Improving Housing Price Prediction through Image-based Submarket Segmentation

submitted in partial fulfillment for the degree of master of science

Hendrik Matthias Tilman König
12261092

master information studies
data science
faculty of science
university of amsterdam

2019-07-05

	Internal Supervisor	External Supervisor
Title, Name	Dr. Andrew Brown	Maurice Hoenderdos MSc
Affiliation	UvA, FNWI, IvI	PricewaterhouseCoopers Advisory N.V.
Email	a.g.brown@uva.nl	maurice.hoenderdos@pwc.com

Universiteit van Amsterdam

Amsterdam
Data Science

ABSTRACT

The value of a real estate property does not only depend on its tangible characteristics, but also on its intangible and contextual assets, most importantly its location. Hence, two identical houses may differ in price if located in different regions, making locational information a crucial component of housing price estimation models. This study aims at combining both dimensions by taking into account both the tangible assets as well as context information of a property. Specifically, it investigates into the effects of submarket segmentation and most notably presents a novel way for replacing manually defined submarket variables with locational information gathered from satellite imagery. These spatial features are learnt by a Deep Convolutional Neural Network and combined with the known assets of the house in order to produce the price estimation. Moreover, we find that using these features in a price estimation model reduces the prediction error by approximately 26% and is as effective as using pre-defined submarkets.

KEYWORDS

Housing Price Prediction, Satellite Images, Convolutional Neural Networks, Transfer Learning

1 INTRODUCTION

Estimating the value of a real estate property is a complex problem and has been studied by econometricians for decades. While early methods involved using traditional linear regression models, they were limited by spatial dependence among properties [3]. This means that house prices in a certain region are affected by those of surrounding properties, which most likely share some characteristics not covered by the sample data.

To better understand this problem, it might be helpful to consider an example from another domain. Let us assume we want to predict a person's body height, using a data sample collected from multiple places across the world. This sample might contain variables such as age, weight, gender and other physical characteristics of an individual and we could now fit a regression model on these predictors to estimate how tall this person is. This method would however be flawed, as a person from Spain might have the exact same age, gender, etc. as a person from the Netherlands, but their height might still differ.[1] Even though there might be variables accounting for this difference, such as for instance the average daily cheese consumption, these relations might not be known or not covered by our data. In any case, we would end up with biased estimations.

If we translate this problem back to housing price predictions, it becomes clear why we must account for inherent differences between regions or, in urban terminology, neighborhoods. Known methods for tackling this problem are incorporating neighborhood relationships, also referred to as Spatial Auto-Regression, or segmenting the sample into submarkets using manually defined neighborhood variables. [6][24].

In this paper we focus on the latter, but use a novel basis for segmentation, namely satellite imagery. Related work done by [1][5][4][41] and [23] has shown that satellite as well as street-level imagery can provide information relevant to housing price estimation. However, these studies are following the assumption that enriching housing data with additional price-related features, in this case found in images, sufficiently accounts for the contextual variation as previously explained.

The present work is placed within a different paradigm, as we assume that inherent price differences among neighborhoods can be addressed by defining homogeneous submarkets. We test this assumption by comparing different price prediction models using (i) house-level features only, (ii) house-level features combined with manually defined neighborhood variables and (iii) house-level features combined with image features extracted from a Deep Convolutional Neural Network model trained on classifying satellite images into different urban areas. We further compare this method to other approaches, in which image features have been extracted from equivalent models trained on housing prices directly. In that context, we also experiment on and optimize previously proposed model architectures. Moreover, we make the following contributions:

- We demonstrate that using features learnt from satellite images has a positive effect on real estate price estimation and is comparable to submarket segmentation based on manually defined neighborhood variables.
- We present that using image features extracted from a Convolutional Neural Network neighborhood classification model outperforms using features from a model directly trained on housing prices.
- We show that using image feature representations extracted from a generic convolutional layer of a pre-trained Convolutional Neural Network model leads to superior results to using feature representations from re-learnt layers.

[1] In fact, the average male body height is 176 cm in Spain and 183 cm in the Netherlands, according to [12].

- We investigate into the learning behavior of the model and find no evidence for its ability to pick up specific features such as points of interests, but rather to learn spatial neighborhood structures.

2 RELATED WORK

Traditional models for housing price estimation

Housing prices are traditionally modeled using hedonic regression models [25][6]. This approach considers housing a heterogenous good which can be broken down into its utility-bearing attributes or characteristics [36], where the prices of these characteristics are referred to as the hedonic prices. Accordingly, the price of a house is determined by collections of different attributes and their hedonic value. In the case of housing, such hedonic characteristics could be property attributes such as age or number of rooms, or locational attributes such as access to public transport. In theory, these hedonic values can be computed by differentiating the hedonic price function with respect to each attribute [26]. In other words, the implicit value of an attribute and its influence on house prices can be estimated by observing property values while holding remaining attributes constant. This method has been applied to various amenities and their impact on real estate value, e.g. neighboring shopping centers [13], urban green space [20], cultural amenities [11], quietness [26] or air pollution [35].

Geospatial data enrichment

When estimating housing prices, the observations are collected from points or regions located in space. Especially if sample data is collected in a relatively dense grid, it must be assumed that the observations are not independent or, in other words, that independence of variables is violated. Intuitively, this means that house prices in a certain region are affected by those in an adjunct neighborhood, arguably because nearby properties will often share structural features or amenities, which in most cases are not covered by the sample data [3].

In regression models, such an independence violation can lead to inefficient parameters and biased estimations [24]. This problem has led to the creation of advanced techniques, where models are trying to capture locational characteristics by either incorporating neighborhood relationships or segmenting the sample into submarkets using one-hot encoded neighborhood variables [6][24]. The former assumes shared latent attributes between properties in neighboring regions, whereas the latter relies on the idea of similarity in the prices of housing characteristics within a submarket. [28] argue that clustering or segmenting the dataset can minimize problems associated with heteroscedasticity. Beyond that, [6] find that when adding valuer-defined submarket variables to an OLS model, housing price predictions are more accurate than predictions made by geostatistical models. While submarkets can be defined by valuer assessment, they could also be defined based on streets [22].

