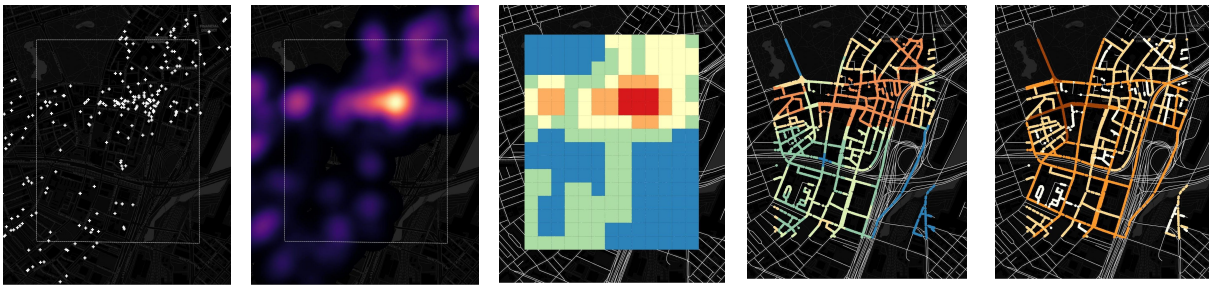
3.2 Approach and Model (Zhuangyuan Fan)

We treated this geolocation task as a classification problem and we assigned each street view image two labels by dividing Boston Chinatown into zones according to two criteria: the local restaurant density and street speed limit. And we used deep convolutional neural networks (DCNN) with two classifiers to infer the location of the street image. The DCNN model will learn the shared deep features to support the classification tasks without actually measuring the detailed built environment attributes.

3.2.1 Label the street view images

In our design, we created two layers of labels for each street view image. Firstly, we labeled each image by restaurant density at location where the image was taken. We subdivided Boston Chinatown into 300 x 300 feet net for restaurant zone labeling and created a restaurant quadratic kernel density map with business location data from referenceUSA. And we calculated the mean density value of each grid. Then all google street view images are labeled according to the zones it falls in (Figure 2).

Secondly, we labeled each image by the street speed limit at its location. There are 6 levels of the speed limits in total, we group the streets at speed limits(5-15) to one group for better-balanced dataset.



a) Restaurants locations b) Restaurant density map c) Restaurant zones with scores d) Assign restaurant zone label e) Street speed limit label

Fig 2. label google street views by location restaurant density and street speed limits

food_zone	*kernel density value*	*sample size*
0	*0-1.7*	*1624*

street limit zone	*street limit value*	*sample size*
0	*5, 10, 15*	*2688*

1	*1.7 - 5.9*	*4184*
2	*5.9 - 11*	*5136*
3	*11- 18.1*	*3608*
4	*18.1-28*	*5112*

Table 1a) images classifier 1: restaurant label

1	*20*	*7848*
2	*25*	*7776*
3	*30*	*1352*

Table 1b) images classifier 2: speed limit

3.2.2 Deep Neural Network for classification

We used a densenet based neural network to classify street view images and infer its location. Following the last convolution layer of DenseNet, we concat two fully connected layers and softmax functions. The loss for the restaurant density task and the street speed limit task are defined as $L_{restaurant}(w)$ and $L_{speed}(w)$ respectively. We used cross-entropy loss and the target is defined as $min_w\{(\sum L_{restaurant}(w) + L_{speed}(w))\}$. The model architecture is illustrated in Figure 2.

3.2.3 Location inference with two labels

With two labels, we could reduce the uncertainty about a photo and outputs a probability distribution over the neighborhood.

4. Experiments

4.1 Model Training (Zhuangyuan Fan)

We divided 19,560 google street view images into training subset(70%), testing set (15%) and a validation set (15%). And the model output are images with their top 1 accuracy label and top 2 accuracy label. From the output model, we could infer the location where the photo was sampled.

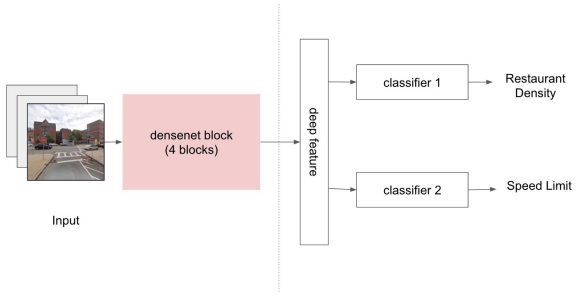


Fig. 3. The model architecture of DCNN

Loss Function	Cross-entropy loss
Classifier (tasks)	2
Optimizer	SDG (learning rate =0.001, decay by 10 every 10 epochs, momentom =0.9)
batch size	32

Table 2. Hyper parameters

4.2 Classification accuracy of the DCNN model

The classification results are shown in fig 3. and table 3. The color in the map corresponds with the color in the table. The model was able to predict 65.7% of images at their correct restaurant zone label and correct speed limit label. And we noted that some locations are subjected to misclassification. If the image was located at an

intersection, the speed limit label it was assigned to only reflect the speed limit of one direction, which will contradict the model condition (Figure 5a). And some images, while not at high density restaurant zones, it also resemble some visual characteristics of the restaurant zones such as color and window decorations(Figure. 5b).

