Two-Step Deep Learning Approach for Classifying Bridges using Street Imagery

Carmen Andrade von Hillebrandt
Stanford University
Civil and Environmental Engineering
candvon@stanford.edu

Abstract

This study presents a two-stage deep learning framework for classifying bridge structures according to the Hazus Earthquake Loss Estimation methodology using Google Street View imagery. The goal is to develop a image-based approach to infrastructure inventorying.

The first stage filters input images to retain those that clearly depict side views of bridges, which are essential for identifying structural characteristics. Several filtering strategies were tested, including K-means clustering on features extracted from pretrained ResNet models and direct scene classification using PlacesCNN. The most effective method combined ResNet18 pretrained on Places365 with K-means clustering.

The second stage classifies filtered images into one of eight Hazus structural types using a ResNet-50 model fine-tuned with class-weighted cross-entropy loss. Multiple optimizers and pretrained backbones were evaluated, with the best results achieved using filtered images and Places365 pretraining. Evaluation metrics included accuracy, macro-average F1, and weighted-average F1.

Results show that image filtering improves classification performance, especially on less frequent classes. Further analysis revealed that most classification errors occurred between classes that differed only by construction era. Grouping such classes led to substantial performance gains, suggesting that while structural information can be learned from imagery, age-based distinctions remain a challenge. The framework offers a step toward semi-automated bridge inventory generation for seismic risk assessments.

1. Introduction

An accurate assessment of regional bridge inventories is fundamental for transportation planning, seismic retrofit design, and post-earthquake recovery. Existing seismic loss estimation tools, such as the FEMA Hazus earthquake loss estimation method, rely on detailed infrastructure databases such as the US National Bridge Inventory (NBI) to assign fragility classes to bridges [6] [17]. The Hazus methodology classifies bridges mainly on material, design, and age. However, many regions around the world lack comprehensive inventories, limiting the applicability of fragility-based seismic risk assessments.

This study proposes an image-based classification system to assign Hazus structural classes using Google Street View imagery. By leveraging only geospatial coordinates, this approach enables the semi-automated creation of bridge inventories without requiring labor intensive, manual classification by engineers. For training and validation, bridge coordinates and corresponding structural classes are obtained from the NBI, and associated street-level images are retrieved using the Google Street View API. The system input includes street-level images taken from multiple angles. The output is the structural classification according to the Hazus methodology.

The classification task is performed using a two-stage deep learning framework that uses convolutional neural networks (CNN) with transfer learning. The first stage filters images to retain those with informative side views of the bridge, while the second classifies the filtered images into Hazus-defined bridge categories. The system thus converts raw street-level imagery into structural classification data that could be used for seismic risk modeling.

The baseline approach for this task involves direct classification of all retrieved images without a filtering stage. In contrast, the proposed two-stage method aims to both improve classification performance by focusing on the most informative images and increase efficiency by reducing the number of images that need to be processed by the more computationally intensive CNN classifier.

2. Related Work

Recent advancements in computer vision have enabled scalable infrastructure classification using street-level imagery, especially for buildings. Ogawa et al. [15] proposed a framework that combines street view imagery and GIS building footprint data to estimate building age and structural type, using state of the art DCNN and Vision Transform (ViT) architectures. Similarly, Kang et al. [13] trained CNNs to classify the functionality of individual building using façade images across North American cities. Similar to our work they performed outlier removal on their street view imagery using a pretrained CNN. An example that also uses California as their testbed, Iannelli and Dell'Acqua [11] used CNNs to estimate building floor counts from street imagery in California, bypassing traditional metrics like building height. These studies demonstrate the potential of using street-level imagery and modern deep learning techniques for large-scale, cost-effective classification in the built environment.

However, these efforts are largely focused on buildings, and street-level image-based classification of bridges remains underexplored. In contrast, most computer vision applications for bridges have focused on damage detection. For example, Deng et al. [4] introduced a pixel-level bridge damage detection model using an ASPP-based deep learning network tailored for identifying deterioration features like delamination and rebar exposure. Their work highlights the difficulty of collecting labeled datasets for damage detection and addresses class imbalance using a weighted Intersection over Union (IoU) loss function. Another example is Alfaro et al. [1], where an approach to identify cracks using a 3D model generated from photographs of an unmanned aerial vehicle (UAV) and the use of a convolutional neural network (CNN) was proposed. While effective for monitoring bridge condition, this line of research does not address structural classification or inventory generation.