Generally, both forms of geospatial enrichment require special attention to be paid to specifying neighborhoods or submarkets, respectively. That is, if the specified neighborhood structure is not representative, various spatial features will not be discerned [31]. Likewise, choosing submarkets that are not homogeneous could lead to inaccurate predictions [6]. These manual design choices present one of the limitations of spatial econometric models.

Nonlinear models

The previously described models share further, inherent limitations, as they all rely on multiple regression analysis (MRA). Methodological problems associated with MRA have been known for some time, such as its inability to adequately deal with nonlinearity. Given the highly complex dynamics of the housing market, MRA models easily fail to grasp the information of the price formation process [9]. Another source of error is function form misspecification, referring to models that suffer from not properly accounting for the relationship between the dependent and observed explanatory variables, e.g. due to wrong assumptions about functional relationships between selling price and housing attributes [43].

Given the restrictions of MRA models, it has been proposed to adapt multiple layered artificial neural network (ANN) models in housing price prediction [14][29][33]. In contrast to MRA models, these are capable of modeling complex nonlinearities from highly heterogeneous data. However, views on the performance of ANNs are conflicting. For instance, [39] concluded that neural networks were not superior to multiple regression analysis. It should however be noted that their study is based on a relatively small sample consisting of 288 real estate transactions, which might explain the disability of their model to generalize from the data. This is supported by [29] as well as reflected in the work by [33], both supporting the conclusion that ANN performs better than MRA only when a moderate to large sample size is used, especially with increasing model complexity. Lastly, [43] add that ANNs perform particularly well with heterogeneous datasets.

Extracting locational information from image data

In general, all the methods described so far benefit from including spatial information into modeling. That is, they all show higher predictive power when being enriched with information about neighboring houses or when segmented into spatial submarkets having common characteristics. The former has led to novel approaches for enriching housing data in order to improve value estimations, namely by making

use of street-level or satellite imagery as source of locational information about a house or a neighborhood.

Street view imagery. Collecting locational information from street-level imagery has initially been applied outside the field of real estate valuation, with the aim of quantifying visual urban elements [15][2][34]. For instance, [34] present a study in which they asked participants to assess street images from different London neighborhoods as beautiful, quiet or happy. They then extract visual features from these annotated images, such as colors, textures and a number of "visual words" and find for example that the color green positively correlates with all three attributes, possibly because green image components represent the presence of nature in a street scene, while gray and dark red correlate negatively. They furthermore find that quiet scenes are also quiet or smooth in the visual sense and moreover point out interesting visual patterns related to the positive or negative perception of urban scenes, which might possibly translate to satellite images as well and/or relate to housing prices.

[15] and [2] use discriminative clustering to identify visual elements in street view images that are characteristic for the location or specific city attributes, such as price or safety. In the latter study, it is shown that there is a relationship between visual elements and housing prices, that is, they find that in San Francisco visual elements corresponding to hedges, gable roofs and tropical plants indicate high real estate values. This finding demonstrates the relation between housing prices and visual elements and might also be applicable to satellite images. It should be noted that these studies do not utilize machine learning methods such as Deep Convolutional Neural Networks to learn image features, but instead those features are image patches represented as HOGs (Histogram of Oriented Gradients). Even though [2] briefly experiment with using features from a Convolutional Neural Network (CNN), they do not carry out further optimization or in-depth analysis of the results even though they achieve higher performance when using them. In that context, it can be assumed that CNNs are able to better capture high-level aspects of visual appearance, arguably at the cost of model interpretability.

Lastly, [5] propose a model specifically aimed at predicting housing prices by feeding latent image features extracted from street view images by a CNN into different regressors, together with house attributes, such as size or age. This data enrichment with visual information enhances the predictions of their housing data-only base model by 4%. In view of this, it stands to reason that street view images do not provide much additional information relevant to house price prediction.

Satellite imagery. While the studies presented earlier gather information from street-level imagery, it is likely that satellite imagery can provide further locational information when predicting a city attribute value, especially since it might be able to better account for structural differences between different areas and/or spatial dependency. This notion was taken up by [4], who use satellite images to predict real estate values in London, Birmingham and Liverpool, thereby outperforming the traditional SAR method. In their approach, they use transfer learning to train a CNN classifier to distinguish between high and low priced areas. The image features learnt by this model are then extracted, fused with house-level features and fed into different regressors, with an ANN yielding the best performance. Beyond that, their approach outperforms the model proposed by [5] on the same housing sales dataset, but using satellite instead of street view imagery.

Most recently, [23] extended this approach by merging features from street view, satellite and property data into one price prediction model. Counter-intuitively, the joint model using all data sources shows a nearly similar performance to the one including either housing or satellite image data only. Besides, the model merging sales data with features from satellite imagery shows superior performance over the one with features from street-level image data.

Altogether, these results suggest that satellite neighborhood images can potentially improve housing price predictions. In the following section, limitations of previous research will be discussed in greater detail, while outlining the contributions of this thesis.

Extending previous research

This thesis will adapt the data enrichment techniques as proposed by [4] or [23] and integrate visual cues from satellite images into the prediction model using Deep Convolutional Neural Networks, as those have evidently demonstrated their ability to learn meaningful features from image data. While borrowing some parts of their methods, we however use a refined methodology and focus on a more specific aspect of housing price modeling. As pointed out earlier, sample segmentation based on one-hot encoded neighborhood variables can improve housing price prediction [28][6][24], while defining these neighborhoods is usually based on manual choices. At the same time, several pitfalls need to be considered, namely that (i) various spatial features might not be discerned [31] or (ii) the resulting submarkets might not be as homogeneous as desired [6] in case the defined neighborhood structure is not representative.

Against that background, this thesis does not purely focus on identifying a suitable model architecture and image feature representation, but also empirically investigates whether

satellite imagery can account for inherent differences between neighborhoods and thereby replace or even outperform manual submarket segmentation.

3 DATA

Real Estate Sales Data

Data is acquired from the public NYC Geodatabase providing geocoded data on real estate sales in New York City from the year 2015 [16]. The dataset is based on annual sales reports by the New York City Department of Finance and all records are provided with longitude and latitude coordinates using the NAD83 geodetic datum as their reference point. Next to the location, the data provides information on the properties as well as the selling price. In total, the dataset contains 30 attributes including information on both sale and property level. However, some of those attribute columns were found to be uninformative or to contain a large quantity of missing values and were removed, resulting in 10 variables used as the housing data feature representation. These features are: Property age, usability, category, number of residential and commercial units, total number of units as well as land and property surface in square feet. Categorical variables are one-hot encoded before modeling.

