House Price Prediction via Visual Cues and Estate Attributes

Sai Surya Vaddi, Amira Yousif, Samah Baraheem, Ju Shen, Tam V. Nguyen*

Department of Computer Science, University of Dayton, Dayton OH 45469, USA

*Corresponding Author: tamnguyen@udayton.edu

Abstract. The price of a house depends on many factors, such as its size, location, amenities, surrounding establishments, and the season in which the house is being sold, just to name a few of them. As a seller, it is absolutely essential to price the property competitively else it will not attract any buyers. This problem has given rise to multiple companies as well as past research works that try to enhance the predictability of property prices using relevant mathematical models and machine learning techniques. In this research, we investigate the usage of machine learning in predicting the house price based on related estate attributes and visual images. To this end, we collect a dataset of 2,000 houses across different cities in the United States. For each house, we annotate 14 estate attributes and five visual images for exterior, interior-living room, kitchen, bedroom, and bathroom. Following the dataset collection, different features are extracted from the input data. Furthermore, a multi-kernel regression approach is used to predict the house price from both visual cues and estate attributes. The extensive experiments demonstrate the superiority of the proposed method over the baselines.

Keywords: house price prediction, multi-kernel learning, visual computing, estate attributes, computer vision, machine learning.

1 Introduction

"There's no place like home."

For us, houses are full of memories. A house may range in complexity from a rudimentary hut to a complex structure of wood, masonry, concrete, or other material, outfitted with plumbing, electrical, and heating, ventilation, and air conditioning systems. However, buying a house relies on many factors, including location, number of bedrooms, its appearance, the size, and most importantly, the price. Housing price plays a significant role in the real estate market. Housing renovation and construction boost the economy by increasing the house sales rate, employment and expenditures. The value of the asset portfolio for households whose house is their largest single asset is highly affected by the oscillation of the house prices. There have been many research works about estimating the house prices. Ahmed and Moustafa [1] estimate the house price from visual and textual features by a simple neural network model [12]. However, only 535 houses in California, United States are used. In addition, the model simply concatenates all visual and textual features; and thus, there is no explanation about the impact of each feature type. Khamis and Kamarudin [9] conduct a study to compare the performance between multiple linear regression (MLR) model [11] and neural network model [12] on estimating house prices in New York. As discussed in [1], predicting the houses' prices is a very difficult task due to the illiquidity and heterogeneity in both the physical and the geographical perspectives of the houses market.

In this paper, we first collect a dataset of 2000 houses in various states in the US. For each house, we collect five images of different locations and 14 attributes, such as the number of rooms, the size in sq ft, the school grades, Home Owner Association (HOA) [18] fee, among others. Then, a computational multi-kernel regression model is implemented to utilize the features from different modalities; and therefore, accurately estimating the house price. The extensive experiments on the newly constructed dataset demonstrate the superiority of our proposed method.

The remainder of this paper is organized as follows. Section 2 summarizes the related work. Section 3 introduces the data collection and the computational model. The experimental results are presented in Section 4. Finally, Section 5 concludes the paper and paves way to the future work.

2 Related Work

A variety of research work has been done to estimate housing prices. These approaches used for house price prediction can be classified into machine learning models, hybrid models, and regression models. We believe that machine learning house price prediction models will help those who work in the real estate market in making a good house purchasing decision.

Ahmad and Moustafa [1] propose using visual and textual features to predict the house price. Then, these extracted features are passed through a neural network model (NN). Moreover, the features are passed into the SVR model to compare. Through experiments the visual features increased the R-value by a factor of three and decreased the Mean Square Error by one. The NN achieves a lower R-value of 0.9 (0-1). Furthermore, good results come from using NN over SVM using the same dataset. Different 11 machine learning regression models are used by Imran et al. [2] and evaluated using three different metrics which are MAPE (Mean absolute percentage error), RMSE (Root Mean Squared Error), and MAE (Mean absolute error) to determine the best model for a better housing price prediction of the capital Islamabad. Lathuili et al. [3] analyze different techniques to perform regression through neural networks. The experimental interested in vanilla deep regression methods –short for convolutional neural networks with a linear regression layer– and their variants. These measures are used to discern between variations due to stochastic effects from systematic improvements. The results show that, in three out of four problems, correctly fine-tuned deep regression networks can compete with problem-specific methods in the literature that are entirely devoted to solving only one task.