This paper aims to fill that gap by applying a deep learning framework to classify bridge structural types using street-level imagery. Unlike prior work that focuses on infrastructure damage or is limited to buildings, our method targets the classification of bridge typologies from images, which supports broader risk modeling and infrastructure planning efforts.

2.1. Manual and Semi-Automated Inventory Approaches

Historically, infrastructure inventories rely heavily on manual input. Rozelle [16] adapted Hazus for global use by taking advantage of Humanitarian OpenStreetMap, a dynamic, crowdsourced geographic database. In the absence of labeled bridge images, some works have explored bridge prototyping using metadata. Cetiner [2] developed a semi-automated approach by fusing semantic and geometric information with relevant NBI data to generate structural models of bridges. This method, while powerful, still requires structural inputs not always available in developing regions. This proposed method builds on this idea by using transfer learning to extract structural information from images and use it for classification rather than modeling.

2.2. Remote Sensing and Multi-Source Imaging for Exposure Mapping

Several studies have explored scalable exposure mapping through satellite or aerial imagery. Wieland et al. [19] proposed a framework that integrates omnidirectional ground images and satellite remote sensing to rapidly estimate building typologies using SVM-based classification. While powerful, these methods often lack the the detail offered by street-level views and are rarely applied to bridges.

2.3. Machine Learning on NBI Data

The National Bridge Inventory (NBI) has been used as a dataset for several machine learning applications. Jooto and Lattanzi [12] combined NBI records with seismic intensity and cost data to predict potential bridge designs using decision trees, Bayesian networks, and SVMs Fiorillo and Nassif [7] applied five different supervised learning techniques (including random forests, neural networks, and gradient boosting) to estimate bridge element conditions. These approaches rely on structured tabular data rather than imagery. Cetiner [2] is a notable exception, integrating imagery with NBI-derived metadata to build detailed 3D bridge models. However, their work focuses on structural modeling, which is more complex than the classification approach needed for regional seismic assessments.

3. Dataset

The dataset used in this project was obtained using the coordinates given by NBI and the bridge's respective Hazus class. The following sections describe the data collection process and the first stage of classification, data filtering.

3.1. Data Collection

The dataset used in this project was obtained using the coordinates given by NBI and the bridge's respective Hazus class. The first step in data collection was deciding which bridges were going to be part of the classification study. For this project, data was filtered to California bridges and those above roads. California bridges were chosen due to having a different seismic design to the rest of United States [6]. Only bridges above roads were chosen due these bridges being the only ones where it's possible to obtain imagery of the side profile, which contains the structural characteristics needed to classify the bridges.

The next step was deciding which Hazus classes were going to be included in the study. There are 23 California bridge classes in the Hazus methodology, which mainly differentiate bridges based on material, structural, and age. However, once bridges are filtered to over highways this number drops to 21 bridge classes. After analyzing the observances of each of the 21 bridge classes, it was decided to only focus on bridge classes that had more than 250 occurrences, to make sure that each class had enough data to have proper training, validation, and test samples. This lowered the total number of classes to 8. As can be seen in Figure 1, the data is unbalanced, with the number of samples going from 250 to more than 2000. To address this class imbalance, a weighted cross-entropy loss function was employed in the classification step. Accuracy metrics that take into account the unbalanced nature of the dataset were also used to address this issue [18].

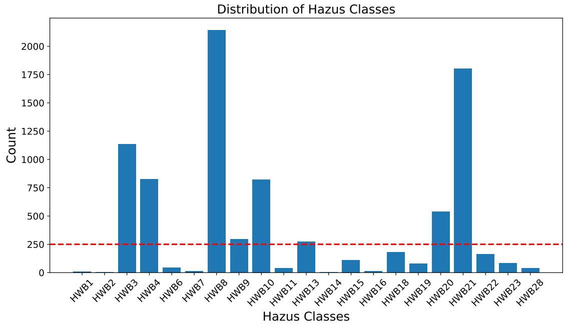


Figure 1: Distribution of Hazus classes in the unfiltered training dataset

To address the limitation of having only a single coordinate per bridge, four images are captured from different angles (North, East, South, and West) at each location to provide a more comprehensive visual representation of the bridge's structure and surroundings. Two examples of this can be seen in Figure 2. This capturing technique led to 25,469 images before data filtering.

