# Using Convolutional Neural Networks to Predict Automobile Ownership with Satellite Images

Shenhao Wang
shenhao@mit.edu

Jason Lu
jasonllu@mit.edu

## Abstract

*This work uses convolutional neural networks (CNNs) to predict the automobile ownership of the households in the United States. The explanatory variables include the socio-economic variables from a large-scale national household travel survey (NHTS) and the satellite images of the built environment of the locations where households live. This study explores three perspectives: (1) prediction accuracy with the end-to-end CNN architectures as opposed to the traditional handcrafted feature engineering, (2) multi-scale feature representations owing to the scale-invariance properties of images, and (3) representation learning by visualizing activation maps and convolutional layers. Correspondingly, we found (1) that using even only the satellite images and the simplest CNN models can achieve higher prediction accuracy than the expensive NHTS dataset; (2) that slightly surprisingly, the multi-scale image preprocessing does not improve prediction; and (3) that CNN models can automatically identify the semantically meaningful regions on the satellite images. Future work should use deeper CNN architectures that take into account the multi-scale feature structure to improve the model performance and facilitate model interpretation.*

## 1. Introduction

An excessive amount of automobiles lead to various urban diseases, such as congestion, air pollution, car accidents, greenhouse gas emission, and accordingly severe time and monetary losses [18, 32]. Given the importance of automobiles in daily life, researchers have long been examining the factors that motivate people to own automobiles [24] with the purpose of nudging people away from using automobiles to more sustainable and healthy travel patterns. The most common explanatory variables for automobile ownership are the socio-economic variables, such as households' income and education, and the built environment factors, such as density, diversity, and design of the urban environment [11, 7]. The built environment factors are intuitive explanatory variables; for example, people are more likely to own automobiles in the sparse suburban regions compared to the dense Boston areas. While the relationship between car ownership and built environment has been an enduring question for decades, it is also found that the built environment factors can only explain a trivial amount of variation in car ownership [7], leading to the belief that the further research into this topic might be unnecessary or even misleading [26].

However, the classical studies often extract the built environment factors from the bird-view satellite images based on experts' knowledge. This approach of using handcrafted features has limitations, since it can eliminate valuable information that can help prediction but is not a-priori known to researchers. This approach is similar to that in the classical methods used in computer vision (CV) tasks, in which researchers use specific filters - such as low-pass filters or edge detectors - to extract information for prediction. Based on the fact that the end-to-end deep learning methods can dramatically improve prediction in image recognition tasks, we conjecture that a convolutional neural network that directly uses satellite images can also achieve better prediction accuracy than the classical methods in predicting automobile ownership.

This study examines households' (HHs) car ownership by using the socio-economic information collected from the national household travel survey (NHTS) and the satellite images of the HHs' census tracts' geographical location with CNNs. Specifically, we explore three questions: (1) **prediction.** whether the CNN models with satellite images can outperform the predictive models with only the classical NHTS dataset that incorporates certain built environment variables; (2) **multi-scale features.** whether the multi-scale satellite images created by using Gaussian filters and downsampling can help improve the CNN models; (3) **representation learning.** whether the CNNs can automatically detect the important geographical regions on the satellite images. In our experiments, the initial dataset comes from the NHTS 2017 [1], and the satellite images are augmented to the dataset by using the geographical location information of the HHs. Our CNN architecture follows the standard LeNet

[1] url: https://nhts.ornl.gov/

and AlexNet [16] architectures.

This paper is organized as following. Section 2 reviews four categories of related studies. Section 3 introduces models, data, and setup of experiments. Section 4 presents the experimental results. Section 5 concludes our findings and discusses future research directions.

## 2. Related Studies

**Car Ownership and Built Environment.** More than a dozen studies have explored the relationship between built environment and travel behavior [2, 6, 17, 30]. Using travel diary data from the travel behavior survey in Denver, Kwoka et al. [17] concluded that the built environment on origins and destinations accounted for the variation of travel mode choices. Using travel diary data from Boston and Hong Kong, Zhang [36] elucidated that the magnitude of density effects on mode choice was similar to that of travel costs. After inspecting more than fifty pertinent studies, Ewing and Cervero [11] substantiated that the built environment factors are significantly associated with travel behavior, although the magnitude of their relationship is not large. In fact for the past several decades, scholars have not achieved a consensus about the magnitude of the built environment on travel behavior. Some studies charged that, although urban density matters to automobile ownership, the magnitude of impact is small, if not inconsequential [26, 27]. However from a methodological perspective, all the past studies used handcrafted built environment features to explain travel behavior. One reason of the weak predictive power of built environment can be attributed to the handcrafted feature engineering process. The feature engineering based on experts' knowledge compress the information from the satellite images, and this information compression might lead to the lose of valuable predictive information.

