WILEY

Instance segmentation of soft-story buildings from street-view images with semiautomatic annotation

Chaofeng Wang[1] | **Sascha Hornauer[2]** | **Stella X. Yu[3,4]** | **Frank McKenna[5]** | **Kincho H. Law[6]**

[1]M.E. Rinker, Sr. School of Construction Management, University of Florida, USA

[2]MINES Paristech, Ecole des Mines de Paris, Paris, Ile-de-France, France

[3]Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, Michigan, USA

[4]International Computer Science Institute, University of California, Berkeley, Berkeley, California, United States

[5]Department of Civil and Environmental Engineering, University of California, Berkeley, Berkeley, California, United States

[6]Department of Civil and Environmental Engineering, Stanford University, Stanford, California, United States

Correspondence
Chaofeng Wang, M.E. Rinker, Sr. School of Construction Management, University of Florida, Gainesville, FL, USA.
Email: chaofeng.wang@ufl.edu

Abstract

In high seismic risk regions, it is important for city managers and decision makers to create programs to mitigate the risk for buildings. For large cities and regions, a mitigation program relies on accurate information of building stocks, that is, a database of all buildings in the area and their potential structural defects, making them vulnerable to strong ground shaking. Structural defects and vulnerabilities could manifest via the building's appearance. One such example is the soft-story building—its vertical irregularity is often observable from the facade. This structural type can lead to severe damage or even collapse during moderate or severe earthquakes. Therefore, it is critical to screen large building stock to find these buildings and retrofit them. However, it is usually time-consuming to screen soft-story structures by conventional methods. To tackle this issue, we used full image classification to screen them out from street view images in our previous study. However, full image classification has difficulties locating buildings in an image, which leads to unreliable predictions. In this paper, we developed an automated pipeline in which we segment street view images to identify soft-story buildings. However, annotated data for this purpose is scarce. To tackle this issue, we compiled a dataset of street view images and present a strategy for annotating these images in a semi-automatic way. The annotated dataset is then used to train an instance segmentation model that can be used to detect all soft-story buildings from unseen images.

KEYWORDS
deep learning, instance segmentation, machine learning, rapid seismic screening, seismic risk, soft-story buildings, street view images

1 | INTRODUCTION

The aftermath of an earthquake may significantly impact a region on several levels, including casualties, damage to the housing stock and infrastructure, and severe economic losses. Even though it is difficult to predict earthquake occurrence, certain architectural constructs are known to be vulnerable to strong ground shaking, causing immediate damage to buildings and infrastructure.[1] It is usually economically feasible to perform pre-earthquake strengthening of such seismically vulnerable structures.[2] However, the screening and identification of these structures on a large scale has proved to be difficult. The screening process includes collecting data of the structural components of a building

(A) Observation of soft-story collapse during the Loma Prieta earthquake (J.K. Nakata, U.S. Geological Survey)
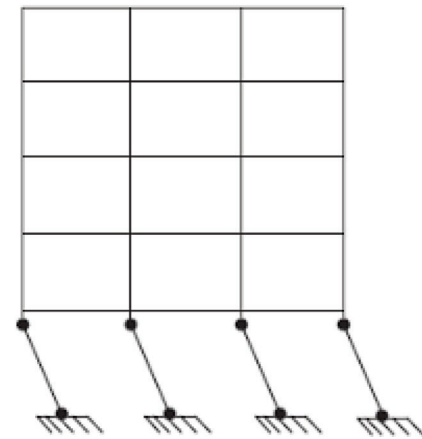
(B) Soft-story failure mechanism

FIGURE 1 Soft-story buildings are vulnerable to moderate and severe earthquakes. An observation of soft-story building collapse is shown in (A) and its failure mechanism is shown in (B)

and then conducting an evaluation of the structural integrity by a licensed professional engineer. A typical example is FEMA-154 Rapid Visual Screening of Buildings for Potential Seismic Hazards: A Handbook,[3] which was developed to provide a reliable methodology to (1) estimate the seismic safety of a large stock of buildings inexpensively and rapidly by their visual appearance, with minimum access to the inside of the buildings; and (2) determine if those buildings require a further detailed examination. Such visual screening procedures generally provide good results when identifying buildings for further risk investigation,[4] but they need to be conducted by experienced professionals.