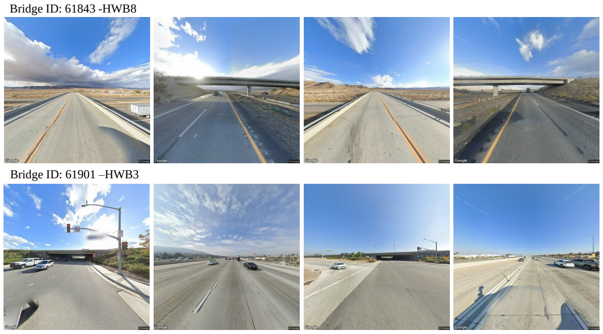


Figure 2: Street View Images of Bridges 61843 and 61901 taken from North, East , South, and West

3.2. Data Filtering Methodology

To determine which Street View images best captured the side profile of each bridge, four classification methods were tested. These methods served to filter out images that did not contain sufficient structural information and helped select the most informative views for the dataset without the need for labeling. Although most of the images were not labeled, 590 images were hand labeled to aid with validation. This corresponds to approximately two percent of the dataset. Three of the filtering approaches take advantage of the ResNet framework [8], which will be described in more detail in the Hazus Classification Method.

Approach 1: Baseline ResNet18 Trained on ImageNet

A standard ResNet18 model pretrained on ImageNet served as the baseline [3]. While this model was not specifically trained on structural or infrastructure-related classes, it provided a baseline for identifying images likely to include large man-made structures. The model was used to extract features that were then run through a K-means clustering algorithm to decide if the image contains a bridge side view image or not.

K-means clustering aims to partition a set of n data points $\{x_1, x_2, \ldots, x_n\}$ into K distinct clusters $\{C_1, C_2, \ldots, C_K\}$ such that the within-cluster variance is minimized. Formally, the optimization problem is:

$$\min_{C_1,\ldots,C_K} \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \mu_k\|^2 \tag{1}$$

where μ_k is the centroid of cluster C_k and $\|\cdot\|$ denotes the Euclidean norm. The algorithm iteratively alternates between assigning points to the nearest centroid and updating each centroid as the mean of the assigned points, until convergence [10].

Approach 2: ResNet18 Trained on Places365

The second approach takes advantage of the Places365 dataset that is specifically designed for scene recognition and includes bridges as one of its classes [20]. By using a model trained on the Places365 dataset, the feature extracted should be more meaningful. Like the baseline approach the pre-trained ResNet18 model is used to extract features and K-means clustering is used to decide which images contain side profile.The training and testing code used throughout this paper were based on the scripts created as part of the Places365 project [1].

Approach 3: ResNet50 Trained on Places365

The third approach follows the same workflow as the second approach, but takes advantage of ResNet50, a larger model that could potential predict side-views with higher accuracy. This higher accuracy however would come with higher computational cost.

[1] https://github.com/CSAILVision/places365/blob/master/run_placesCNN_basic.py

Approach 4: Direct Scene Classification with PlacesCNN

The fourth approach uses the wideresnet18 model trained on the Places365 dataset to directly classify scene types. This method allowed for precise identification of scene categories such as “bridge” or “viaduct.” Only images with high confidence in bridge-related categories were retained. This approach is like that used by Kang et al. to manage the uncontrolled quality of street images for building classification [13]. This method has as an advantage that it is a direct and interpretable way of dividing data that does not use clustering.

3.3. Data Filtering Results

Four approaches were used for data filtering in this study to identify side-view bridge images. Three of the approaches used K-means clustering, while the fourth approach used direct scene classification. To compare the efficacy of the four approaches, 590 images were hand labeled. To assure that these would be as representative as possible of the entire dataset, Hazus class, image direction, and location of the bridge were taken into account when choosing the bridges to hand label.