**CNN.** Our critique is similar to the critique in the CV field, in which researchers have found strong evidence that the end-to-end automatic feature learning of CNNs can achieve much higher prediction accuracy than the traditional handcrafted feature engineering methods [20, 4]. While some debate still exists concerning the appropriate degree of automation [21], it is undeniable that the automatic representation learning can retain more valuable information than the classical methods. In the recent years, researchers have developed for image recognition tasks several CNN benchmark architectures, such as AlexNet [16], GoogLeNet [31], and ResNet [14]. Recently, the architectures tend to become deeper to facilitate the representation learning, but also lighter to limit model complexity [12, 3]. For simplicity, our experiment uses the AlexNet architecture.

**Multi-Scale Features.** The best CNN models combine certain properties of images with the innovative deep architectures. One important property of images is the hierarchical multi-scale features. Even in classical CV studies, researchers used the spatial pyramid idea for natural scene detection [19]. This spatial pyramid idea is also often designed into CNN architectures in recent years, since the CNN models are good at identifying the high-level semantically meaningful features while may not be good at retaining the low-level and high-resolution features. This multi-scale feature representation can be achieved in different ways. Researchers designed the spatial pyramid pooling layer [13], the multi-scale pyramid pooling layer [33, 37], and dilated convolutional layer [34] to capture the multi-scale feature maps; designed CNN architectures with skip connections [23, 22, 14] to combine high- and low-resolution feature layers; designed separate CNN architectures for multiple-scale images with weight sharing to retain the scale-invariance property [8]; combined the high-level feature maps with conditional random fields to extract local information from CNNs' activation maps [9]. In our study, we will focus on the simplest approach of using Gaussian filters to generate the images with various scales and train the CNNs with the augmented image datasets.

**Representation Learning.** CNNs are both predictive and interpretable, since the CNN models can automatically detect semantically meaningful and generalizable features at the deeper layers of the architecture. For example, Zeiler and Fergus [35] found that higher level representations can disentangle gender and wearing glasses from the image of head portraits. Zhou et al. [38] found that the hidden layers in CNN can identify semantically meaningful objects in scene detection. The typical methods of interpreting the representations in CNNs involve case-based, neuron-based, or gradient-based methods. The case-based interpretation refers to those studies that use representative examples to explain the modeling results [10, 15]. Neuron-based interpretation refers to those that maximize neurons' activation or the whole activation space to identify the images in the input space [35, 25, 28]. Gradient-based methods, which can also be referred to as attribution and salience maps, focus on the gradients of the softmax scores regarding inputs [5, 1, 29]. In all these approaches, researchers rely on visualization to present their findings [38, 39].

## 3. Experiment Setup

### 3.1. Models

**Architecture.** Our CNN architectures are similar to the benchmark AlexNet [16] architecture, with convolutional layers followed by fully connected layers. The socio-economic information from the NHTS dataset is augmented by using a feedforward NN architecture to the CNN part in the last layer. The architecture is visualized in Figure 1.

We manually adjusted the hyperparameters based on a

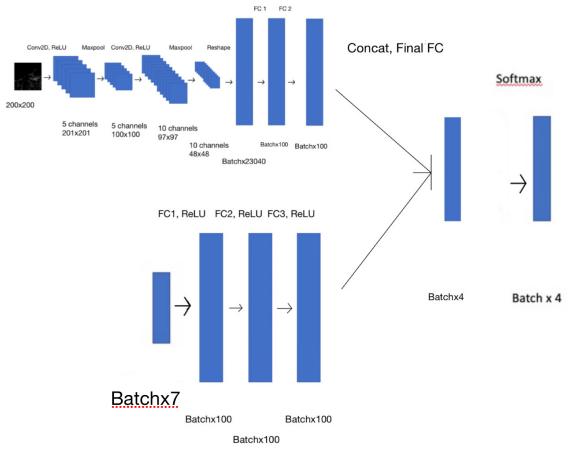



Figure 1: AlexNet + Feedforward Architecture

prespecified hyperparameter space as documented in Table 1. The bolded values are the final hyperparameters that we used to examine the predictive performance.