As some houses were labelled with implausibly low selling prices, we account for this by thresholding for price and only keeping properties sold for more than $100,000, while dropping 29,370 records with an average price of $7,933. Further outliers are removed by mapping the selling prices onto a distribution with mean 0 and standard deviation 1 and excluding all samples deviating more than one standard deviation from the mean. Due to the highly right-skewed distribution of prices, this approach can be compared to setting an upper bound and discarding houses with extremely high price values.

Furthermore, duplicate entries for the same properties sold multiple times were dropped, while only keeping the most recent ones. After cleaning, the final dataset contains 54,204 records with mean selling price $1,040,051 and standard deviation $1,448,341, where the latter is most likely influenced by the highly diverse nature of the included properties and reflected in the distribution of houses with respect to selling price of the final dataset as visualized in Figure 1.

Satellite Image Data

Satellite imagery is provided by the New York City Department of Information Technology & Telecommunications [19]. These images are captured every two years during spring and summer months and cover the entire state of New York. They are furthermore provided in most recent form, which means they are from 2018 in our case. Before being published, the images have been corrected to remove distortions caused by elevation changes and camera angles. Images are provided as four-band raster files in the .jp2 format and contain meta-information such as the geo-location of each image pixel, which is in a similarly projected coordinate system to the real estate sales data.

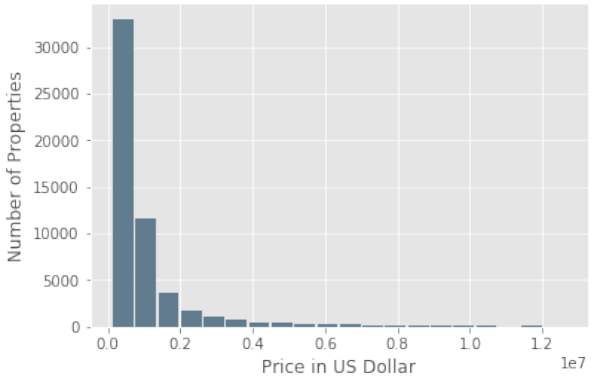


Figure 1: Sample distribution of housing prices.

In order to generate a set of satellite images matching the locations of the sold houses, the image files are first merged into a three-band (RGB) raster dataset. Then, for each house an image is being cut out from the mosaic, with the house at the center point and surroundings within a 500ft radius. The resulting images are stored as colored .jpg files, with size set to 2000x2000.

4 METHODOLOGY

Estimators

Given their prominence in existing literature, and the properties of the sample data at hand, nonlinear estimators are used for predicting the housing prices [9][43]. More specifically, we choose Multilayer Perceptron (MLP) and Random Forest (RF) regressors. Both are types of machine learning algorithms and will be briefly explained in the following section.

Multilayer Perceptron. In basic terms, the MLP consists of a layered set of interconnected neurons. The neurons, or nodes, are connected by weighted output signals. These output signals can be seen as a linear function of the output from all previous nodes. While stacking those functions would simply result in a linear mapping, inputs are made nonlinear by applying a transfer function, such as the logistic or more commonly used ReLu function, before pushing them through the next layer. This enables the MLP to approximate highly nonlinear data, even with only one hidden layer [27][18]. The MLP learns from labelled data containing a possibly large number of input and corresponding output vectors. Weights

are adjusted by repeatedly presenting the network with new samples from the training data, calculating the error and correcting the weights of each node via gradient optimization [37]. This iterative training process can be performed until the desired accuracy has been achieved, or until it does not lead to further improvement.

Before training MLP models, we bring our data on approximately the same scale. For this, we remove the median and scale the data according to the Interquartile Range, thereby accounting for the influence of outliers in our sample.

Random Forest. As mentioned earlier, Random Forest regressors are another machine learning algorithm capable of creating a nonlinear mapping between an input and output vector [7] and have also been successfully applied to real estate price estimation [8]. The main principle of the RF is to create an ensemble of learners which are trained by means of bootstrapping from the original dataset. That is, each learner represents an individual regression tree grown on a separate bootstrap sample. Regression trees are built top-down from a root node corresponding to the best predictor variable in terms of variance reduction. Following this principle, they further break down the dataset into smaller subsets whereas each decision node represents a binary test against the selected variable. Final leaf nodes amount to the numerical target, that is the normalized sum of output values for all instances within the resulting subset. In a RF, the prediction is then made by averaging over the outputs of all individual trees or learners.

Hyperparameter Settings. For the MLP, we use 1 hidden layer with 10 nodes. The RF uses 10 estimators and all features for each split and terminates when less than two instances remain in the branch. These hyperparameters are held constant and are not further tuned, as the aim of this thesis is not to minimize the absolute error for each model, but to test for the effect of adding neighborhood variables or image features to the model.

Housing Price Estimation

We first fit the regressors on the real estate sales data while holding out locational information. According to the literature, performance should improve when incorporating those into the model. In theory, these variables should moderate for structural or implicit differences between neighborhoods and capture the effects of these differences on housing prices. Here, locational information is provided in the form of fine-grained neighborhood variables as defined by the New York Department of Finance, segmenting the housing sales data into 255 submarkets. We therefore fit a second model onto the dataset including neighborhood information and use both models as baselines against those using image features. The process of extracting these features will be addressed in the following section.

Image Feature Extraction

We chose two pre-trained, state-of-the-art convolutional neural network (CNN) models to gather information from the satellite imagery. These are Inception-V3 [38] and ResNet-50 [17]. The following sections will provide an overview of the general functionality of CNNs, the selected models specifically and the exact method used for extracting the image features.

Convolutional Neural Networks. A CNN is a type of artificial neural network that is able to learn representations of the spatial structure of multi-channel images. Their very large number of parameters must be learnt from training examples, comparable to the learning procedure as understood in light of MLPs. Historically, the absence of both sufficient amounts of training data and readily available computational power have presented bottlenecks in creating models capable of complex tasks such as generic object recognition. This has however changed with the rise of GPUs and the introduction of a deep convolutional neural network able to classify the 1.2 million high-resolution images in the ILSVRC-2012 contest while achieving superior performance over all previous methods [21].