Accuracy was chosen as the primary evaluation metric, as the classification task is approximately balanced. In this dataset, each bridge typically includes images from four directions (North, South, East, West), and—assuming no major occlusions—either the North–South or East–West directions are expected to contain side views. As a result, about 50% of the images should be labeled as side views. Accuracy can be defined as the ratio of correctly predicted labels to the total number of images in the test set:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{Total Number of Samples}} \tag{2}$$

As shown in Table 1, the K-means classifiers outperform the direct scene classification approach by at least 8%. Among the K-means methods, those using Places365 features perform better than the method using ImageNet features. The ResNet18 and ResNet50 variants yield similar accuracy, though ResNet50 incurs a higher computational cost. For this reason, the ResNet18 (Places365) configuration was selected for downstream use.

Table 1: **Accuracy comparison of side-view filtering methods**

Method	Accuracy (%)
(1) Baseline ResNet18 (ImageNet)	79
(2) ResNet18 (Places365)	**84**
(3) ResNet50 (Places365)	83
(4) Direct Scene Classification	71

To further evaluate classification performance, confusion matrices are shown in 3 . The confusion matrix helps identify the distribution of different types of errors:

- **True Positive (TP)**: Correctly identified side view
- **True Negative (TN)**: Correctly identified non-side view
- **False Positive (FP)**: Missed non-side view
- **False Negative (FN)**: Missed side view

In the context of this study, false negatives are more problematic than false positives, as they reduce the number of valid samples for the next classification step. Maximizing true positives is also important, as it increases the pool of useful data. Based on both metrics, the classifier using Places365 with ResNet18 demonstrates the best performance and was selected for the Hazus classification task.

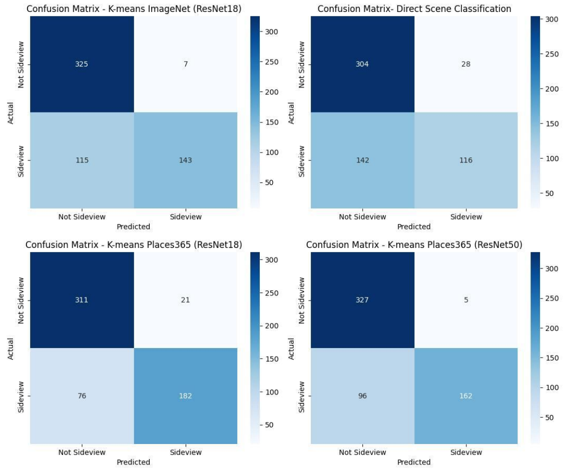


Figure 3: Confusion Matrices for the four data filtering techniques

A t-SNE plot of the selected filtering method can be seen in Figure 4. A t-SNE plot is a way to visualize high-dimensional data into 2D space, while preserving the local structure of the data. The plot shows that the chosen method has effectively group images into semantically meaningful clusters in the feature space.

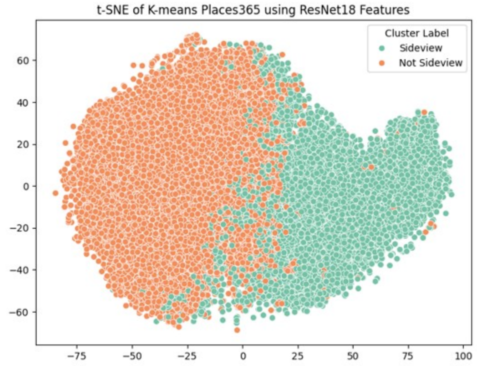


Figure 4: t-SNE of K-means Places365 using ResNet18 features

Figure 5 shows the distribution of Hazus classes after filtering. Although filtering dropped the number of samples for both "HWB13" and "HWB9" to below 250, these classes were kept for further classification.

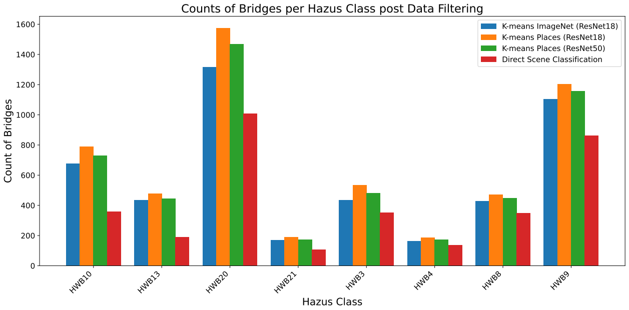


Figure 5: Confusion Matrices for the four data filtering techniques

4. Hazus Classification Methodology

The goal of this classification task is to assign a Hazus structural class to each bridge based on images taken from Google Street View. This is treated as an 8-way image classification problem, where each input is an image (ideally a side view) of a bridge, and the output is one of the eight Hazus-defined structural classes. For the purpose of this study, the dataset was dividing into three parts: 70% for training, 15% for validation, and 15% for testing.