| Hyperparameters | Values |
|---|---|
| Depth | [**4 layers**, 6 Layers] |
| Fully Connected Width | [60, **80**, **100**] |
| Fully Connected Depth | [**1**, **2**, 3] |
| Optimizer | [SGD, **ADAM**] |
| Loss Function | **Cross Entropy** |
| Learning Rate | [0.1, 0.01, **0.001**] |
| Epochs | [20, 50, 100, **150**, 200] |
| BatchSize | [50, 100, **200**] |

Table 1: Hyperparameter Space

Although we experiemented with the basic stochastic gradient descent, we ended up using the Adam optimizer with a learning rate of 0.001. As for epochs, we found that 150 epochs were enough for the models to converge. As for the depth of the network, we tried two different structures. One includes includes 2 convolutional layers (each followed by ReLU and maxpool) and 2 fully connected layers, and the other uses 3 convolutional layers (each followed by ReLU and maxpool) and 3 fully connected layers. We found that the network with only 4 layers outperformed the network with 6 layers while also taking less time to train. For our networks augmented with the NHTS data set, we only used one fully connected layer in the CNN since that output was to be fed into a final fully connected layer with the feed forward network for the NHTS data.

Our network that uses only the NHTS data set is a simple feedforward neural network with 9 fully connected layers each with a ReLU layer in between that is all fed into a final softmax layer. The width of the final fully connected layer in this network is 80 units, but it is 100 units whenever the feed forward network is augmented.

To create the architecture that augments the NHTS dataset, we simply drop the last fully connected layer of our CNN and concatenate the output of the previous fully connected layer to output of our feed forward network, feed them through a final fully connected layer, and then a softmax layer. Note that we decreased the number of fully connected layers in the feed forward network to 3 for these combined networks.

**Multi-Scale Inputs.** To implement multi-scale feature pyramids in CNNs, we first reprocessed our data sets by downsizing each image twice — each after a Gaussian blur — which results in two new datasets. The images in those data sets are of size 100 x 100 and 50 x 50. We then create a new CNN of the same architecture as before, however, here it takes in 3 inputs in its forward pass: first the 200 x 200 images, then the 100 x 100 images, and finally the 50 x 50 images. We take the outputs from each forward pass and concatenate them, feeding them through a final fully connected layer and softmax activation. The convolutional layers used by each image is the same, allowing for weight sharing. As before, we augment the data from the NHTS dataset by dropping the last fully connected layer of our CNNs and concatenating the outputs of the previous fully connected layers to output of our feed forward network, feed them through a final fully connected layer, and then pass them through a softmax activation. We also tested the CNN model without weight sharing across the images of different scales.

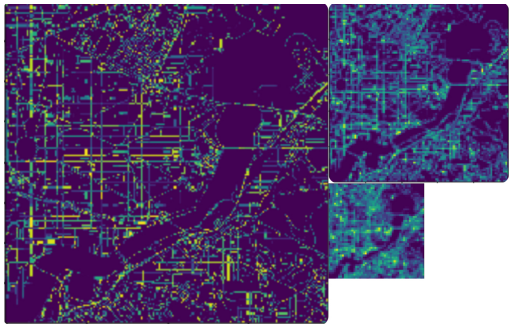


Figure 2: Multi-Scale Feature Inputs in CNNs

**Representation Learning Mechanisms.** We obtain the filters of our first few layers by just visualizing the weight arrays. We implement the activation map by extracting the outputs from our CNN model after the second convolution, ReLU, and max pool. We then normalize that output, convert it from a tensor to a numpy array, and resize it to the size of our training data. Now, we are able to invert the heatmap to obtain our feature map. Finally, we apply a color map jet using OpenCV to make the feature map more vis-

ible and combine it with our original image to display the activations.

### 3.2. Datasets

Our data sets include two sources: the public national household travel survey (NHTS) collected in 2017 and the urban satellite images augmented to the NHTS dataset. The NHTS dataset was a nation-wide survey conducted by the Federal Highway Administration (FHWA) every eight years, which happened in 2001, 2009, and 2017. In this survey, individuals reported their travel behavior (e.g. automobile ownership and activity patterns), socio-economic information (e.g. income and education background), built environment information (e.g. urban region and density), and geographical location information (e.g. census tracts). With the census tract information, we collected the satellite images of the census tracts where the HHs are living by using Open Street Map [2] and augmented the satellite images to the NHTS dataset. While the whole NHTS dataset has about 1 million observations, this course project only a sub-sample, which is about $4,000$ observations.