One of the most seismically vulnerable building types that can be identified by such visual screening is the soft-story (SS) building, which is a common archetype with distinct visual characteristics whereby the first floor is not as stiff as the upper floors. This structural defect makes SS buildings vulnerable to both moderate and severe earthquakes (see Figure 1). SS building is very common in the United States while at the same time, many of its regions are situated in zones of active seismicity. As a result, a series of building reinforcements and mandatory retrofit projects have been launched since 2009,[5] aiming to reduce the structural deficiencies and to improve the performance of SS buildings during earthquakes. However, screening large inventories for SS buildings is challenging because the aforementioned FEMA method requires professional knowledge and is time consuming. Deep learning-based image analysis shows great potential for this purpose.

Deep learning techniques, for example, deep convolutional neural networks (CNNs), have been applied to image analysis and achieved impressive performance in various applications in civil engineering. Gao[6] used VGGNet to classify damages to structural components. Wang[7] used AlexNet and GoogLeNet to classify damages to masonry historic structures. Guo[8] used a meta learning-based CNN model to classify defects on building walls. Cha[9] used faster R-CNN for detecting cracks on the surface of concrete structural components. Czerniawski[10] used DeepLab for segmenting structural components from RGB-D images. Narazaki[11] used SegNet for bridge component recognition. There is a line of research investigating buildings using images taken from the street level. Such images can provide rich information and are inexpensive to obtain. Kang[12] tested several CNN architectures (AlexNet, VGG, and ResNet) for classifying street view images of different types of buildings (e.g., church, apartment, industrial buidling, museum, hospital, parking garage, hotel, etc.), with an accuracy of 70%–75%. Yu[13,14] used ResNet and Inception V3/V4 for classifying street view images of buildings into SS or non-SS categories. Gonzalez[15] compared five different models (VGG16, VGG19, InceptionV3, ResNet50, and Xception) for classifying images into concrete or masonry buildings, with accuracy scores ranging from 60 to 82%. Rueda-Plata[16] used VGG, InceptionV3, Xception, ResNet50 to classify images of masonry buildings into two classes: rigid or flexible roof diaphragm, with an accuracy of 80%. Pelizari[17] tested Xception, InceptionResNet-v2, and NasNet-A for classifying building street view images into Seismic Building Structural Types,[18] with an accuracy of 81%. Different from full image classification, Lenjani[19] used an object detection method (faster R-CNN) to detect buildings from 360 panoramas from both pre- and post-disaster street view images. The work focused on detecting the instances of buildings from the environment, without further analyses of the properties of the detected buildings, that is, classifying the detected buildings. The model resulted in an average precision (AP) of 85.47%. Kalfarisi[20] also used object detection method to identify buildings from street view images and classify the detected

(A) Classification: this is an image of a non-SS building

(B) Classification: this is an image of a SS building

(C) Classification: this is an image of a non-SS building
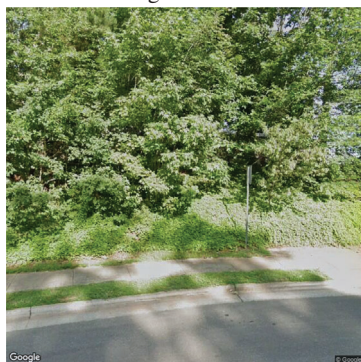
(D) Instance segmentation: there are neither SS nor non-SS buildings in the image

(E) Instance segmentation: there is a SS building in the image, and the location of the building is identified

(F) Instance segmentation: there are a SS and a non-SS building in the image, and the locations of the buildings are identified

FIGURE 2 Using full image classification and instance segmentation for inferring on street view images regarding buildings. Instance segmentation provides more accurate information.

instances as SS building and non-SS buildings, with an accuracy of 75.14% at high confidence level. Wang[21] used instance segmentation (mask R-CNN) to identify masonry building instances from street view images and classify them into two categories: reinforced or un-reinforced, with a mean average precision (mAP) 78.97% at intersection over union (IoU) ≥ 0.5. Many of these approaches demonstrated that it is feasible to use deep learning-based methods to identify buildings at risk from street view images. This is much more efficient compared to traditional visual screening methods.