4.1. ResNet-50 Architecture

A ResNet-50 convolutional neural network is used as the backbone for the classifier. ResNet50 is a deep convolutional neural network (CNN) architecture that is a variant of the popular ResNet architecture. The architecture of ResNet50 is divided into four sections: the convolutional layers, the identity block, the convolutional block, and the fully connected layers. The convolutional layers are used to extract features from the input image, while the identity and convolutional block are used to process and transform the features. As the last part of the network, the fully connected layer is used to make the final classification. Figure 6 gives an overview of the ResNet50 Architecture

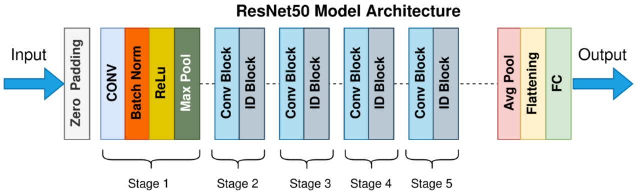


Figure 6: ResNet-50 Architecture
`https://commons.wikimedia.org/wiki/File:ResNet50.png`

In the context of this project, a pretrained ResNet-50 with weights from either ImageNet and Places365 were used. The original fully connected classification head was replaced with a new linear layer that outputs logits for the 8 Hazus classes. Only the final ResNet block and the new classification layer were unfrozen and fine-tuned during training.

The training and testing of the ResNet model were built on top of the Places365 demo and training codebase [20]. However, changes were required to accommodate the problem structure. In particular, the validation and test logic had to be changed to handle multiple images per bridge and evaluate performance at the bridge level. This change was not done in the training stage, to increase the number of instances that the model could learn from. This change allowed for a more realistic testing scenario, where predictions would most likely be made for a bridge using multiple viewpoints.

4.2. Loss Function

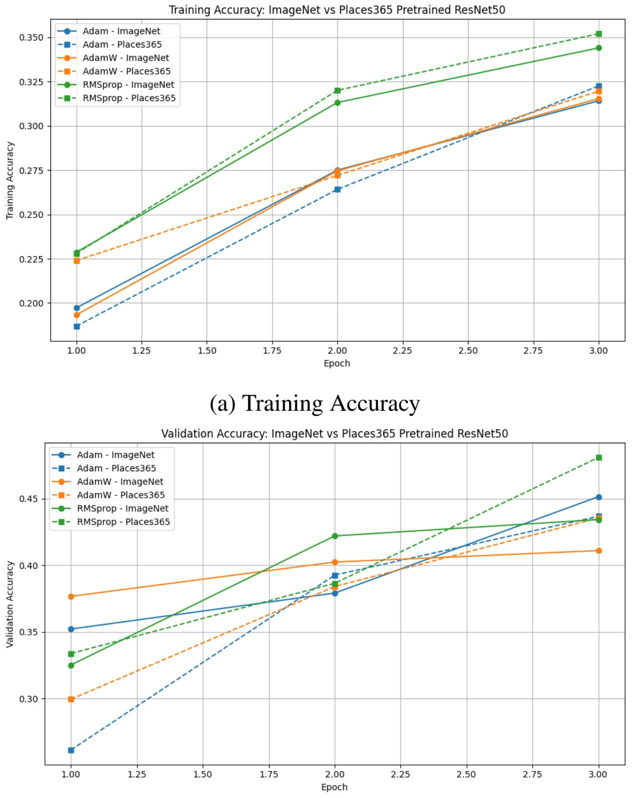
To account for the class imbalance in the data cross-entropy loss with class weighting was used:

$$\mathcal{L} = \sum_{i=1}^{N} -w_{y_i} \log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right) \tag{3}$$

where f_j is the j-th element of the vector of class score f, f_{y_i} is the score for true class y_i, and w_{y_i} is the class weight to address imbalance. In this case the weights were set to be inversely proportional to the frequency of each class in the filtered training set.