## 4. Results

### 4.1. Prediction Accuracy

Table 2 summarizes the predictive performance of all the CNN models. The first feedforward network that only uses NHTS dataset achieves a training accuracy of 64.57% and a testing accuracy of 62.09%. While the accuracies are not very high, this does mean that the features we chose from the dataset are correlated to auto ownership. Intuitively, an increase in house hold size, the number of workers in the household or the number of drivers in the house would also mean that the number of cars owned would increase since more people need cars to get to work. An increase in family income would mean that the household would be more likely to have the money to pay for more cars. If the house is owned, it would make sense that the household would have enough stability to own multiple cars. The metropolitan statistical area would be used for many different households, but the larger the area, the more likely it is that more cars are owned. Finally, urban areas require the use of cars much more than rural areas.

The CNN network that takes in only satellite images yields an training accuracy of 76.74% and testing accuracy of 73.90%. Our first observation is that this accuracy already proves to be much better than the accuracies from only using the NHTS dataset, which means that there are many features hidden within the images that predict auto ownership much better than the features we used from the NHTS dataset. Since these machine learning models are hard to interpret from the outside, we will expand more about this under section 4.3 Representation Learning.

The network that augments NHTS data with satellite images shows significant improvements, and it achieves a training accuracy of 83.27% and a testing accuracy of 82.11%. It makes sense that this model performs the best, since it has the most information — both from the NHTS dataset and the satellite images. With the final fully connected layer at the end, this model is also able to learn and utilize the features from NHTS data and satellite images in conjunction.

### 4.2. Multi-Scale Feature Pyramid

The network that uses satellite images with image pyramids and weight sharing yields us a training accuracy of 75.59% and testing accuracy of 73.00%. Since this accuracy is a bit lower than the CNN that does not use image pyramids, it is likely that our method of creating the image pyramids is not complex enough. The small difference in accuracy would mean that the smaller images — from the Gaussian blur and down sample — do not contain any new information that the original image does not have.

The multi-scale feature pyramid network with weight sharing augmented with NHTS data achieves a training accuracy of 83.18% and a testing accuracy of 80.54%. Again, the testing accuracies are not far from the accuracies without image pyramids, so it is likely that our smaller images do not contain new information that the original image does not have.

We also implement our multi-scale feature pyramids networks without weight sharing. The accuracies of those networks can be found in Table 2 under the last column. The accuracies are much worse than what we get with weight sharing. This further solidifies the idea that our Gaussian pyramid images do not contain any useful information since now that the weights are no longer shared, the network is more reliant on the smaller Gaussian pyramid images, which do not contain useful information. It is even possible that these smaller images contain useless information that simply decreases the accuracies of our models. Nonetheless, it is also possible that our network is not deep/complex enough to capture all the information from the image pyramids.

### 4.3. Representation Learning

Since our GPU and training algorithm limits the number of channels outputted from the first layer, we are unable to obtain simple filters like edge detectors. As a result, the filters shown in Figure 3 are much less generic. However, these filters do match up with the filters found in most networks' second layer. These filters find image patches that activate each of the layer 1 and 2 neurons most strongly. In our case, these filters find traffic lines and buildings. This

[2] https://www.openstreetmap.org/map=5/38.007/-95.844

| Data | Train Acc | Train Loss | Test Acc | Test Loss | Test Acc (With MSF and WS) | Test Acc (With MSF but no WS) |
|---|---|---|---|---|---|---|
| NHTS | 64.57% | 1.11 | 62.09% | 1.12 | N/A | N/A |
| Satellite Images | 76.74% | 0.98 | 73.90% | 1.00 | 73.00% | 72.33% |
| NHTS + Satellite Images | 83.26% | 0.91 | 82.11% | 0.92 | 80.54% | 74.35% |

Table 2: Model Performance of CNNs. (MSF: Multi-Scale Features. WS: Weight Sharing. Convergence of all the models can be found in Appendix I.)

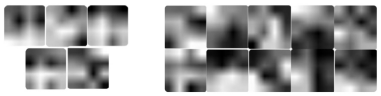


Figure 3: Layer 1 and Layer 2 Filters

intuitively makes sense as the areas with buildings indicate that there are more people who will likely need to drive. The traffic lines indicate areas for driving, and so if they exist, then there will be much more cars.

