Are you in a safe building?

Amgoth Pavan Kumar

1 Introduction

Earthquakes cause significant casualties and economic damage, heavily influenced by building structural integrity. Quickly identifying structural deficiencies is crucial for assessing seismic vulnerability, especially in multi-story buildings with varied stiffness. This project aims to develop machine learning models to classify buildings from Google Street View (GSV) images, enhancing seismic risk assessments and disaster preparedness. This automated approach promises more efficient and accurate earthquake exposure modelling for better risk management.

2 Data

For training, we have a total of 2,516 images categorized into five classes: Steel Buildings (A), Concrete (B), Masonry (C), Wooden Framed (D), and Steel with Panel (S). Each class exhibits distinct structural and material properties, and the distribution of image sizes across these categories varies, as detailed in the figure 1. For testing, a total of 478 images are available.

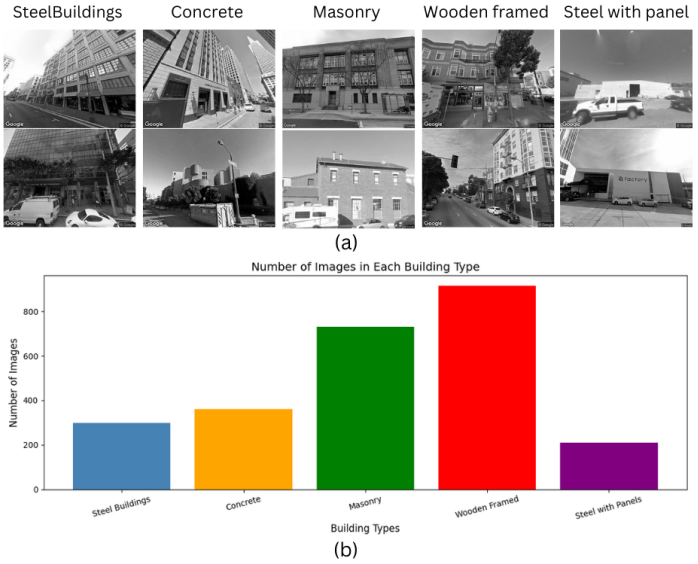


Figure 1: (a) Sample images of each class, (b) Distribution of images in each class

3 Models

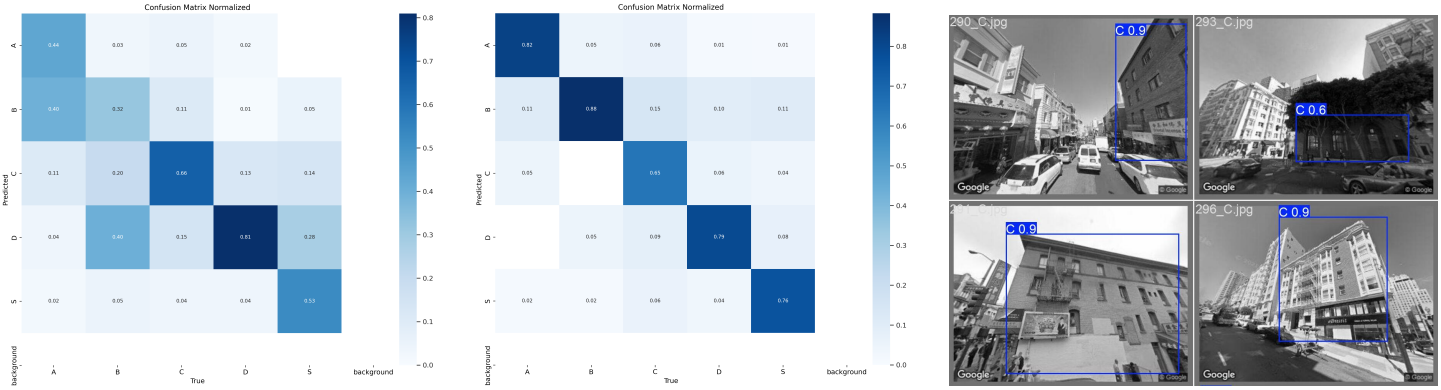
In this section, we will discuss the machine learning models employed in the project and the preprocessing techniques applied to the data. Various models were tested to determine the most effective approach for classifying building types based on GSV images. Each model was evaluated for its performance in terms of accuracy, precision, and generalization ability.