As the name suggests, CNNs perform convolutional operations. This means, instead of learning separate weights per input feature, it learns shared weights over a local region. Given an image as an input matrix, the network therefore keeps its spatial information by not considering each pixel separately, but by taking its spatial neighborhood into account. The convolution, which is illustrated in Figure 2, is performed by filters that move over the input image while repeating the operation for each pixel location. Those filters can be considered feature detectors learnt by the network, each performing a different operation on the input image, and extracting different kinds of image features. Thereby, a set of feature maps is created in each convolutional network layer, as seen in Figure 3, with their output being maximized when corresponding structures are seen in the input.

As the network gets deeper, activations are made nonlinear by applying transfer functions and output maps are downsized using pooling techniques before they are flattened into a feature vector and fed into fully-connected layers. These layers then map the learnt features to the desired output class, that is in case of a classification task.

ResNet-50. The ResNet-50 is a CNN model from the family of Residual Networks, which won the 2015 ImageNet LSVRC competition [17]. Prior to their introduction, increasing network depth has frequently led to training issues and resulted

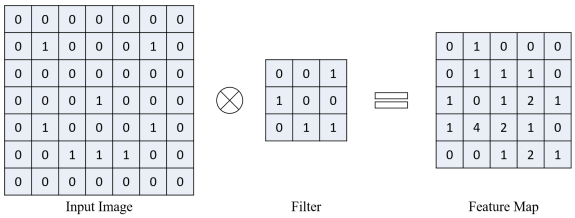


Figure 2: Example of a convolutional operation.

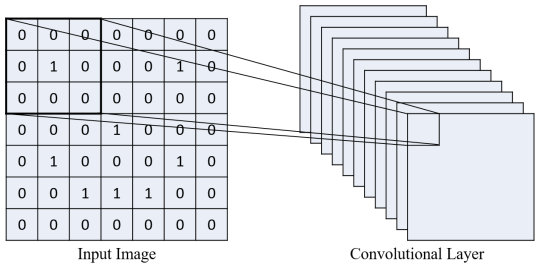


Figure 3: Illustration of a convolutional layer.

in performance drops as well as high computational expenses. Residual Networks use skip connections, where the original input to a stack of layers is added to its output, before being passed through a nonlinear transfer function. Hence, the output of early layers is largely preserved throughout the network, while it learns a residual mapping between the inputs and outputs. Furthermore, it allows the network to skip blocks of layers as weights gravitate towards zero, making their output matching the input and allowing the network to analogously skip those blocks during back-propagation. This, along with dimensionality down- and up-sampling before and after each residual block, leads to reduced computational costs and arguably more efficient network training. ResNet-50 is 50 layers deep and has 23,867,786 parameters.

Inception-V3. The first runner-up in the 2015 ImageNet LSVRC competition, Inception-V3, presents another model architecture that tackles training problems and the large computational demand of deep CNNs [38]. The main idea behind the Inception architecture is based on the assumption that most activations in a dense connection model architecture (i.e. where every output channel is connected every input channel) are dispensable, for instance because they are equal to zero. Inception networks therefore use modules that together approximate a sparse architecture while being densely constructed, as they are made out of convolutional building blocks. These modules first extract spatial information at various scales and then aggregate the outputs of each filter into one layer that serves as input for the next module. As this stacking would tremendously increase the number of parameters, 1x1 convolutions are applied to reduce channel dimensionality of the input layer before performing expensive 3x3 or 5x5 convolutions. This way the network can grow in depth and width without running into computational bottlenecks. Inception-V3 is 311 layers deep and has 23,885,392 parameters.

Transfer Learning. Given the large number of parameters learnt by these models and the relatively small size of the dataset at hand, we do not train them from scratch, but apply transfer learning, where image representations are learnt by CNNs from large-scale datasets such as Imagenet [30]. These descriptors were shown to have sufficient representational power to be used in different domains and even for fine-grained object classification tasks, where distinctive class features might be subtle and hard to pick up.

Considering our problem as a fine-grained classification task, we extract image representations from pre-trained ResNet-50 and Inception-V3 models and use those to train classifiers on different tasks within our target domain. However, as [40] have empirically demonstrated, transferability of image features strongly depends on the layer from which they are extracted and on whether the network has been fine-tuned. Most importantly, they have shown that image representations tend to become more specific towards the final network layers. Thus, the less target and source domain are related, the more the representational power of the final layers diminishes. This problem can be worked against by allowing layers to re-learn, thereby adapting to the target domain.

On that basis, we test for different image representations as well as training modes. Overall, the performance of the classifiers we build is assumed to indicate the quality or meaningfulness of the image features, as strongly discriminative class features should embed information somewhat meaningful to the price estimation model.

Training Procedures

We train models on price as well as neighborhood classification tasks. For the price classification, the model is trained on distinguishing between two classes, which intuitively can be described as cheap and expensive houses. That is, we bin the sales data into ten equally sized price regions and keep the top and bottom 10%, after finding that a 10-class classification model does not achieve sufficient performance. For neighborhood classification, we predict which borough of New York City the image belongs to. In this setup, boroughs act as proxies for neighborhoods as they require the network to pick up spatial characteristics, while containing a large enough number of training examples per class.

It should be noted that previous work only involved CNN models trained on price prediction or classification tasks [4][23]. This being the case, it can be considered a benchmark method which we test against by using models that are supposed to explicitly learn spatial neighborhood information instead.

We distinguish between three training modes. In the first mode, all models are initialized with locked parameters and their final layer removed. Instead, the output from the second to last layer is globally averaged and fed into an additional set of fully-connected layers, followed by a classification layer. The fully-connected layers both contain 128 nodes and are retrained over 5 epochs and $n_{samples}/batchsize$ iterations, where batch size is set to 32. In order to account for eventual learning biases and to increase the generalization ability of the model, we augment our training data by randomly rotating, shifting and horizontally flipping each image in the training set. Furthermore, we normalize each image and set input size to 299x299 and 224x224 in the RGB color space for Inception-V3 and ResNet-50, respectively.

In the second training mode, we test for transferability of features from different convolutional layers, as we want to find out whether generic image descriptors from earlier layers are more effective than more specific feature representations extracted from final layers. To that end, we keep the model parameters locked and move backwards through the network, while selecting different convolutional layers and feeding their output directly into a set of additional layers as described before.