In the aforementioned examples of extracting building information from street view images, most focused on full image classification.[19–21] used either object detection or instance segmentation, which might be a better choice than full image classification for the focal scenario of this paper, that is, finding SS buildings in the environment. We proposed to use full image classification in our previous study.[14] The full image classification model takes a given picture as input and returns the classification for determining whether the object of a specific class is displayed in the picture or not. Full image classification works well for images containing one and only one building, such as Figure 2B. For a random street view image, it is often that there is no building captured (Figure 2A) or there could be multiple buildings captured in one imaged, and they might belong to different classes (Figure 2C). In addition to buildings, there could be other objects present in an image. For example, in Figure 2B, there are two trees, two cars, a lawn, the sky, and the pavements; the building occupies only a small fraction of the image. A full image classifier takes all pixels into consideration, increasing the difficulty for training a model that should make inferences based on the features of buildings.

Object detection or instance segmentation can locate and classify buildings in street view images. It overcomes the aforementioned difficulties faced by full image classification: if there is no building, it gives no prediction, as shown in Figure 2D; if there are multiple buildings present, it will locate all buildings along with the category prediction of each building, such as shown in Figure 2F. Work done by Kalfarisi et al.[20] shows that object detection method is suitable for detecting SS buildings from street view images. However, there are difficulties to achieve high performance. Because street view images are captured by cameras mounted on moving vehicles, they might contain very complex scenes and noises.
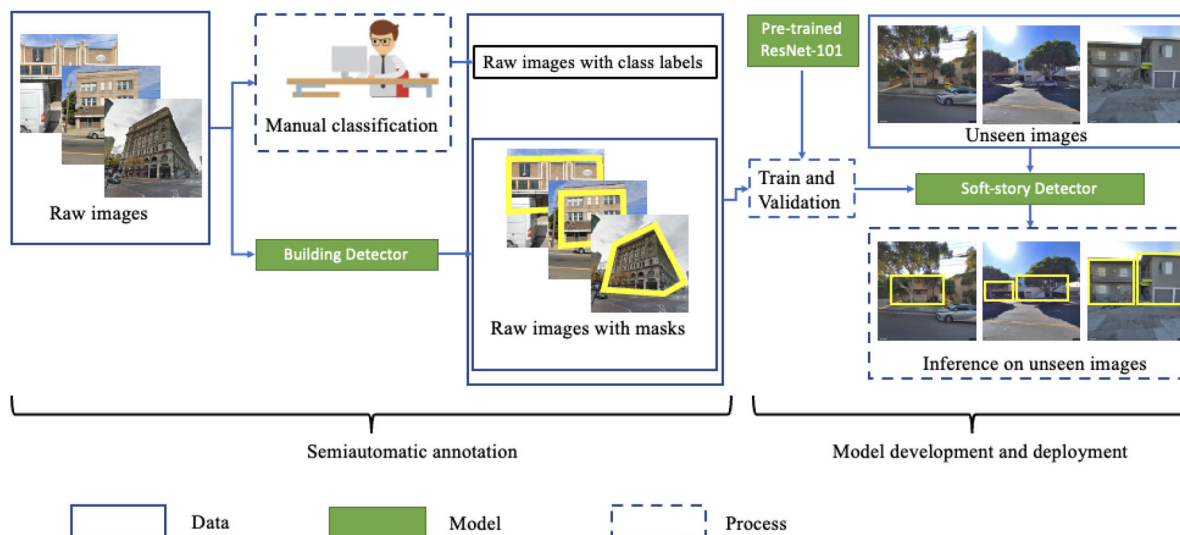
FIGURE 3 Overall workflow of the proposed method: the engineer work with the building detector to semiautomatically annotate the dataset, which is used for training the soft-story detector