4.3. Optimization

Three different optimizer were compared in this study: Adam [14], AdamW [5], and RMSprop [9]. As shown in Figure 7, RMSprop achieved the highest training and validation accuracy after three epochs and was selected as the final optimizer. RMSprop updates parameters through an



(a) Training Accuracy

(b) Validation Accuracy

Figure 7: Effect of optimizer in training and validation accuracy

adaptive learning rate that is based on the magnitude of recent gradients. It uses a moving average of squared gradients to normalize updates, which helps stabilize training. Weight decay was also tested but did not yield significant improvements and was therefore excluded.

4.4. Data Augmentation

To improve generalization and reflect realistic variations in street-level imagery, a series of data augmentation techniques were used during training through the torchvision.transforms module. These transformation were chosen reflect realistic variations that could occur when capturing a bridge image from different lighting conditions, distances, or angles. The augmentations included in this study were:

- **RandomResizedCrop**: Crops random portion of image
- **RandomHorizontalFlip**: Horizontally flips images
- **ColorJitter**: Randomly adjusts brightness, contrast, saturation, and hue

4.5. Bridge-Level Prediction Aggregation

While training was conducted at the image level, evaluation during validation and testing was performed at the bridge level. Each bridge typically had multiple images from different angles. To generate a single predicted class for a bridge, the mean of the softmax outputs of all the associated images was taken and then the class with the highest average probability was selected. This method proved to be the most consistent among other tested, such as majority voting and choosing the highest probability over all classes and images.

5. Hazus Classification Results

The results of the Hazus bridge classification models are presented in three parts: (1) training and validation performance across model configurations, (2) test set performance broken down by individual Hazus classes, and (3) test set performance based on grouped structural classes where age is not used as a distinguishing feature.

5.1. Training and Validation Performance

The three Hazus Classification models tested in this study were: the baseline using pretrained weights from Imagenet and no filtering of images, filtered image dataset using pretrained weights from Imagenet, and filtered image dataset using pretrained weights from Places. Each model was trained for 60 epochs. The training and validation accuracy at each epoch can be seen in figure 8.

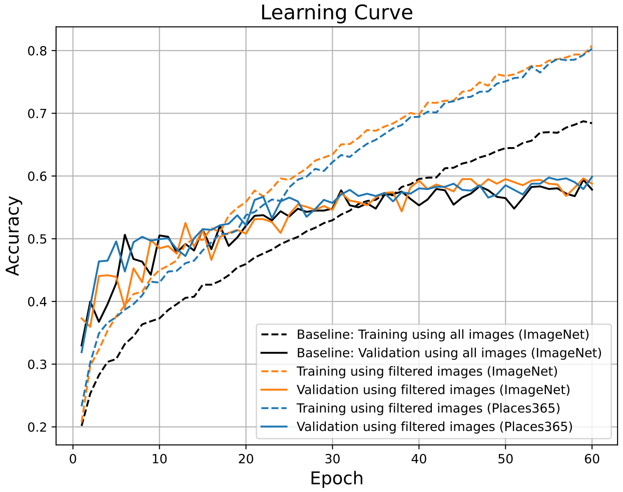


Figure 8: Learning Curves for Training and Validation

All three models achieved similar maximum validation accuracies of approximately 60%. The baseline model showed lower training accuracy compared to the filtered models, likely due to the inclusion of non-informative images that obscure key structural features. Both filtered models (ImageNet and Places365 pretrained) had comparable training performance, suggesting that filtering alone significantly improves the quality of the training data. However, the noticeable gap between training and validation accuracy in all cases suggests that the models are overfitting to the training set. Additional regularization strategies or larger datasets may be needed to close this gap and improve generalization.

5.2. Test Performance by Individual Hazus Class

Model performance on the test set is evaluated using three metrics: overall accuracy, macro-average F1-score, and weighted-average F1-score. While accuracy is commonly used, it can be misleading in the presence of class imbalance, as it may be disproportionately influenced by the performance of the most frequent classes.

To better assess the model's performance across all Hazus classes, the macro-average and weighted average F1-score are reported. F1 score can be defined as:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

Where

- **Precision** = $\frac{\text{TP}}{\text{TP}+\text{FP}}$
- **Recall** = $\frac{\text{TP}}{\text{TP}+\text{FN}}$

The macro-average F1 treats all classes equally by averaging the F1-scores of each class, regardless of how many samples each class has. In contrast, the weighted-average F1 weights each class's F1-score by the number of true instances, providing a more representative picture of overall performance on the imbalanced dataset.