In another work, Truong et al. [4] mainly discuss combining different models to create a unified model and validating multiple techniques in model implementation on regression. This study provides an optimistic result for housing price prediction. There are three different types of machine learning methods and Hybrid Regression techniques. The Random Forest has the lowest error on the training set, but it is prone to be overfitting. The time complexity of XGBoost [15] and LightGBM are the best. Even though Hybrid Regression and Stacked Generalization Regression deliver satisfactory results. Chen et al. [5] build an array of machine learning models to predict the price of a product given its image. Additionally, this work visualizes the features that result in higher or lower price predictions. For price regression, they use linear regression on histogram of oriented gradients, convolutional neural network (CNN) features, and a baseline for price segment classification using a multiclass SVM. Moreover, they use several recently-developed methods to visualize the image features that result in higher or lower prices. Nissan et al. [6] leverage various algorithms to predict real estate property prices in Montreal. The study suggests a prediction model that predicts asking and selling prices based on features, such as location, area, rooms, nearest police station, fire station, etc. They use many regression models for regression prediction. These regression methods include linear regression, SVR, KNN, Regression Tree, and Random Forest Regression. The proposed prediction models predict the asking price with an error of 0.0985, and the selling price with an error of 0.023.

Meanwhile, Mysore et al. [13] compare different algorithms, such as linear regression, support vector regressor, XGBoost[15] regressor, CatBoost regressor, and others to predict the house prices. CatBoost Regressor and SVR have the highest accuracy by 90%, and the accuracy improves up to 15%. Park et al. [14] compare 20 selection and extraction features algorithms for housing in King County, USA with Support Vector Regression (SVR) to predict the house prices. After all experiments, the price prediction accuracy increases by 0.21, from 0.65 to 0.86. There is no difference between PCA- SVR and feature selections – SVR in price perdition. Liu [16] predicts the real estate price and then predicts the real estate market by using multiple linear regression models and the least square method to predict and analyze the real estate market.

Recently, Kumar et al. [17] use the Decision Tree machine learning algorithm to predict house prices in Mumbai city, where additional features, such as air quality and crime rate are used to help in predicting the prices. In fact, the result for using Decision Tree Regressor gives an accuracy of 89%. Hamizah Zulkifley et al.[19] examine the attributes used in previous studies and predict house prices using different machine learning models, such as SVR, Artificial Neural Network (ANN), and XGBoost. The aim of this study is to help other researchers to develop a better model that accurately predicts the house price. As a result, this aids the house buyers and investors to determine the realistic prices of houses and use the best machine learning model. Nur et al.[20] rely on three factors: physical conditions, concept, and location to predict the price of a house. Different tests, such as linear regression and particle swarm optimization (PSO) methods are leveraged to predict the house price.

Fig. 1. Our visual data collected from Zillow [21], in the United States. Each house instance is accompanied by five images, namely, the frontal side, the living room, the kitchen, the bathroom and the bedroom.

From the literature review, there is a legitimate need for us to collect a new benchmark dataset. In addition, we also propose a computational model to tackle this house price prediction problem.

3 Proposed Work

This research focuses on evaluating the impact of images and estate attributes in predicting house prices. In the first stage, we rely on two popular real-estate listing websites, namely Zillow [21] and Redfin [22]. The data are collected from these websites manually to acquire accurate data. The outcome of this stage is a dataset consisting of 2000 houses. Each house instance consists of 14 estate attributes and five images. It is to be noted that since we have images in our dataset, we must implement a mechanism to translate the image data into a type compatible with our model down the stream. Therefore, in the next stage, the features are extracted in the compatible format from all the images. Then, in the third stage, we present the multi-kernel regression for the house price prediction problem. Next, different models are evaluated for predicting the price of each house on the newly collected dataset. The predictions are compared with ground truth values in terms of Mean Absolute Error computations to assess the performance. The mentioned steps are elaborated in detail in the subsequent subsections.

