# Satellite Imagery Based Property Valuation

## 1. Overview

Accurately estimating real estate prices is a critical problem in urban planning, finance, and real-estate markets. Traditional valuation models rely heavily on structured tabular features such as location, size, and number of rooms. However, satellite imagery provides rich visual context about surroundings such as greenery, urban density, road connectivity, and neighborhood layout, which may influence property value.

This project explores whether **combining satellite images with tabular housing data** can improve property price prediction. A **multimodal deep learning model** is developed that fuses visual features extracted using a pretrained Convolutional Neural Network (CNN) with tabular features processed using a Multilayer Perceptron (MLP).

## 2. Dataset Description

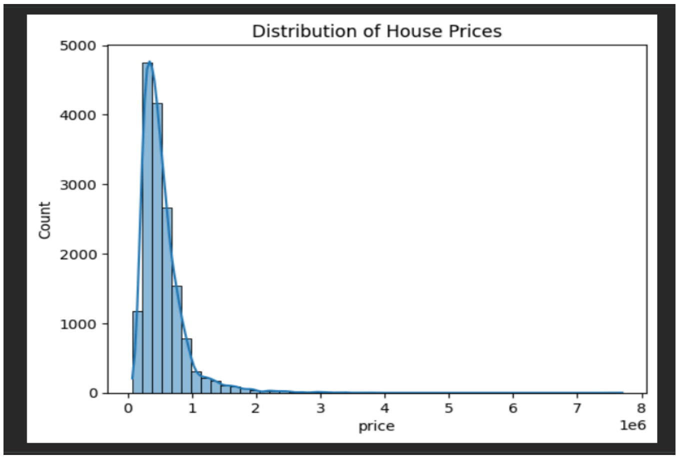
The dataset consists of:

- **Tabular housing attributes** such as:
    - Bedrooms, bathrooms
    - Living area and lot size
    - Number of floors
    - Waterfront, view, condition, grade
    - Latitude and longitude

- **Satellite images** corresponding to each property, stored as PNG files.

- **Target variable:** Property price.

To stabilize training and reduce skewness, the target variable was transformed using: y=log(1+price)



# 3. Exploratory Data Analysis (EDA)

## 3.1 Price Distribution

The original price distribution is heavily right-skewed, with a long tail of high-value properties. After applying the log transformation, the distribution becomes more symmetric, which is more suitable for regression using neural networks.

## 3.2 Feature Relationships

Correlation analysis of tabular features shows:

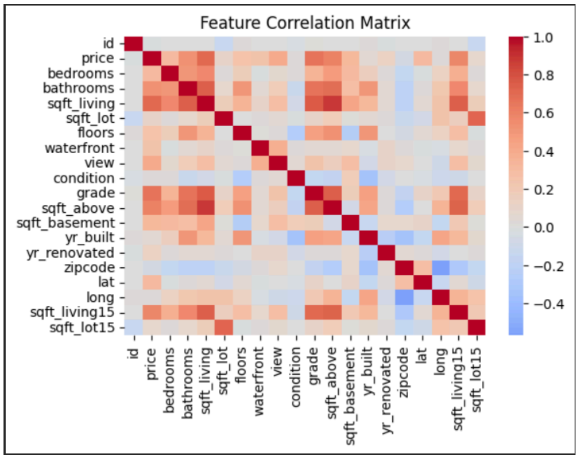
- Strong positive correlation between price and living area.
- Moderate correlation with grade, bathrooms, and view.
- Weaker correlation with geographical coordinates alone.

## 3.3 Satellite Image Inspection

Random satellite images reveal diverse visual patterns such as:

- Dense urban layouts
- Waterfront properties
- Green and suburban neighborhoods

These observations suggest that images potentially contain valuable contextual information, but the signal may be subtle and noisy.



mapbox
© Maxar © Mapbox © OpenStreetMap

mapbox
© Maxar © Mapbox © OpenStreetMap

# 4 Data Preprocessing

### Image Preprocessing

- Resize to 224 × 224
- Convert to tensor
- Normalize using ImageNet statistics
- Data augmentation (training only):
  - Random horizontal flip
  - Random rotation
  - Color jitter

Validation and test images use **only resizing and normalization** to ensure consistency.

### Tabular Preprocessing

- Standardization using `StandardScaler`

- The scaler is fitted on training data only and reused for validation and test sets

# 5. Model Architecture

### Image Encoder

- Backbone: Pretrained ResNet (ResNet50)
- Final classification layer removed
- Output embedding size: 2048
- Additional CNN head: Linear(2048 → 256) + SiLU + Dropout

### Tabular Encoder

A fully connected neural network:

- Input → 128 → 64 → 32
- SiLU activations
- Batch Normalization
- Dropout for regularization

### Feature Fusion

- Image and tabular embeddings are fused using concatenation
- A learnable parameter α\alphaα balances image and tabular contributions:

### Regression Head

- Fully connected layers:
    - (256 + 32 + 256) → 256 → 128 → 1
- Dropout applied to reduce overfitting
- Final output predicts log(price)

# 6. Training Strategy

### Loss Function

- Smooth L1 Loss (Huber Loss)
- Robust to outliers in price distribution

### Optimization

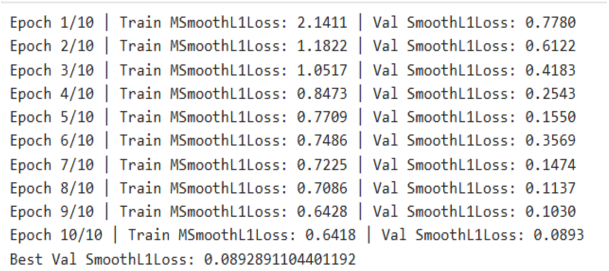
- Optimizer: Adam
- Initial learning rate: 1e-3
- Gradient clipping applied (max norm = 1.0)