Table 2 summarizes the test performance of three model configurations: the baseline model trained on all images (no filtering), and two filtered models pretrained on ImageNet and Places365 respectively. Although the baseline model achieves the highest accuracy (0.603), it underperforms in macro and weighted F1-scores, suggesting it is biased toward majority classes. In contrast, the filtered model using Places365 features achieves the highest macro-average F1 (0.535), indicating more balanced performance across all Hazus structural types.However, the Baseline does have higher performance metrics than the model trained on filtered images and ImageNet, which can indicate that ImageNet does not work as well in this context.

Table 2: Test set performance metrics across different model configurations

Metric	**Baseline**	**ImageNet**	**Places365**
Accuracy (%)	60.3	57.7	60.2
Macro F1 (%)	51.4	49.2	53.5
Weighted F1 (%)	58.9	56.3	59.3

The confusion matrices of the Baseline model and the filtered image model (Places365) are shown in Figure 9. The results are generally comparable, although the Baseline model was evaluated on a slightly larger test set due to the lack of image filtering.

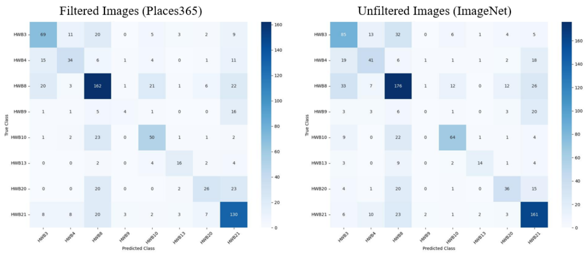


Figure 9: Confusion Matrix for Baseline (ImageNet) and Filtered Images (Places365)

The confusion matrices however do reveal recurring misclassifications between certain pairs of classes. Notably, these errors occur most frequently between classes that are structurally and materially similar, and differ primarily in terms of construction era. This observation motivates the next section, in which we investigate model performance when such age-based distinctions are removed and similar structural types are grouped together.

5.3. Test Performance by Grouped Hazus Classes

In this section, we evaluate model performance after grouping Hazus classes by structural type, ignoring differences in construction year. This reflects the intuition that the models may be able to differ between design and material type, but cannot differentiate between structures that only differ by age. Table 3 presents the test performance of each model under this grouped classification scheme.

Table 3: Test set performance metrics across different model configurations using class grouping

Metric	**Baseline**	**ImageNet**	**Places365**
Accuracy (%)	69.9	71.2	73.4
Macro F1 (%)	66.0	68.2	69.8
Weighted F1 (%)	69.8	71.9	73.3

All three models show improvements across all metrics after grouping, with the most significant gains observed for the filtered ImageNet model. The Baseline model shows the smallest relative improvement, which may indicate a lower ability to distinguish structural forms. These results suggest that while all models struggle with distinguishing construction eras, the filtered models are better at recognizing structural form.

Figure 10 shows the confusion matrices for the grouped classification task. The filtered Places365 model yields a matrix with clearer diagonal dominance and fewer off-diagonal errors, indicating more consistent and accurate predictions across all structural categories.

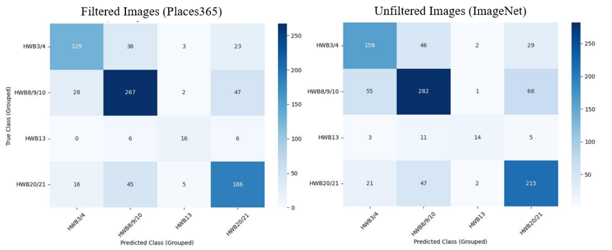


Figure 10: Confusion Matrix for Baseline (ImageNet) and Filtered Images (Places365)

6. Conclusion

This study developed a two-stage deep learning framework to classify bridge structural types defined by the Hazus Earthquake Loss Estimation methodology using Google Street View imagery. The proposed system addresses the lack of labeled infrastructure datasets by filtering raw image collections to retain only side-view bridge images and then classifying them using a fine-tuned ResNet-50 model.

Among the methods evaluated, the combination of image filtering and ResNet-50 pretrained on Places365 achieved the most balanced performance across Hazus classes, as measured by macro and weighted F1-scores. While the baseline model trained on unfiltered images achieved slightly higher raw accuracy, its lower F1-scores suggest it relied more on class priors than structural features.