Commonly, buildings in a street view image can be heavily occluded by vegetation. Furthermore, an image may not be able to capture a building because of an improper viewpoint. One way to improve the model performance is to increase the training data to cover variations as much as possible. However, annotating a large number of images (using bounding boxes or polygons) for training a detection or segmentation model can be tedious, labor-intensive, and time-consuming. To address this issue, a semi-automatic strategy to annotate the dataset was developed, based on a model pretrained on another large dataset.

In this study, we propose to use instance segmentation for better screening of SS buildings; we also propose a scalable semi-automatic strategy for creating the training data. The development of this SS building identification method included three tasks: (1) the collection of street images; (2) semi-automatic annotation of the collected images; and (3) the training of the identification model. The trained model can then be used for analyzing new street view images.

2 | METHODOLOGY

2.1 | Overall workflow

The objective of this research is to develop an instance segmentation model that is capable of identifying SS buildings from street view images with the minimum amount of human annotation. The complex contexts in street view images present a unique challenge. The model must be trained on a large dataset to facilitate identifying building objects and further classifying them into different categories. This study proposes dividing the process into two consecutive steps. The overall workflow is shown in Figure 3.

Step 1: Semi-automatic annotation of SS buildings

The first step is called the semi-automatic annotation. The objective of this step is to obtain training data for use in the next step, that is, to obtain the mask annotations of SS and SS building objects. First, an instance segmentation model is trained on a large dataset, in which buildings are annotated so that the trained model can identify buildings from an image. This model is called the *building detector*, as shown in Figure 3. An engineer will then work with the detector to annotate SS and non-SS buildings in a small dataset: The engineer is first given a raw street view image of a building and asked to classify the building as SS or non-SS. The image is then fed into the *building detector* to generate the mask on the building. Details of this semi-automatic annotation step can be found in Section 4.

Step 2: Training the SS building detector

In the second step, the annotated small dataset is then used for training the second model, which can not only detect building objects from an image but is able to classify them into SS/non-SS categories. This model is called the SS building detector or *SS detector*. The details of the second step are explained in Section 5.
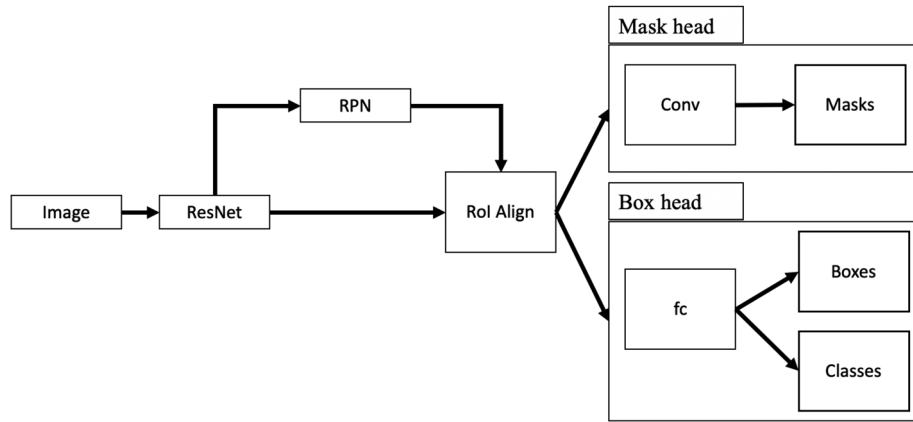
FIGURE 4 The architecture of Mask R-CNN[22]