3.1 Data Collection

The collected dataset is composed of 2000 sample houses from 10 States and 39 cities in the United State. The dataset is collected and annotated manually from publicly available information on websites that sell houses. The most popular real-estate listing websites, namely Zillow [21] and Redfin [22] are used. The outcome of this step is a dataset consisting of 2,000 houses with 14 estate attributes and five images. Therefore, each house is represented by both visual and textual data. The visual data is a set of five images for the frontal image of the house, the living room, the kitchen, the bathroom and the bedroom, as shown in Figure 1. The textual data shows the numeric attributes of the house, such as zip code, price, number of bedrooms, number of bathrooms, house size, built year, lot area, HCA, and schools' area, as can be seen in Table 1. The house price in the dataset ranges from $445.00 to $4,682,883.00. Finally, the dataset is split into two subsets, namely, training set and testing set. In particular, 80% of data is used for training purposes and the remaining 20% used for testing purposes.

Table 1: An excerpt of the estate attributes in our collected house dataset.

NO	Address	Zip	PRICE	bed	bath	size	build year	lot area	HOA	E	M	HS	Sea
1	5316 Hidden Creek Cir,Mason,OH 45040	45040	420,000	4	4	2,019	1992	0.38	185	5	6	3	1
2	6309 Tarton Fields Ln, Mason, OH 45040	45040	630,000	4	5	4,526	1997	0.51	495	5	6	3	1
3	3533 W Cortland St, Chicago, IL 60647	60647	865000	4	4	3103	2021	0.23	0	5	6	3	1
4	2627 N Talman Ave, Chicago, IL 60647	60647	715000	3	3	1990	1963	0.07	0	5	6	3	1
5	1392 Corral Way, Frankfort, KY 40601	40601	145000	3	2	1230	1999	0	0	5	6	3	1
6	212 Bracken Ct, Frankfort, KY 40601	40601	349900	5	3	2820	2006	0.51	0	5	6	3	1
7	6913 Blake Dr, Fort Wayne, IN 46804	46804	260000	3	3	2688	1965	0.98	90	5	6	3	1
8	350 Twillo Run, New Haven, IN 46774	46774	150000	3	2	1780	1959	0.201	0	5	6	3	1
9	6008 S 14th Pl, Phoenix, AZ 85042	85042	345000	4	2	1365	1972	0.175	0	5	6	3	1
10	905 E Pedro Rd, Phoenix, AZ 85042	85042	430000	4	2	1967	2002	0.16	110	5	6	3	1
11	2044 N Sawyer Ave, Chicago, IL 60647	60647	550000	5	3	2200	1986	0.07	0	5	6	3	1
12	2545 W Logan Blvd, Chicago, IL 60647	60647	950000	3	3	2500	1890	0.18	0	5	6	3	1
13	2406 22nd St, Troy, NY 12180	12180	185000	5	3	3148	1977	0.109	0	5	6	3	1
14	2311 14th St, Troy, NY 12180	12180	77500	3	1	1076	1890	0.05	0	5	6	3	1
15	4414 Park Ave, Richmond, VA 23221	23221	521000	4	2	1637	1950	0.14	0	5	6	3	1
16	3222 Garrett St, Richmond, VA 23221	23221	400500	4	2	1441	1945	0.16	0	5	6	3	1
17	10420 Groton St, Orlando, FL 32817	32817	313000	3	2	993	1960	0.18	0	5	6	3	1
18	1809 Angela Dr, Orlando, FL 32817	32817	380000	3	2	1724	1973	0.5	0	5	6	3	1
19	5116 Yaupon Dr, Arlington, TX 76018	76018	269500	3	2	1516	1985	0.18	0	5	6	3	1
20	4206 Oakside Ct, Arlington, TX 76016	76016	276000	3	2	1632	1981	0.116	0	5	6	3	1

3.2 Computational Model

Inherently, each pixel corresponds to a feature for any given image. Therefore, if an image contains million pixels, this means that, on a fundamental level, the image can be identified in terms of these million feature dimensions. The presence of such magnitude of dimensions naturally proves to be computationally expensive with respect to the computer's memory resources and power consumption when it comes to processing images. As our dataset consists of 10,000 images in total, it is essential to devise a mechanism to compress each of these images such that each image can be represented in terms of fewer dimensions in order to make downstream computations more feasible. In this section, we elaborate on the feature extraction process, and the downstream model that uses these compressed images as part of its input to predict prices. A visual representation of our computational model framework is illustrated in Figure 2.