In the third training mode, we let the model re-learn parameters by unfreezing parts of the network before training it. Again, we move backwards through the network such that we begin with unfreezing the final layers or modules, as their feature representations arguably tend to be the most specific ones and thus require to be adapted to our target domain. Moreover, we use the same hyperparameters and image augmentation techniques in all training modes.

To reduce computational expenses, we perform all experiments on a subset of the data, using only records from the boroughs Manhattan, Brooklyn and Staten Island (n=15132, 13959, 4707, respectively). While maintaining the proportion of class labels, data is randomly split into training and testing sets, such that we use 85% of our data for training and keep the remaining part for final testing. Training data is split again, cutting another 20% off for validation during the training procedure.

For both tasks, we aim at achieving a test accuracy of at least 80%, as further optimization efforts would exceed the scope of this thesis. Lastly, feature vectors from both the last convolutional layer and the first fully-connected layer of the final models are merged with the housing data.

Implementation Details

All implementations are made in Python 3.7 using the Scikit-Learn package [32] for the regression models and Keras [10] with a Tensorflow backend for the image feature extractors. We furthermore use a 98GB RAM CPU-node in one of the Distributed ASCI Supercomputer 4 (DAS-4) clusters for the image collection, as creating the mosaic led to memory issues when tried on a local machine with a 16 GB RAM CPU. On the other hand, CNN training is performed on a local machine, equipped with a GeForce GTX 1070 GPU with 8GB memory.

5 RESULTS

Metrics

Model performance is measured by the Mean Absolute Error (MAE) they produce. The MAE represents the sum of absolute values of the residuals divided by the number of datapoints and thus provides a natural measure of average estimation error. Moreover, it was found suitable for the purpose of this research, which is an evaluation and intercomparison of different models and their performance errors. For each model or experiment, we perform three-fold cross-validation and report the mean produced error. We construct 95% confidence intervals based on the standard deviation of the errors and use these intervals to determine statistical significance.

Baseline

As a baseline, we calculate the MAE using the mean price of all houses as a fixed prediction value and end up with a MAE of $792k. By comparing our results against the error of this random prediction, we can assess the predictive power of the models we use and ensure that they learn meaningful information from the sample data.

Results from CNN Classification Models

Image Classification using pre-trained models. We first initialize pre-trained models with fully locked parameters and use the output from their second to last layer to retrain a set of additional layers on both the price bin and the borough classification tasks. Test set performance is reported for each model in the form of Accuracy, Precision and Recall and presented in Table 4-a. While Inception-V3 achieves an accuracy of 70% on the price bin classification task, ResNet-50 assigns nearly every image to the same class, which results in 50% accuracy. It should be noted that training accuracy is already more than 90% after the first epoch, which indicates that the model is highly overfitting.

A similar beavior is found for the borough classification task. Again, we find that ResNet-50 overfits while Inception-V3 yields an accuracy of 67%. Detailed results from this experiment can be found in Table 4-b. In the light of these

Model	Accuracy	Precision	Recall
ResNet-50	0.49	0.24	0.49
Inception-V3	**0.70**	**0.76**	**0.70**

(a) Performance scores for price bin classification task, using fully locked pre-trained models.

Model	Accuracy	Precision	Recall
ResNet-50	0.45	0.22	0.45
Inception-V3	**0.67**	**0.70**	**0.67**

(b) Performance scores for borough classification task, using fully locked pre-trained models.

Model	Accuracy	Precision	Recall
Inception-V3	0.83	0.84	0.83

(c) Model performance for borough classification task, using feature representation from first convolutional layer.

Model	Accuracy	Precision	Recall
Inception-V3	0.80	0.82	0.80

(d) Model performance for price bin classification task, with final Inception module retrained.

Figure 4: Experimental results from CNN classification models. Highlighted values indicate superior model performance.

results, we perform all further experiments on the Inception-V3 model only.

Image Classification using different layer outputs. Next, we experiment with different layer outputs to train our classifiers on. We therefore keep the parameters locked, but cut off the network after each module or convolutional layer and feed its output into the block of fully-connected layers.

For the borough classification, this procedure mostly results in similar or worse performance, except when the feature representation from the network's very first convolutional layer is used. Feeding its output directly into the classification block, we achieve an accuracy of 84% and thus superior results to the previous setup. All performance metrics can be seen in Table 4-c. This effect however does not translate to the price bin classification task. Here we do not achieve an improvement over the initial setup when using representations from other layers.

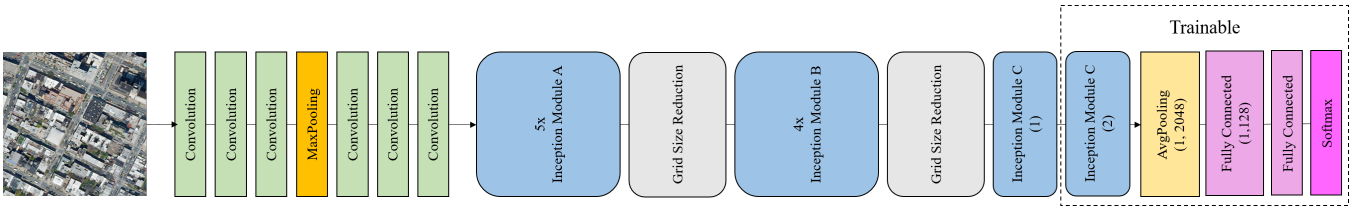
Considering these results, the borough classification model is not further experimented on, as we met our threshold of 80% accuracy. Consequently, the final experimental round focuses on the price bin classification model only.

Image Classification using re-learnt parameters. In the third experimental round, network modules are unlocked and retrained on price bin classification. It should be noted that we could only retrain the last three modules before running into computational bottlenecks. However, this does not seem to be a restriction, as we find that retraining only the final Inception module leads to an accuracy of 80% and thus superior performance to previous training modes, as shown in Table 4-d. Both this and the best performing borough classification model are used as our image feature extractors. Their architecture is visualized in Figure 5 for further clarification.