2.2 | Models

The two models—the building detector and the SS detector—have the same architecture, which is based on the Mask R-CNN approach originally developed by He et al.[22] This approach is an extension of the faster R-CNN algorithm[23] by adding a branch for predicting segmentation masks on each region of interest (RoI) in parallel with the existing branch for classification and bounding box regression. The street view image is first input into a pretrained backbone CNN (ResNet with 101 layers), which is connected with a region proposal network (RPN) for proposing RoIs from the feature maps output by the backbone. The RoIs are then fed into two parallel branches: the mask branch and the bounding box branch. The mask branch is a small fully convolutional network (FCN) applied to each RoI, predicting a segmentation mask in the pixel level. The box branch is a set of fully connected (fc) layers that yield the class softmax and the bounding box regression. During the backpropagation, the loss is calculated for each RoI: $L = L_{cls} + L_{box} + L_{mask}$, where L is the total training loss; L_{cls} is the loss of the classification and L_{box} is the loss of bounding box; and L_{mask} is the loss of the mask. More details about the algorithm and the detailed definitions of L_{cls}, L_{box}, and L_{mask} are referred to Ren et al.[23] and He et al.[22] Figure 4 shows the architecture of the mask R-CNN.

3 | TRAINING DATA

Common objects in context (COCO) dataset[24] is one of the most popular datasets for image segmentation and object detection. The COCO dataset was created with the goal of gathering a large number of images of complex everyday scenes that contain common objects in their natural context. This dataset eventually gathered 328k images, in which 2.5 million instances were annotated. It consists of images covering 91 categories that can be easily recognized by human beings. A subset of this dataset was later annotated at the pixel-level, including 164K images covering 80 "thing" classes and 91 "stuff" classes. This subset is called the COCO-Stuff,[25] which is used to train the building detector.

The dataset for training the SS detector is created from a small dataset containing raw street view images. The procedure is described in the semi-automatic annotation Section 4. The raw images are downloaded using Google Static Street View API. These images are photographed from vehicles on the roads. Given the location of a target (i.e., a building), the API will calculate the heading direction to find the image that contains the object. A building could appear in several images from different angles, as shown Figure 5. The API returns the image taken at the closest point. We show street view examples of SS and non-SS buildings in Figure 6.

In addition to the location of the target, other parameters used for calling the API are "pitch" and "field of view" (FOV). Pitch is set to be 0ž, which means that the view of the camera is flat horizontal; FOV is the parameter that determines the horizontal FOV. For a fixed-size street view image, the FOV controls the zoom level. It has been experimentally determined that the targeted buildings can be captured in most images when FOV is around 60ž. Using the above parameters, a set of
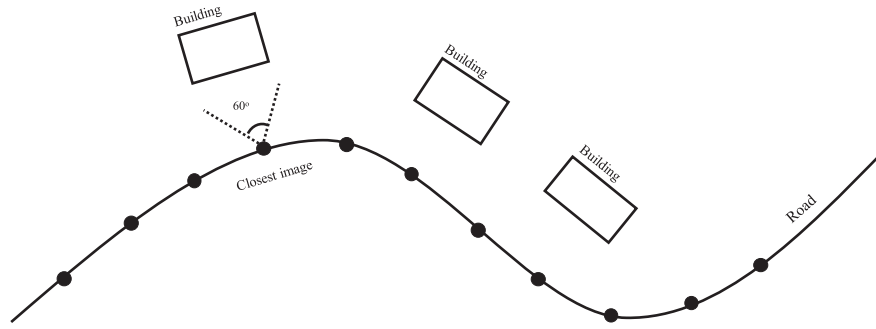
FIGURE 5 Street view image acquisition. The query returns the image taken at the point closest to the building.