3.2.1 Feature Extraction

Feature extraction is an impactful technique employed to counter the curse of dimensionality. The term 'curse of dimensionality' refers to the difficulties involved in performing computations on data with huge numbers of dimensions, features, or attributes. In essence, this method involves mapping a set of attributes to a smaller feature space while preserving the true 'nature' of the data. To extract the features, a pre-trained neural network is used to obtain a compressed representation of each image in our dataset. More specifically, we use a pre-trained Convolutional Neural Network namely PlacesCNN [23] for the feature extraction process as inspired by [24]. A convolutional neural network is a widely used variant of a neural network that specializes in classifying images. The PlacesCNN [23] is trained with 2.5 million images in different scenes. It can identify images of various scenes spanning across 205 scene categories. However, before each image is fed into this network, specific transformations are applied for the sake of compatibility. We first resize images to 256x256 dimensions. Then, converting images to grayscale and centering the contents are applied. The transformed image is then fed into the Neural Network as an input. Note that originally, the PlacesCNN [23] was designed to classify images in terms of the 205 scene categories. This final result is contained in the last layer, i.e. output of the network. Hence, we instead extract deep learning features from within this network that reside in its penultimate layer. These deep learning features are of size 4096. As a result, we collect this array of 4096 decimal numbers corresponding to each image input and use these features to represent each image in our dataset. This flow is clearly illustrated in Figure 2.

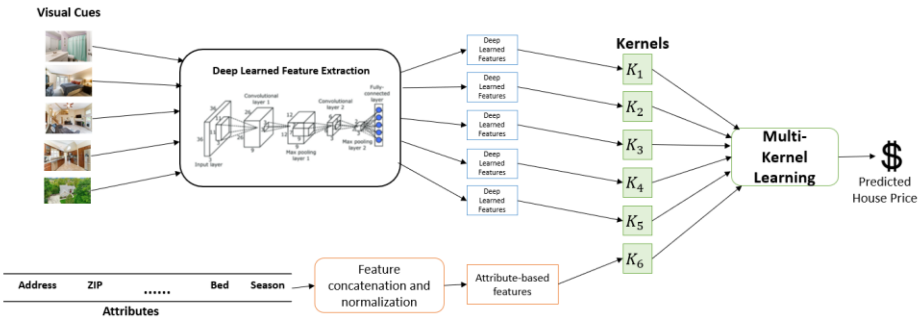

Fig. 2. The overview of our proposed computational framework. The input images are fed into a CNN to extract the deep learned features. In addition, the estate attributes are normalized and concatenated to form another feature set. The features from separate kernels are fed into a multi-kernel regression model. Finally, the model is trained to predict the house price.

3.2.2 **Multi-Kernel Regression for House Prediction**

Following the feature extraction stage, there are six feature sets. One naive way is to concatenate all features and feed them into a regression model such as Support Vector Regressor (SVR). However, the features are in high-dimensional data. In addition, we consider each feature set reflecting a unique perspective of the house. Here, a single feature set can be used for kernel learning. By using all the given feature sets, we present a multi-kernel support vector regression method for the house price prediction. Suppose we are given a set of *n* training examples

$$\{(x_1, z_1), (x_2, z_2), \ldots, (x_n, z_n)\}$$

, where *x* is the feature and *z* is the corresponding label. We aim to learn a regression function:

$$f(x) = \omega^T \phi(x) + \beta \quad (1)$$

that can best predict the label of an unseen data point x, where ω is a weight in the kernel feature space, $\phi(x)$, the kernel feature map of x, and β, a threshold constant, f can be solved through the following dual optimization problem:

$$\max_{\alpha^+,\alpha^-} \; -\frac{1}{2}(\alpha^+ - \alpha^-)^T K(\alpha^+ - \alpha^-) - \epsilon \sum_{i=1}^{l}(\alpha_i^+ + \alpha_i^-) + \sum_{i=1}^{l} z_i\,(\alpha_i^+ + \alpha_i^-)$$
$$s.t. \sum_{i=1}^{t}(\alpha_i^+ + \alpha_i^-) = 0 \qquad (2)$$
$$\alpha_i^+ \in [0, C]$$
$$\alpha_i^+ \in [0, C] \quad i = 1, \ldots., l$$

, where K is a kernel matrix and C adjusts the tradeoff between the regression error and the regularization of f. Solving α+, α−, and β using KKT (Kurash-Kuhn-Tucker) conditions applied to (2), the regression function of (1) becomes

$$f(x) = \sum_{i-1}^{l}(\alpha^+ - \alpha^-)K(x, x_i) + \beta_, \qquad (3)$$

In the regression function in (3), K is usually a single kernel function. Since a combination of kernels with nonnegative coefficients yields a legitimate kernel, we can combine them into one kernel. In our case since we have a set of $\mathcal{K}$ kernel matrices, we use the summation operation to sum up all kernel matrices

$$\mathcal{K} = \sum_i \; K_i \qquad (4)$$

In this work, we combine multiple kernels from different modalities, such as visual cues and attributes. For the testing, we extract the features from the test house, compute the kernel, and feed the kernel to the trained model to predict the corresponding house price.