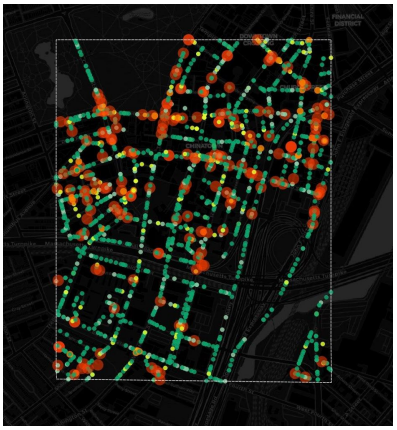
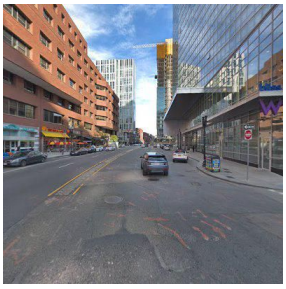


Fig 4. Classification accuracy map

description		Accuracy	# of sample
Have both labels corrected		65.7%	1179
top 1 restaurant label correct	top 2 speed limit correct	13%	234
top 1 speed limit corrected	top 2 restaurant label correct	9.3%	162
top 2 restaurant label correct	top 2 speed limit correct	2.23%	40
other		10%	180

Table 3. classification accuracy of DCNN model with 1795 GSV images in test and validation set



5a) Street intersection:
Restaurant Zone Label: 4
Restaurant Zone Pred:4
Speed Zone Label: 15
Speed Zone Pred: 35

5b) Overlooked feature: color
Restaurant Zone Label: 1
Restaurant Zone Pred:4
Speed Zone Label: 20
Speed Zone Pred: 25

Fig 5. examples of incorrectly classified images

4.3 Location Inference with label

After classification, we will use the predicted labels of the image to infer its location. For example, as is shown in Figure 6, the image labeled as restaurant zone 4 and speed limit zone 25 is most likely to exist in these four cells. The image looks are still very diverse, but most of them present visually narrower streets and tall facade. This shows that we could use more classes (tasks) in the model to improve the geolocation accuracy in the future.

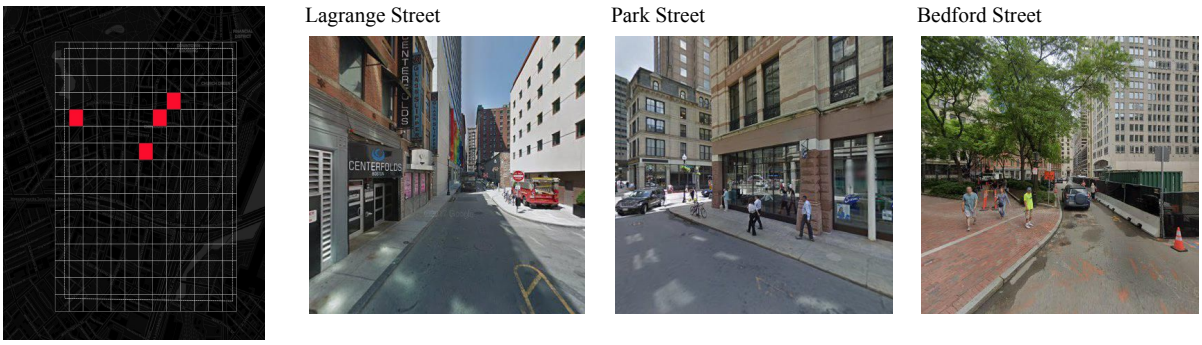


Fig 6. Top 4 confident locations these images fall into

5. Conclusion and limitation

Image geolocation is similar to a cognitive task that experience with the context will enable people and animals to tell where they are even without a map. We presented a street-level geotagging method using knowledge from urban studies to label images by its intangible characteristics shaped by human design and intervention. Future work will involve embedding more classification tasks in one deep neural network model and scaling up the region of study.

References

[1] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, "Geometry-Aware Learning of Maps for Camera Localization," presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2616–2625.

[2] J. Gehl, *Life Between Buildings: Using Public Space*. Island Press, 2011.

[3] F.-Y. Gong, Z.-C. Zeng, F. Zhang, X. Li, E. Ng, and L. K. Norford, "Mapping sky, tree, and building view factors of street canyons in a high-density urban environment," *Build. Environ.*, vol. 134, pp. 155–167, Apr. 2018.

[4] J. Hays and A. A. Efros, "IM2GPS: estimating geographic information from a single image," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, 2008, pp. 1–8.

[5] C.-B. Hu, F. Zhang, F.-Y. Gong, C. Ratti, and X. Li, "Classification and mapping of urban canyon geometry using Google Street View images and deep multitask learning," *Build. Environ.*, vol. 167, p. 106424, Jan. 2020.

[6] P. Mirowski *et al.*, "Learning to Navigate in Cities Without a Map," in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Curran Associates, Inc., 2018, pp. 2419–2430.

[7] E. Muller-Budack, K. Pustu-Iren, and R. Ewerth, "Geolocation Estimation of Photos using a Hierarchical Model and Scene Classification," presented at the Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 563–579.

[8] N. Naik, S. D. Kominers, R. Raskar, E. L. Glaeser, and C. A. Hidalgo, "Computer vision uncovers predictors of physical urban change," *Proc. Natl. Acad. Sci.*, vol. 114, no. 29, pp. 7571–7576, Jul. 2017.

[9] K. Ozaki and S. Yokoo, "Large-scale Landmark Retrieval/Recognition under a Noisy and Diverse Dataset," *ArXiv190604087 Cs*, Jun. 2019.

[10] S. Ruder, "An Overview of Multi-Task Learning in Deep Neural Networks," *ArXiv170605098 Cs Stat*, Jun. 2017.

[11] T. Weyand, I. Kostrikov, and J. Philbin, "PlaNet - Photo Geolocation with Convolutional Neural Networks," in *Computer Vision – ECCV 2016*, Cham, 2016, pp. 37–55.