(A)   (B)   (C)   (D)   (E)
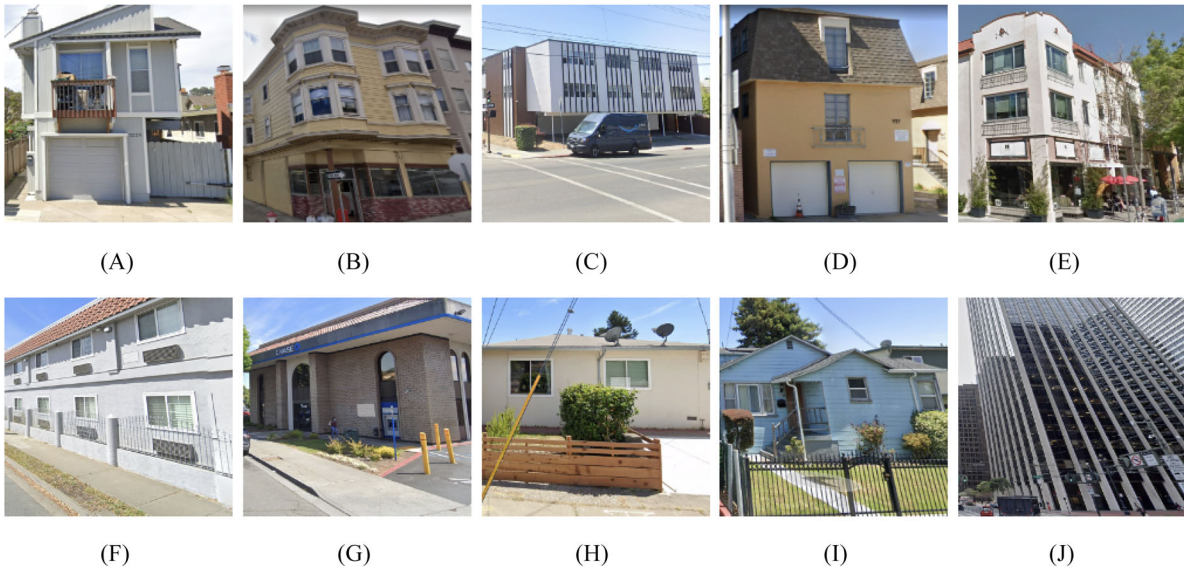
(F)   (G)   (H)   (I)   (J)

FIGURE 6 Examples of soft-story and non-soft-story buildings; A–E are soft-story; F–J are non-soft-story

raw street view images is downloaded. These images are used as the input to the semi-automatic annotation procedure, which is detailed in the next section.

4 | SEMI-AUTOMATIC ANNOTATION OF BUILDING INSTANCES

Annotating images is quite labor-intensive and time-consuming. In this study, a two-stage workflow that requires minimum human supervision is developed. It utilizes an available annotated dataset in order to semi-automatically annotate the building instances from street view images.