4 Experiments

In order to assess the performance of our Multi-Kernel Support Vector Machine model, this research undertakes a layered approach in evaluation of various SVM models differentiated by kernel configuration. In this section, the used evaluation metrics are discussed first, followed by the experimental results.

4.1 **Evaluation Metrics**

The Multi-Kernel Regression model is evaluated against eight baseline models in terms of Mean Absolute Error (MAE) which is widely used in literature [20, 25, 26]. It is to be noted that the target variable, which is the house price, is scaled down to the ten thousandths of its original value. Below is the formulation of MAE.

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \qquad (5)$$

, where y is the predicted price, x is the ground truth, and n is the total number of data points in our testing dataset. Regarding the baselines, we consider the popular regression methods such as Linear SVR, linear regression, regression tree, non-negative least square, nearest neighbor, and K-nearest neighbor. For all baselines, all features, i.e., visual cues and estate attributes are concatenated.

Table 2. Comparison between our proposed method and the baselines in terms of MAE. The best performance is marked in **bold-face**.

Method	**MAE ↓**
Linear SVR	16.39
Linear Regression	21.40
Regression Tree	16.77
Non-negative least square	48.64
Nearest neighbor	21.13
K-nearest neighbor (K=5)	17.93
K-nearest neighbor (K=10)	17.68
K-nearest neighbor (K=20)	17.37
K-nearest neighbor (K=30)	17.34
K-nearest neighbor (K=50)	17.69
K-nearest neighbor (K=100)	18.14
Multi Kernel Regression (Ours)	**14.70**

4.2. **Experimental Results**

As seen in Table 2, our Multi Kernel Regression model achieves the lowest error MAE with 14.70. Meanwhile, the highest error MAE is obtained by using the non-negative least square model [25], reaching 48.64. The K-nearest neighbor (KNN) based regression performs better when the number of neighbors, K, increases from 1 to 30. However, when K is too large, i.e., 50 or 100, the results of KNN get worse. Our model performs better than the regression models and also better than the linear kernel SVR model which has the MAE of 16.39. Our model also performs better than KNN models. We attribute this to the ability of the Multi-Kernel model to be more accommodating of each data point's variation across different modalities in the given feature set, which is not the case in all of our baseline models. Since our dataset is a combination of heterogeneous attributes i.e, estate attributes and images of five different locations pertaining to the property, the insensitivity of the baseline models to this level of modularity contributes to the poor accuracy of predicted house price. Overall our Multi-Kernel Regression model is superior to all the tested models in terms of accuracy.