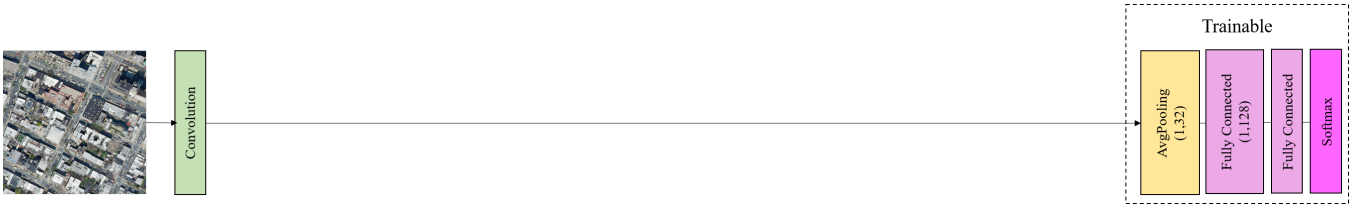
Results from Price Estimation Models

The Effect of Neighborhood Variables. We first fit our regression models on the real estate sales data while holding out locational information and compare the results to those from models including one-hot encoded neighborhood variables. Error values from these tests can be seen in Table 6-a.

We find that including pre-defined neighborhood variables decreases the MAE by 26% for the RF regressor and by 15% for the MLP, from $594k to $438k and from $602k to $509k respectively. While the performance difference between both models is relatively low when using house-level features only, we find that RF performs much better than the MLP as soon as neighborhood variables are included, indicating the advantage of tree-based models over neural networks when applied on structured data with high cardinality features. Differences between model errors are all statistically significant.



(a) Proposed model architecture for price bin classification.



(b) Proposed model architecture for borough classification.

Figure 5: Schematic view of the best performing model architectures. Output shapes are provided for those layers from which feature representations are extracted. For simplification purposes, the auxiliary classifier is not depicted in the drawing.

Model	HF	HF + NBHD
RF	**$594k**	**$438k**
MLP	$602k	$509k

(a) HF and NBHD.

Model	HF + IF	HF + NBHD + IF	IF only
RF	**$445k**	**$465k**	**$621k**
MLP	$659k	$518k	$698k

(b) HF, NBHD and IF extracted from the price bin classification model's first fully-connected layer.

Model	HF + IF	HF + NBHD + IF	IF only
RF	**$435k**	**$443k**	**$604k**
MLP	$573k	$523k	$711k

(c) HF, NBHD and IF extracted from the borough classification model's last convolutional layer.

Figure 6: Resulting MAE values from combining different information sources. HF refers to housing features, NBHD refers to neighborhood variables and IF refers to image features. Highlighted values indicate superior model performance.

The Effect of Image Features. In the following step, we concatenate image and housing features into a single feature vector, testing for the impact of image feature representations from both the last convolutional and the first fully-connected layer of the final CNN models.

First we extract features from the price bin classification model and find that using features obtained from the first fully-connected layer leads to significantly better results than using those from the last convolutional layer. While both approaches yield better results than using solely housing features, they do not outperform the usage of neighborhood variables. Furthermore, including both neighborhood variables and image features does not improve the model performance. Results are presented in Table 6-b.

Secondly, features are extracted from the borough classification model. Here we find an opposite effect, namely that using a feature vector obtained from the last convolutional layer leads to significantly better results than the one from the first fully-connected layer, even though this difference is only $5k and thus rather small in absolute error terms. Most interestingly, features extracted from the borough classification model have a stronger impact on the housing price prediction than those from the previous test, leading to significantly lower prediction errors than the latter and moreover resulting in an absolute error of $435k, which is the lowest error across all experiments. However, the difference between this and the error resulting from the model combining housing features and neighborhood variables is not statistically significant. Again, combining all information sources does not improve the prediction, see Table 6-c.

Moreover, RF vastly outperforms MLP in each experiment. We furthermore find that using image features alone yields the least accurate predictions, even though errors are still below our baseline of $792k.

Model	HF	HF + NBHD	HF + IF
RF	**$409k**	**$288k**	**$292k**
MLP	$433k	$303k	$373k

(a) Residential properties.

Model	HF	HF + NBHD	HF + IF
RF	**$738k**	**$520k**	**$495k**
MLP	$955k	$801k	$714k

(b) Commercial properties.

Figure 7: Resulting MAE values from combining different information sources after splitting data into commercial and residential properties. Highlighted values indicate superior model performance.

Separating Residential and Commercial Properties

As the sales dataset contains houses of various kinds, we decide to perform another series of tests on two distinctive subsets, that is considering residential and commercial properties separately. Since there are nearly twice as many residential as commercial commercial properties, we are especially interested in relative differences between feature combinations. Furthermore, we are only using image feature representations from the last convolutional layer of the borough classification model, as those yielded the best results in previous experiments.

We find that for residential properties, the usage of image features reduces the prediction error by 33% compared to a housing features-only model. As seen in Table 7-a, it results in an error of $292k, which is not significantly lower than the error produced by a model using neighborhood variables.

In the case of commercial properties, we achieve an improvement of 39% when incorporating image features into the price prediction model. Beyond that, the resulting error is significantly lower than the error resulting from a model using neighborhood variables. One can observe further details in 7-b. Like before, RF outperforms MLP in each of our tests.

6 DISCUSSION

Class Activation Mapping

Altogether, our results indicate the potential of satellite images to be used for neighborhood segmentation. Under this assumption, the features extracted from the CNN model should be structural elements or high-level concepts of a certain neighborhood or region, in contrast to specific, ad hoc elements such as parks or other amenities.

In order to further look into this hypothesis, we create Class Activation Maps as proposed by [42]. These maps can be understood as a visualization tool highlighting which regions of an image are important for class discrimination and used by the CNN to identify a certain class. In detail, this is done by average-pooling the gradients of the softmax layer with respect to the output of the last convolutional layer and multiplying these pooled gradients with each channel in the feature map. Intuitively, we hereby amplify channel output values according to their importance in making the class prediction. Feature maps are then averaged per image region and their output activations are plotted in a heat map. Lastly, this heat map is projected on top of the original input image. We focus on the borough classification CNN model, which was shown to be the most effective feature extractor, and create class activation maps for random images from each class. Images are discarded if their prediction probability is below 0,9 to account for model uncertainty.