Confusion matrices revealed that most misclassifications occurred between classes that are structurally similar and differ primarily by construction year. Grouping such classes showed a significant performance boost across all models. This supports the hypothesis that while the model can reliably learn visual cues associated with structural form and material, it struggles to distinguish age-based variants that may not be visually distinct.

For future work, expanding the dataset would be the most significant improvement. The current pipeline was limited by the number of images retrievable from Google Street View, both in terms of the total number of bridges and of usable views per bridge. Increasing the dataset size would likely improve model generalization and robustness. It would also be valuable to test this method in other regions, especially outside of California or the United States, where bridge designs and construction practices may differ significantly. Such cross-regional testing would help assess the robustness of the model. Finally, a hybrid approach that combines visual features with easily obtainable metadata could further improve classification performance.

7. Acknowledgments

Andrade was the sole contributor to this work. Public code used in this study:

- `https://github.com/CSAILVision/places365/blob/master/run_placesCNN_basic.py`
- `https://github.com/CSAILVision/places365/blob/master/train_placesCNN.py`
- `https://docs.pytorch.org/tutorials/beginner/transfer_learning_tutorial.html`

References

[1] M. C. Alfaro, R. S. Vidal, R. M. Delgadillo, L. Moya, and J. R. Casas. Structural damage detection using an unmanned aerial vehicle-based 3d model and deep learning on a reinforced concrete arch bridge. *Infrastructures*, 10(2), 2025.

[2] B. Cetiner. *Image-Based Modeling of Bridges and Its Applications to Evaluating Resiliency of Transportation Networks*. PhD thesis, University of California, Los Angeles, 2020.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[4] W. Deng, Y. Mou, T. Kashiwa, S. Escalera, K. Nagai, K. Nakayama, Y. Matsuo, and H. Prendinger. Vision based pixel-level bridge structural damage detection using a link aspp network. *Automation in Construction*, 110:102973, 2020.

[5] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011.

[6] FEMA. Hazus – multi-hazard loss estimation methodology 5.1, earthquake model technical manual. Technical report, Federal Emergency Management Agency, 2022.

[7] G. Fiorillo and H. Nassif. Application of machine learning techniques for the analysis of national bridge inventory and bridge element data. *Transportation Research Record*, 2673(7):99–110, 2019.

[8] 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 (CVPR)*, June 2016.

[9] G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. Coursera: Neural Networks for Machine Learning, 2012. pp. 26–31.

[10] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.

[11] G. C. Iannelli and F. Dell'Acqua. Extensive exposure mapping in urban areas through deep analysis of street-level pictures for floor count determination. *Urban Science*, 1(2), 2017.

[12] A. Jootoo and D. Lattanzi. Bridge type classification: Supervised learning on a modified nbi data set. *Journal of Computing in Civil Engineering*, 31(6):04017063, 2017.

[13] J. Kang, M. Körner, Y. Wang, H. Taubenböck, and X. X. Zhu. Building instance classification using street view images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145:44–59, 2018. Deep Learning RS Data.

[14] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

[15] Y. Ogawa, C. Zhao, T. Oki, S. Chen, and Y. Sekimoto. Deep learning approach for classifying the built year and structure of individual buildings by automatically linking street view images and gis building data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:1740–1755, 2023.

[16] J. R. Rozelle. International adaptation of the hazus earthquake model using global exposure datasets. Master's thesis, University of Colorado, 2018.

[17] F. H. A. F. U.S. Department of Transportation (USDOT) and B. of Transportation Statistics (BTS). "national bridge inventory 2008-present[datasets]. Technical report, U.S. Department of Transportation, Federal Highway Administration, 2020.

[18] Z. Y. Y. S. W. Chen, K. Yang and C. Chen. A survey on imbalanced learning: latest research, applications and future directions. *Artificial Intelligence Review*, 57(6):137, 2024.

[19] M. Wieland, M. Pittore, S. Parolai, J. Zschau, B. Moldobekov, and U. Begaliev. Estimating building inventory for rapid seismic vulnerability assessment: Towards an integrated approach based on multi-source imaging. *Soil Dynamics and Earthquake Engineering*, 36:70–83, 2012.

[20] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.