If we take a look at the activation maps shown in Figure 4, we can see that the areas with traffic lines and buildings are highlighted, and this makes sense given the filters shown in Figure 3. The areas without roads have no highlight since the people there are unlikely to have cars since there are no roads to drive on. Areas with buildings are likely to contain people, and thus, it is more likely to have more automobiles as people need to drive around.

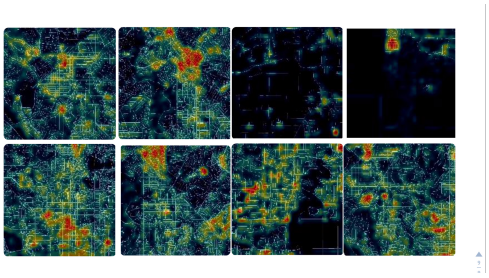


Figure 4: Activation Maps

## 5. Conclusion

In contrast to the classical method of handcrafting built environment features such as connectivity and accessibility, this study examines how to predict automobile ownership by directly using satellite images. The experiments demonstrate the power of deep learning. A simple CNN with only two layers can achieve 11.9 % higher prediction accuracy than the predictive power of a model that uses the most important variables in the NHTS dataset. The CNN model is also interpretable, since the lower level convolutional filters are more similar to classical spatial filters and the high level activation maps have captured the geographical regions that have denser buildings and roads. Considering the contrast between the tremendous monetary cost (probably millions of dollars) of collecting NHTS dataset and the cheapness (four weeks of one student's work) of collecting satellite images, this CNN approach provides an viable and efficient alternative for understanding the travel behavior in the urban settings than the classical methods.

It is intriguing to consider why the multi-scale feature structure does not improve the model performance. It can be caused by the fact that our image pyramid is constructed at the stage of data preprocessing, which is not an ideal way to incorporate the multi-scale information. More advanced methods, such as creating the skip connections [33], dilated convolutional layers [34], auto-encoder architectures [22], or the pyramid pooling layers [13], could potentially improve the model performance. To further interpret the CNN models, we can also adopt more advanced methods, such as using activation maximization [35, 10] or visualizing the attribution maps, to understand the semantic meanings of the neurons and the sensitivity of the CNN system. We will explore these research directions in the future.

## Author Contributions

S.W. conceived of the presented idea; S.W. preprocessed the data and provided the draft codes for empirical experiments; J.L. conducted the experiments for prediction, multi-scale features, and representation learning. S.W. wrote the sections about abstract, introduction, related studies, and conclusion; J.L. wrote the sections about experimental setup and results.

## References


[1] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. MÃžller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(Jun):1803–1831, 2010.

[2] D. Baldwin Hess and P. Ong. Traditional neighborhoods and automobile ownership. *Transportation Research Record:*

*Journal of the Transportation Research Board*, (1805):35–44, 2002.

[3] P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[4] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[5] Y. Bentz and D. Merunka. Neural networks and the multinomial logit for brand choice modelling: a hybrid approach. *Journal of Forecasting*, 19(3):177–200, 2000.

[6] C. R. Bhat and J. Y. Guo. A comprehensive analysis of built environment characteristics on household residential choice and auto ownership levels. *Transportation Research Part B: Methodological*, 41(5):506–526, 2007.

[7] R. Cervero and J. Day. Suburbanization and transit-oriented development in china. *Transport Policy*, 15(5):315–323, 2008.

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2016.

[9] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

[10] D. Erhan, Y. Bengio, A. Courville, and P. Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

[11] R. Ewing and R. Cervero. Travel and the built environment: a meta-analysis. *Journal of the American planning association*, 76(3):265–294, 2010.

[12] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] B. Kim, C. Rudin, and J. A. Shah. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in Neural Information Processing Systems*, pages 1952–1960, 2014.

[16] 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.

[17] G. J. Kwoka, E. E. Boschmann, and A. R. Goetz. The impact of transit station areas on the travel behaviors of workers in denver, colorado. *Transportation Research Part A: Policy and Practice*, 80:277–287, 2015.

[18] C. Lave and L. Lave. Fuel economy and auto safety regulation: Is the cure worse than the disease? In *Essays in Transportation Economics and Policy: A Handbook in Honor of John R. Meyer*, pages 257 – 291. 1999.