3.1 YOLO

We employed YOLOv8n-cls, a classification variant of the YOLO (You Only Look Once) family. YOLOv8n-cls is robust in complex scenes, excelling at identifying key features even in cluttered environments, making it crucial for accurately classifying dominant buildings amid other objects. Due to our dataset's limited size, we fine-tuned a pretrained YOLOv8n-cls model rather than training one from scratch. Fine-tuning allows us to utilize the model's transferable low-level feature detectors, such as edges and textures. Pretrained on extensive datasets like ImageNet, YOLOv8n-cls can be adapted for our specific task by refining its deeper layers. This strategy enables us to harness YOLOv8n-cls's advanced feature extraction while enhancing its performance for building classification.

We fine-tuned YOLOv8n-cls on our full dataset (training: 2013, validation: 503 images). We validated the model using top-1 accuracy (percentage of correctly classified images by the model's highest-confidence prediction), achieving 0.587.In figure 2 confusion matrix shows the classification performance. The highest value for class D (0.81) indicates better performance in identifying this class. Misclassification rates are evident between classes A and B, suggesting overlap in visual features. Class S showed moderate performance, with considerable misclassification(0.53).

Due to significant misclassification, we used a pretrained YOLOv8n model to detect and remove outliers[1]. After manual annotation and training, we refined the dataset to 1811 images. Detected images are shown in the figure 2. We retrained YOLOv8n-cls on the refined dataset, achieving a top-1 accuracy of 0.776. The updated confusion matrix shows improved accuracy, especially for classes A (0.82), B (0.88), and S (0.76). Confusion between classes, notably between B and D, has significantly reduced, indicating better model performance post-outlier removal.



(a) (b) (c)

Figure 2: (a) Confusion matrix for YOLO on the entire dataset. (b) Confusion matrix for YOLO on the preprocessed dataset. (c) Object detection using YOLO.

Results in figure 3 show decreasing training and validation losses, stabilizing around epoch 30. Top-1 accuracy improves rapidly, reaching stability around 0.77. Top-5 accuracy remains consistently high at 1.0, indicating strong model reliability post-outlier removal.

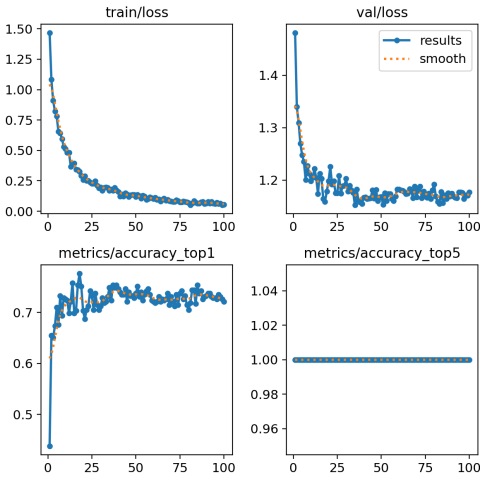


Figure 3: Results of YOLOv8n-cls after outlier removal

3.2 ResNet

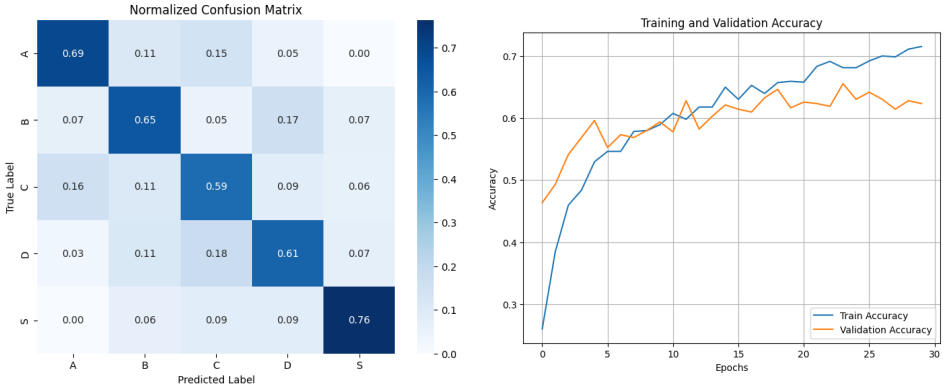
We chose ResNet50 pretrained on ImageNet for our task due to its proven effectiveness in feature extraction and deep learning efficiency. ResNet50's architecture, with residual connections, prevents vanishing gradient issues, making it well-suited for our complex building classification problem. Its pretrained weights offer robust low-level feature recognition, allowing us to quickly adapt the model to our relatively small dataset.

We trained ResNet50 on the dataset after outlier removal, achieving an accuracy of 0.655. Preprocessing included handling class imbalance by assigning computed class weights and augmenting the training data with rotation, shifting, shearing, zooming, and flipping to enhance generalization. The confusion matrix in figure 4 shows reasonable performance, with high accuracy for class S (0.76) but notable confusion between classes B, C, and D. The training and validation accuracy plot in figure 4 indicates steady improvement, with validation accuracy stabilizing around 0.62.