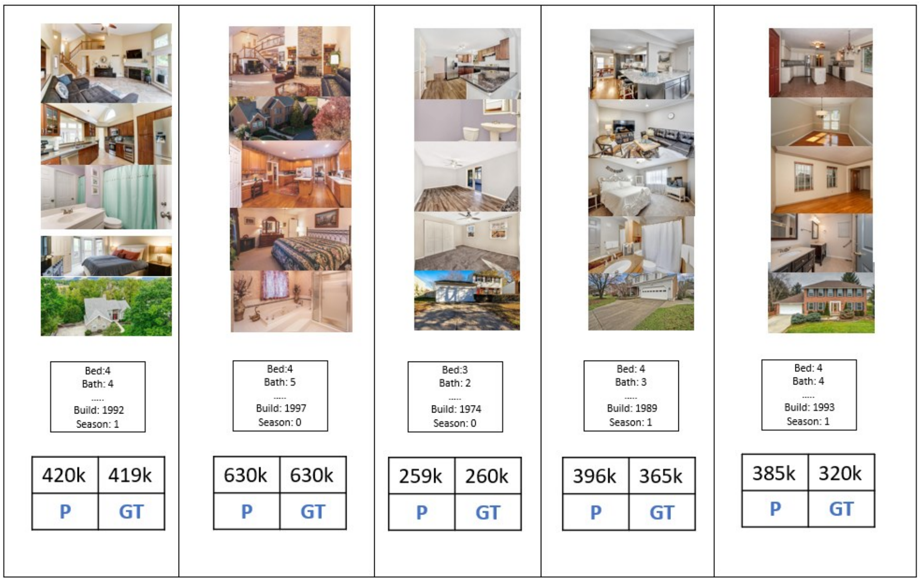

Fig 3. The visualization of the predicted house price. P is for predicted price whereas GT is for the ground truth price.

Table 3. Ablation studies of our proposed multi-kernel regression model with various kernel configurations. In particular, the kernel is configured only with a single feature.

Kernel Configuration	**MAE ↓**
Bathroom Image	19.35
Bedroom Image	19.59
Living Room Image	19.02
Kitchen Image	18.51
House Exterior Image	21.42
Estate Attributes	18.81

Figure 3 shows the visualization of the predicted house price. Each column in the figure represents a sample data point from our dataset. As can be seen, the first three examples show very good prediction results. In the last two examples, the predicted price is higher than the actual price. The reason may be the images of these houses contain attributes of a more expensive house, however, the house belongs to a city that has a low average house price.

4.3. **Ablation Studies**

Following the baseline comparison, we further analyze the price predictions from the proposed Multi-Kernel Regression model with different kernel configurations, namely, single kernel for bathroom, bedroom, living room, kitchen, house exterior, and estate attributes. Table 3 summarizes our findings. Interestingly, from our studies, it can be seen that the image of the kitchen alone is marginally better in predicting the house price than the estate attributes that consist of variables, such as square footage, number of bedrooms, number of bathrooms, zip code etc. One obvious observation is that our model produces most accurate results when the kernel comprises all features, as indicated in Table 2.

5 Conclusion and Future Works

During the past few years, the real estate market has been rapidly rising. Accurately predicting the house price via analyzing the features of a given house is an interesting problem. In this research, we first collect a database of 14 estate attributes and five images for 2,000 houses across more than 39 cities in the United States of America. For the images, their dimensionality is reduced through transfer learning techniques to make computations more efficient. To this compressed image dataset and the numerical estate attributes, a Multi-Kernel Regression model is used to predict the price of each house. We further compare the model with other baseline models in terms of Mean Absolute Error. Moreover, we conduct an ablation study on how our model performs in case the data comprises only one individual feature. It is found that the Multi-Kernel Regression model produces the best result when all the features are considered. In particular, the Multi Kernel Regression model achieves 14.7 in terms of MAE.

For future works, we aim to extend this work by addressing class imbalances. In particular, we make sure to have an equal number of houses in each class, such as accross price ranges and zip codes. In addition, we plan to add more samples to our dataset. The sample size of our dataset can be increased to more accurately represent the diverse housing options across the United States. We further aim to explore feature-engineering options on the dataset.

Acknowledgment

This research was supported by the National Science Foundation (NSF) under Grant 2025234.

References

1. E. H. Ahmed and M. Moustafa, “House Price Estimation from Visual and Textual Features,” *Proceedings of the 8th International Joint Conference on Computational Intelligence*, 2016, doi: 10.5220/0006040700620068.