[19] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178. IEEE.

[20] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[21] Q. Liao and T. Poggio. When is handcrafting not a curse? Technical report, 2018.

[22] T.-Y. Lin, P. Dollr, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[23] M. Long and J. Wang. Learning multiple tasks with deep relationship networks. *arXiv preprint arXiv:1506.02117*, 2, 2015.

[24] D. McFadden. Conditional logit analysis of qualitative choice behavior. 1974.

[25] G. Montavon, W. Samek, and K.-R. Muller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.

[26] D. Pickrell. Transportation and land use. In *Essays in Transportation Economics and Policy*. 1999.

[27] P. Schimek. Household motor vehicle ownership and use: how much does residential density matter? *Transportation Research Record: Journal of the Transportation Research Board*, (1552):120–125, 1996.

[28] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[29] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org, 2017.

[30] P. G. Swartz and P. C. Zegras. Strategically robust urban planning? a demonstration of concept. *Environment and Planning B: Planning and Design*, 40(5):829–845, 2013.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. Cvpr, 2015.

[32] M. Wachs. Transportation policy, poverty, and sustainability: history and future. *Transportation Research Record: Journal of the Transportation Research Board*, (2163):5–12, 2010.

[33] D. Yoo, S. Park, J.-Y. Lee, and I. So Kweon. Multi-scale pyramid pooling for deep convolutional representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 71–80, 2015.

[34] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[35] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

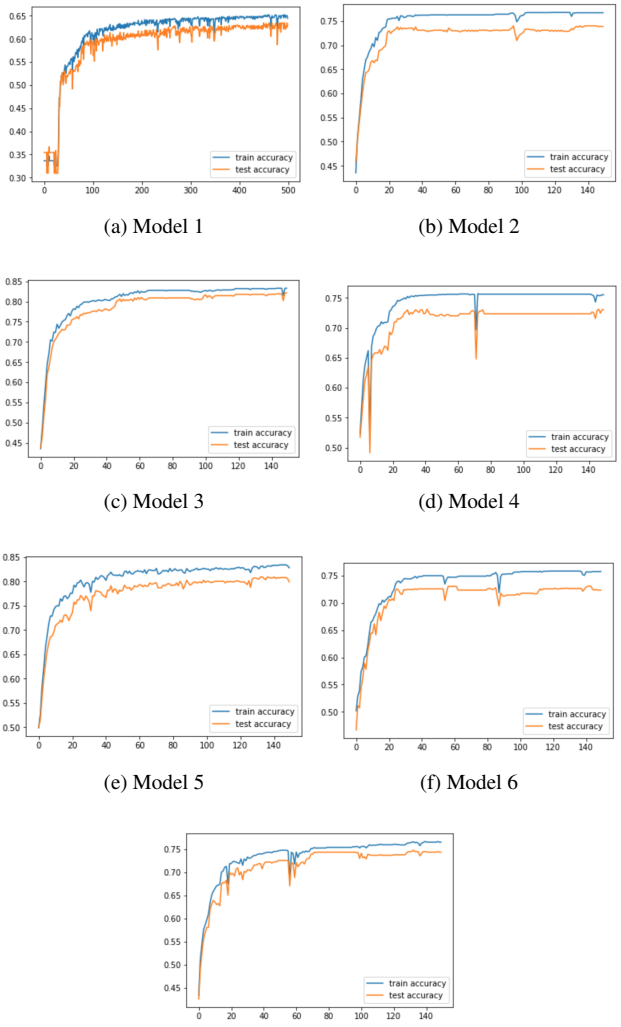
[36] M. Zhang. The role of land use in travel mode choice: evidence from boston and hong kong. *Journal of the American planning association*, 70(3):344–360, 2004.

[37] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[38] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014.

[39] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 2921–2929. IEEE, 2016.

## Appendix I. Convergence of CNN Models




(a) Model 1 (b) Model 2

(c) Model 3 (d) Model 4

(e) Model 5 (f) Model 6

(g) Model 7

Figure 5: Convergence of CNN Models; Model 1: NHTS dataset; Model 2: only satellite images; Model 3: NHTS + satellite images; Model 4: satellite images with pyramid and shared weights; Model 5: NHTS + satellite images with pyramid and shared weights; Model 6: satellite images with pyramid and no shared weights; Model 7: NHTS + satellite images with pyramid and no shared weights;