3.3 Swin Transformer

Swin Transformer, or Shifted Window Transformer, enhances traditional vision transformers by optimizing local and global feature extraction through a hierarchical design and shifted window mechanism [2].

As noted in Section 3.1, the dataset of 2,516 training images had issues like misclassifications, blurriness, and duplication, impacting accuracy. To address this, we prioritized object detection and data augmentation before classification.

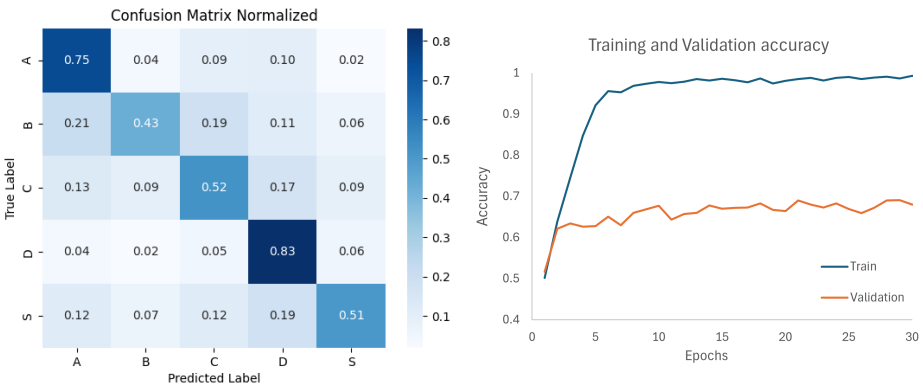


(a) (b)

Figure 4: (a) Confusion matrix of ResNet50 after outlier removal, (b) Training and validation loss for ResNet50 model.

In the object detection phase, 200 images per class were annotated for transfer learning, identifying 3,274 building instances (AP 50: 0.490) and 625 high-confidence instances in the test set (AP 50: 0.827).

Cropped images from detected instances created refined training and test datasets, each clearly labelled. Data augmentation ensured to prevent overfitting. SWIN Transformer with the refined datasets achieved a validation accuracy of 0.660 after 50 epochs, compared to 0.678 with the original images5. The confusion matrix shows the poor prediction between classes A & C, B & C, and B, C & S.



(a) (b)

Figure 5: (a) Confusion matrix of Swin transformer, (b) Training and validation accuracy of Swin transformer

Contrary to expectations, preprocessing did not improve validation accuracy, leading to the suspension of further multi-classification efforts. This result implies that the shifting window demonstrated its ability to extract local features without cropping the 'windows' in the preprocessing.

3.4 Other Models

Name	Validation Accuracy	Test Accuracy (Kaggle Private Score)
YOLOv8n-cls	0.776	**0.661**
ResNet50	0.655	0.598
Swin Transformer	0.660	0.548
VGG19	0.557	0.476
SVM with Oversampling	0.817	0.320
Random Forest with Oversampling	0.879	0.257

Table 1: Validation and Test Accuracy of Various Models

4 Conclusion

This study aimed to classify building types using models like YOLOv8n-cls, ResNet50, and Swin Transformers while addressing dataset issues such as misclassification, blurriness, and duplication through object detection and data augmentation. YOLOv8n-cls, ultimately, delivered the best performance, achieving a top-1 accuracy of 0.776 on validation and 0.661 on test dataset(kaggle private score) and securing 2nd rank on the leaderboard.

The success of YOLOv8n-cls can be attributed to its ability to effectively capture both global and local features through its object detection framework. Unlike ResNet50, which struggled with overfitting due to a lack of spatial awareness and complex architecture for the given dataset size, YOLOv8n-cls excels at distinguishing buildings even in cluttered environments. The built-in feature pyramid structure in YOLOv8n enabled it to retain critical spatial information, allowing for better context understanding and reducing the confusion between overlapping building types.

On the other hand, Swin Transformer did not show a significant improvement in performance despite the preprocessing steps. The shifting window mechanism already captured local features effectively, and cropping the images for preprocessing did not enhance the model's performance beyond what YOLO could achieve. YOLOv8n-cls's detection capability, paired with an efficient classification mechanism, enabled it to outperform other models by focusing on both accurate localization and feature extraction, providing the best balance between accuracy and computational efficiency. This made YOLOv8n-cls the optimal choice for our task of building classification.

References

[1] Jian Kang et al. "Building instance classification using street view images". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 145 (2018).

[2] Ze Liu et al. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), pp. 9992–10002.