2. Imran, U. Zaman, M. Waqar, and A. Zaman, “Using Machine Learning Algorithms for Housing Price Prediction: The Case of Islamabad Housing Data,” *Soft Computing and Machine Intelligence Journal*, vol. 1, no. 1, Jul. 2021, doi: DOI:10.22995/scmi.2021.1.1.03.
3. S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, “A Comprehensive Analysis of Deep Regression,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 9, pp. 2065–2081, Sep. 2020, doi: 10.1109/TPAMI.2019.2910523.
4. Q. Truong, M. Nguyen, H. Dang, and B. Mei, “Housing Price Prediction via Improved Machine Learning Techniques,” *Procedia Computer Science*, vol. 174, pp. 433–442, 2020, doi: 10.1016/j.procs.2020.06.111.
5. R. R. Yang, S. Chen, and E. Chou, “AI Blue Book: Vehicle Price Prediction using Visual Features,” *arXiv:1803.11227 [cs]*, Mar. 2018
6. N. Pow, E. Janulewicz, and L. (Dave) Liu, “Applied Machine Learning Project 4 Prediction of real estate property prices in Montréal.” *Course project, COMP-598, Fall/2014, McGill University*, 2014.
7. A. Rawool, D. Rogye, S. Rane, and V. Bharadi, “House Price Prediction Using Machine Learning.” *Iconic Research And Engineering Journals*, vol. 4, no. 11, pp. 29–33, Aug. 2021.
8. B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. “Learning Deep Features for Scene Recognition using Places Database.” Advances in Neural Information Processing Systems 27 (NIPS), 2014
9. A. Bin Khamis and N. Kamarudin, “Comparative Study On Estimate House Price Using Statistical And Neural Network Model,” *International Journal of Scientific & Technology Research*, vol. 3, no. 12.
10. S. Qiu and T. Lane, “A Framework for Multiple Kernel Support Vector Regression and Its Applications to siRNA Efficacy Prediction,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 6, no. 2, pp. 190–199, Apr. 2009, doi: 10.1109/tcbb.2008.139.
11. A. Chakraborty and D. Goswami, “Prediction of slope stability using multiple linear regression (MLR) and artificial neural network (ANN),” *Arabian Journal of Geosciences*, vol. 10, no. 17, Sep. 2017, doi: 10.1007/s12517-017-3167-x.
12. W. Gerstner, “Time structure of the activity in neural network models,” *Physical Review E*, vol. 51, no. 1, pp. 738–758, Jan. 1995, doi: 10.1103/physreve.51.738.
13. S. Mysore, “Prediction of House Prices Using Machine Learning,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 6, pp. 1780–1785, Jun. 2022, doi: 10.22214/ijraset.2022.44033.
14. B. Park and J. K. Bae, “Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data,” *Expert Systems with Applications*, vol. 42, no. 6, pp. 2928–2934, Apr. 2015, doi: 10.1016/j.eswa.2014.11.040.
15. Y. Zhao, G. Chetty, and D. Tran, “Deep Learning with XGBoost for Real Estate Appraisal,” *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2019, doi: 10.1109/ssci44817.2019.9002790.
16. G. Liu, “Research on Prediction and Analysis of Real Estate Market Based on the Multiple Linear Regression Model,” *Scientific Programming*, vol. 2022, pp. 1–8, May 2022, doi: 10.1155/2022/5750354.

17. G. K. Kumar, D. M. Rani, N. Koppula, and S. Ashraf, "Prediction of House Price Using Machine Learning Algorithms," *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, Jun. 2021, doi: 10.1109/icoei51242.2021.9452820.
18. E. Chang, "What Is A Homeowners Association, or HOA?," *Bankrate*. https://www.bankrate.com/real-estate/what-is-an-hoa/ (accessed Jul. 13, 2022).
19. N., S. Abdul Rahman, N. H. Ubaidullah, and I. Ibrahim, "House Price Prediction using a Machine Learning Model: A Survey of Literature," *International Journal of Modern Education and Computer Science*, vol. 12, no. 6, pp. 46–54, Dec. 2020, doi: 10.5815/ijmecs.2020.06.04.
20. A. Nur, R. Ema, H. Taufiq, and W. Firdaus, "Modeling House Price Prediction using Regression Analysis and Particle Swarm Optimization Case Study : Malang, East Java, Indonesia," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017, doi: 10.14569/ijacsa.2017.081042.
21. Zillow Real Estate, https://www.zillow.com/, last retrieved on July 15, 2022.
22. Redfin: Real Estate, Homes for Sale, MLS Listings, Agents, , last retrieved on July 15, 2022.
23. B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, 2018.
24. T. V. Nguyen, L. Liu, K. Nguyen, "Exploiting generic multi-level convolutional neural networks for scene understanding", *International Conference on Control, Automation, Robotics and Vision*, pp. 1-6, 2016.
25. T. Sherk, M-T.Tran, T.V. Nguyen, "SharkTank Deal Prediction: Dataset and Computational Model", *International Conference on Knowledge and Systems Engineering*, pp. 1-6, 2019.
26. T. V. Nguyen, J. Feng, S. Yan, "Seeing Human Weight from a Single RGB-D Image", in *Journal of Computer Science and Technology*, Vol. 29(5), pp. 777-784, 2014.