### Learning Rate Scheduling

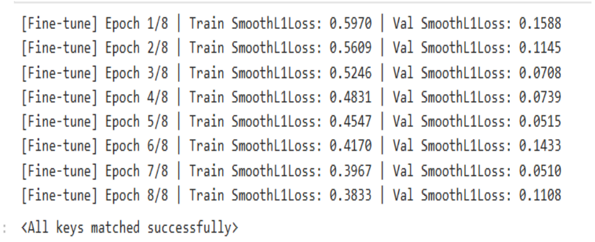
- ReduceLROnPlateau
- Learning rate reduced when validation loss plateaus

### Regularization

- Dropout in tabular encoder and regression head
- Weight initialization using Xavier initialization



```
Epoch 1/10 | Train MSmoothL1Loss: 2.1411 | Val SmoothL1Loss: 0.7780
Epoch 2/10 | Train MSmoothL1Loss: 1.1822 | Val SmoothL1Loss: 0.6122
Epoch 3/10 | Train MSmoothL1Loss: 1.0517 | Val SmoothL1Loss: 0.4183
Epoch 4/10 | Train MSmoothL1Loss: 0.8473 | Val SmoothL1Loss: 0.2543
Epoch 5/10 | Train MSmoothL1Loss: 0.7709 | Val SmoothL1Loss: 0.1550
Epoch 6/10 | Train MSmoothL1Loss: 0.7486 | Val SmoothL1Loss: 0.3569
Epoch 7/10 | Train MSmoothL1Loss: 0.7225 | Val SmoothL1Loss: 0.1474
Epoch 8/10 | Train MSmoothL1Loss: 0.7086 | Val SmoothL1Loss: 0.1137
Epoch 9/10 | Train MSmoothL1Loss: 0.6428 | Val SmoothL1Loss: 0.1030
Epoch 10/10 | Train MSmoothL1Loss: 0.6418 | Val SmoothL1Loss: 0.0893
Best Val SmoothL1Loss: 0.0892891104401192
```



```
[Fine-tune] Epoch 1/8 | Train SmoothL1Loss: 0.5970 | Val SmoothL1Loss: 0.1588
[Fine-tune] Epoch 2/8 | Train SmoothL1Loss: 0.5609 | Val SmoothL1Loss: 0.1145
[Fine-tune] Epoch 3/8 | Train SmoothL1Loss: 0.5246 | Val SmoothL1Loss: 0.0708
[Fine-tune] Epoch 4/8 | Train SmoothL1Loss: 0.4831 | Val SmoothL1Loss: 0.0739
[Fine-tune] Epoch 5/8 | Train SmoothL1Loss: 0.4547 | Val SmoothL1Loss: 0.0515
[Fine-tune] Epoch 6/8 | Train SmoothL1Loss: 0.4170 | Val SmoothL1Loss: 0.1433
[Fine-tune] Epoch 7/8 | Train SmoothL1Loss: 0.3967 | Val SmoothL1Loss: 0.0510
[Fine-tune] Epoch 8/8 | Train SmoothL1Loss: 0.3833 | Val SmoothL1Loss: 0.1108

: <All keys matched successfully>
```

# 7. Evaluation Metrics

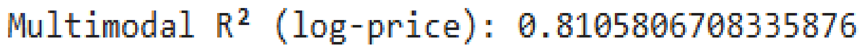
Models were evaluated using:

- **RMSE (Root Mean Squared Error)**
- **MAE (Mean Absolute Error)**

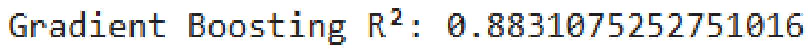
- **R² score**

Metrics were computed on:

- Log-price scale
- Original price scale (after inverse log transform)



```
Multimodal R² (log-price): 0.8105806708335876
```



```
Gradient Boosting R²: 0.8831075252751016
```

## 8. Explainability with Grad-CAM

To interpret model predictions, Grad-CAM was implemented manually (without OpenCV) on the final convolutional layer of the CNN. The heatmaps highlight regions in satellite images that influence price predictions, such as:

- Dense residential areas
- Proximity to roads
- Presence of greenery or open spaces

Grad-CAM visualizations varied across training runs due to:

- Random weight initialization
- Data augmentation
- Stochastic optimization

This behavior is expected and reflects different local minima learned by the model.

# 9. Analysis and Discussion

Although satellite images add contextual information, several factors explain why multimodal performance was lower:

1. **Weak visual signal:**
   Property price depends more on interior features than satellite-level visuals.
2. **Noisy image-to-price mapping:**
   Many properties share similar surroundings but differ greatly in price.
3. **Limited fine-tuning:**
   The CNN was pretrained on natural images, not satellite imagery.
4. **High tabular dominance:**
   Tabular features already explain most variance in price.

Importantly, **a lower multimodal $R^2$ is not an error**—it is a valid and insightful outcome demonstrating that not all additional modalities improve predictive power.

# 10. Conclusion

This project demonstrates an end-to-end multimodal learning pipeline for property valuation using satellite imagery and structured data. While satellite images did not improve performance over tabular features alone, the experiment highlights critical insights about modality relevance and data alignment.

The work emphasizes:

- Proper preprocessing and validation
- Robust training strategies
- Honest interpretation of results

Future improvements could include:

- Higher-resolution imagery
- Domain-specific CNN pretraining
- Attention-based fusion mechanisms
- Additional geospatial context