As one can see in Figure 8, activations span over the whole receptive field while considering various aspects of the image such as parked cars, greenery, building roofs or contours. It should be noted that the model also seems to pick up shadowing which might introduce a learning bias to the prediction model. However, it can be stated that class activations are not triggered by specific objects, but an ensemble of features which are hard to pinpoint individually. At the same time, we do not explicitly exclude the possibility of our model to also pick up specific objects, such as points of interest from the imagery, as suggested in previous work [4]. However, in this case it is assumed that it uses these objects to build higher-level representations of a geographical region. That

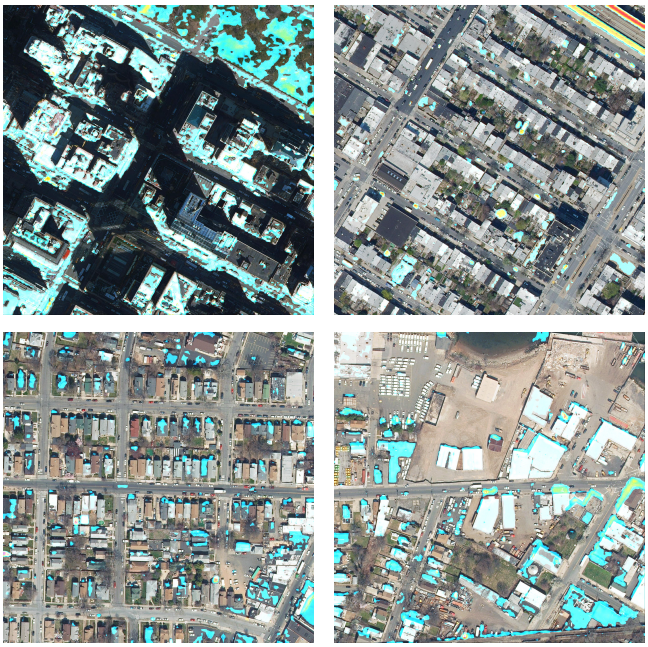


Figure 8: Class Activation Maps for different borough predictions. *Top left:* Manhattan. *Top right:* Brooklyn. *Bottom:* Staten Island.

is, images that all show a certain amount of greenery, density of cars or particular building shapes might be found to belong into the same class.

One could further assume that if a neighborhood shares some of these attributes, it also shares other characteristics, which brings us back to the idea of shared latent features between properties in neighboring regions. It furthermore picks up the notion of similarity in the prices of housing characteristics within a submarket. This might explain our observation that features extracted from the price bin classification model do not outperform those from the borough classification model; a park, basketball court, etc. might simply have a different value and thus impact on the property price across neighborhoods, making it hard to map them to specific price regions.

Random Forest Feature Importance

Finally, we look at the feature importances as determined by the RF estimator. We consider the best performing price estimation model which uses image features extracted from the borough classification model and compare three feature combinations as shown in Figure 9.

Most interestingly, one can see that next to house-level features such as "tot_sqft" (property size) or "yr_built" (construction year), the RF mainly considers building categories when no locational information is provided. On the other hand, the RF assigns higher importances to neighborhood variables as soon as they are fused into the model. In detail, we see that neighborhood variables such as "CHELSEA" or "SOHO" displace most of the categorical building variables. A similar effect can be observed when merging housing and image features. Those latent features, which are represented as numerical values, furthermore oust the variables "tot_unit" (total number of building units) and "res_unit" (number of residential building units), leaving only three house related features in the list of most important features.

Moreover, as it is in the nature of a RF regressor to reduce variance by subsetting the data, we can consider these values another indicator for the capability of satellite images to serve as a basis for submarket segmentation.

Study Limitations

In general, this study and its results are based upon very specific data. That is, it only considers New York City as its sampling location and furthermore uses sales and image data from only one source, respectively. This way of data collection has various consequences. First of all, findings are restricted to New York City and therefore cannot be generalized to other places. NYC has its own unique character and a different urban design than for instance cities like Paris or Amsterdam. One could gain more general insight by drawing from multiple sampling locations instead.

The generalization ability of the proposed system is furthermore limited due to possible biases in the sample data. That means, for instance, that there might be different results if images were collected by a different satellite, under different weather conditions, at different scales, etc. We can also not ensure that sales data has been collected in a proper manner, as there might have been systematic flaws in the data collection process. In that light, all reported results or errors cannot be interpreted as general prediction errors, but statistical errors specific to the dataset at hand.

This also applies to the reported model performances. As we used fixed hyperparameters and did not optimize our models on the data and task at hand, we cannot view their errors as a general performance measure for those models, but as statistical errors produced under a specific set of circumstances which do not only include model hyperparameters but also span over the way we prepared our data. Furthermore, other models might achieve similar or better results if properly optimized. Again, one could only draw general conclusions after comparing models under a range of different conditions.

Another limitation is presented in the way our dataset was cleaned. By removing nearly 35% of the available data, we might have forfeited representational power of our sample with regards to actual market realities. At the same time, data set selection was made to ensure that model training was only performed in well-sampled regions of label space,

Feature	Importance
tot_sqft	0.349
yr_built	0.168
tot_unit	0.116
land_sqft	0.096
13 CONDOS - ELEVATOR APARTMENTS	0.082
res_unit	0.074
com_unit	0.037
10 COOPS - ELEVATOR APARTMENTS	0.008
26 OTHER HOTELS	0.007
07 RENTALS - WALKUP APARTMENTS	0.007

(a) HF

Feature	Importance
tot_sqft	0.295
yr_built	0.189
land_sqft	0.058
13 CONDOS - ELEVATOR APARTMENTS	0.054
GREENWICH VILLAGE-WEST	0.025
tot_unit	0.021
UPPER EAST SIDE (59-79)	0.020
res_unit	0.017
CHELSEA	0.015
SOHO	0.015

(b) HF and NBHD

Feature	Importance
tot_sqft	0.257
yr_built	0.062
13 CONDOS - ELEVATOR APARTMENTS	0.050
6	0.049
land_sqft	0.048
13	0.033
11	0.030
19	0.029
27	0.027
0	0.023

(c) HF and IF

Figure 9: Top-10 Random Forest feature importances for different combinations of information sources.

as we did not want our models to overfit to characteristics of unrealistically cheap or extremely expensive properties.

7 CONCLUSION

Overall, we have shown that satellite images can potentially serve as segmenters for real estate submarkets. For that purpose, we extracted image feature representations from generic layers of the pre-trained Inception-V3 model and merged those with known property features before fitting a Random Forest regressor. This method yields results comparable to the usage of manually defined and highly fine-grained neighborhood variables. It furthermore outperforms previously proposed methods, in which image descriptors are extracted from specialized CNN models trained on property price prediction or classification tasks, and thus requires a less complex training procedure at lower computational cost. Further experimentation is required in order to see if and how this approach can be applied to other datasets.

8 ACKNOWLEDGMENTS

I would like to express my deep gratitude to my supervisors Dr. Andrew Brown and Maurice Hoenderdos MSc for their guidance and encouragement during the planning and development of this research work. I have been extremely lucky to have supervisors who cared so much about my work, and who always took their time to answer my questions or discuss my ideas.

REFERENCES

[1] E. Ahmed and M. Moustafa. House price estimation from visual and textual features. *arXiv preprint arXiv:1609.08399*, 2016.

[2] S. M. Arietta, A. A. Efros, R. Ramamoorthi, and M. Agrawala. City forensics: Using visual elements to predict non-visual city attributes. *IEEE transactions on visualization and computer graphics*, 20(12):2624–2633, 2014.

[3] S. Basu and T. G. Thibodeau. Analysis of spatial autocorrelation in house prices. *The Journal of Real Estate Finance and Economics*, 17(1):61–85, 1998.

[4] A. J. Bency, S. Rallapalli, R. K. Ganti, M. Srivatsa, and B. Manjunath. Beyond spatial auto-regressive models: Predicting housing prices with satellite imagery. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 320–329. IEEE, 2017.

[5] Z. Bessinger and N. Jacobs. Quantifying curb appeal. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 4388–4392. IEEE, 2016.

[6] S. C. Bourassa, E. Cantoni, and M. Hoesli. Spatial dependence, housing submarkets, and house prices. 2005.

[7] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[8] M. Čeh, M. Kilibarda, A. Lisec, and B. Bajat. Estimating the performance of random forest versus multiple regression for predicting prices of the apartments. *ISPRS International Journal of Geo-Information*, 7(5):168, 2018.

[9] J.-H. Chen, C. F. Ong, L. Zheng, and S.-C. Hsu. Forecasting spatial dynamics of the housing market using support vector machine. *International Journal of Strategic Property Management*, 21(3):273–283, 2017.

[10] F. Chollet et al. Keras, 2015.

[11] D. E. Clark and J. R. Kahn. The social benefits of urban cultural amenities. *Journal of regional science*, 28(3):363–377, 1988.

[12] N. R. F. Collaboration et al. A century of trends in adult human height. *Elife*, 5:e13410, 2016.

[13] F. Des Rosiers, A. Lagana, M. Thériault, and M. Beaudoin. Shopping centres and house values: an empirical investigation. *Journal of Property Valuation and Investment*, 14(4):41–62, 1996.

[14] A. Q. Do and G. Grudnitski. A neural network approach to residential property appraisal. *The Real Estate Appraiser*, 58(3):38–45, 1992.

[15] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012.

[16] GIS Lab, Newman Library, Baruch CUNY. 2015 New York City Real Estate Sales. https://geo.nyu.edu/catalog/nyu-2451-34678.

[17] 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.

[18] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[19] E. Kamptner. NYC Orthoimagery. GitHub repository. https://github.com/CityOfNewYork/nyc-geo-metadata/blob/master/Metadata/Metadata_AerialImagery.md.

[20] F. Kong, H. Yin, and N. Nakagoshi. Using gis and landscape metrics in the hedonic price modeling of the amenity value of urban green space: A case study in jinan city, china. *Landscape and urban planning*, 79(3-4):240–252, 2007.

[21] 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.

[22] S. Law. Defining street-based local area and measuring its effect on house price using a hedonic price approach: The case study of metropolitan london. *Cities*, 60:166–179, 2017.

[23] S. Law, B. Paige, and C. Russell. Take a look around: using street view and satellite images to estimate house prices. *arXiv preprint arXiv:1807.07155*, 2018.

[24] J. LeSage and R. K. Pace. *Introduction to spatial econometrics*. Chapman and Hall/CRC, 2009.

[25] S. Malpezzi. Hedonic pricing models: A selective and applied review, w housing economics and public policy: Essays in honor of duncan maclennan, red. t. oâĂŹsullivan, k. gibb, 2003.

[26] M. L. McMillan, B. G. Reid, and D. W. Gillen. An extension of the hedonic approach for estimating the value of quiet. *Land economics*, 56(3):315–328, 1980.

[27] M. Minsky and S. Papert. Perceptron: an introduction to computational geometry. *The MIT Press, Cambridge, expanded edition*, 19(88):2, 1969.

[28] B. A. Newsome and J. Zietz. Adjusting comparable sales using multiple regression analysis-the need for segmentation. *The Appraisal Journal*, 60(1):129, 1992.

[29] N. Nghiep and C. Al. Predicting housing value: A comparison of multiple regression analysis and artificial neural networks. *Journal of real estate research*, 22(3):313–336, 2001.

[30] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[31] L. Osland. An application of spatial econometrics in relation to hedonic house price modeling. *Journal of Real Estate Research*, 32(3):289–320, 2010.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[33] S. Peterson and A. Flanagan. Neural network hedonic pricing models in mass real estate appraisal. *Journal of Real Estate Research*, 31(2):147–164, 2009.

[34] D. Quercia, R. Schifanella, and L. M. Aiello. The shortest path to happiness: Recommending beautiful, quiet, and happy routes in the city. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 116–125. ACM, 2014.

[35] R. G. Ridker and J. A. Henning. The determinants of residential property values with special reference to air pollution. *The Review of Economics and Statistics*, pages 246–257, 1967.

[36] S. Rosen. Hedonic prices and implicit markets: product differentiation in pure competition. *Journal of political economy*, 82(1):34–55, 1974.

[37] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[38] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[39] E. Worzala, M. Lenk, and A. Silva. An exploration of neural networks and its application to real estate valuation. *Journal of Real Estate Research*, 10(2):185–201, 1995.

[40] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[41] Q. You, R. Pang, L. Cao, and J. Luo. Image-based appraisal of real estate properties. *IEEE Transactions on Multimedia*, 19(12):2751–2759, 2017.

[42] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[43] J. Zurada, A. Levitan, and J. Guan. A comparison of regression and artificial intelligence methods in a mass appraisal context. *Journal of Real Estate Research*, 33(3):349–387, 2011.