The procedure of the semi-automatic annotation is outlined in Figure 7. A set of raw images of the street view are first collected from Google Maps. (Note that these images do not belong to the COCO-Stuff dataset.) A trained structural engineer is then asked to classify these images into two categories: one contains SS buildings (566 images) while the other contains non-SS buildings (736 images). These images are then inputted into a segmentation model, that is, the building detector trained on the COCO-Stuff dataset, so that buildings can be identified and re-annotated: each identified building has a predicted mask (polygon) indicating its location in the image. We can assign a new class name (SS/non-SS) to the predicted mask since the image has been labeled previously with the class name by the engineer. Thus a dataset of street view images containing building objects limited to only two classes is created. Each building is annotated with a mask (converted into a polygon by contouring the mask) and a class label (SS/non-SS).
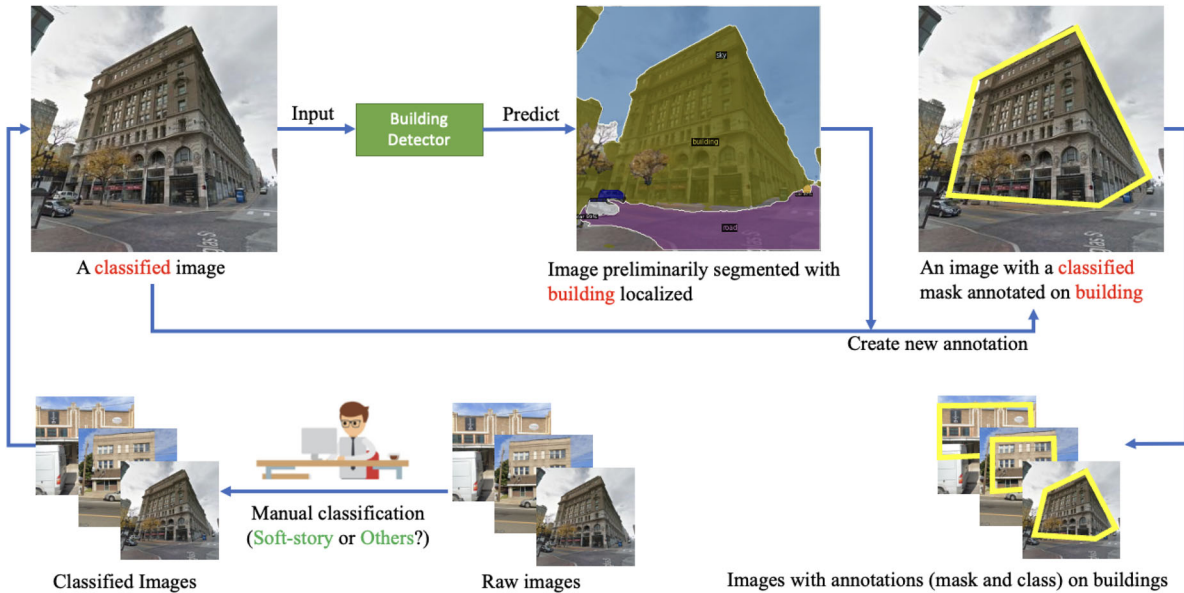
FIGURE 7 Semi-automatic annotating. The engineer classify images firstly. The classified images are input to the building detector, which generate masks for building instances. The generated masks and the engineer's classifications are combined to form the new annotation. This figure serves as a detailed illustration of the semiautomatic annotation procedure in Figure 3.

Once all the images in the small dataset are annotated with SS/non-SS polygons, the second step of the pipeline is to use them to train the SS detector. This is detailed in the next section.

5 | TRAINING THE SOFT-STORY DETECTOR

This section shows the training of the SS detector; note, the aforementioned dataset created using the semi-automatic annotation method can now be used for training the SS identification model.

The SS detector is also a mask R-CNN model, as described in Section 2.2. It has the same architecture as the building detector: the network uses a ResNet-101 conv5 backbone with dilations in conv5, and standard conv and fc heads for mask and box prediction, respectively. The only difference is that the previous building detector is trained on the COCO-Stuff dataset, while the SS detector is trained on the street view dataset annotated using the semi-automatic approach described in the previous section. The input of the model is a raw image. The output of the model is polygons, indicating the locations of the detected buildings and the corresponding classes (SS/non-SS) and bounding boxes.

The optimizer is stochastic gradient descent. The base learning rate of the training is set as 0.0025, which is based on the training on the COCO dataset. We selected the learning rate based on our experience of training models on the COCO dataset. The computational platform is FRONTERA at Texas Advanced Computing Center. We used one NVIDIA Quadro RTX 5000 GPU. The training took 25 h. The training metrics of this model are presented in Figure 8, which shows that the training converged after 30,000 iterations. The performance of a segmentation model can be evaluated by the AP of all classes at a particular IoU level, which is defined as the intersection of the prediction area and ground truth area divided by the union of prediction and ground truth areas. Another popular metric for measuring the performance of segmentation models is the mAP over a series of IoU levels. For this converged training, the mAP on the testing set is 0.795 and the mean IoU is 0.763, indicating a good and acceptable performance given that the size of the training data is small.

A few prediction examples are presented next, followed by a discussion on the performance. The first row of Figure 9 shows the example of a SS building: (A) the raw street view image; (B) the prediction; and (C) the ground truth. The second row shows the example of a non-SS building. The classes are predicted correctly, and the masks are predicted with high accuracy.

One common issue is that buildings are often located far away from the camera, occupying only a very small fraction of the street view image. Although this limitation can be a problem regarding image classification, it can be addressed by the instance segmentation model, as shown in Figure 10.
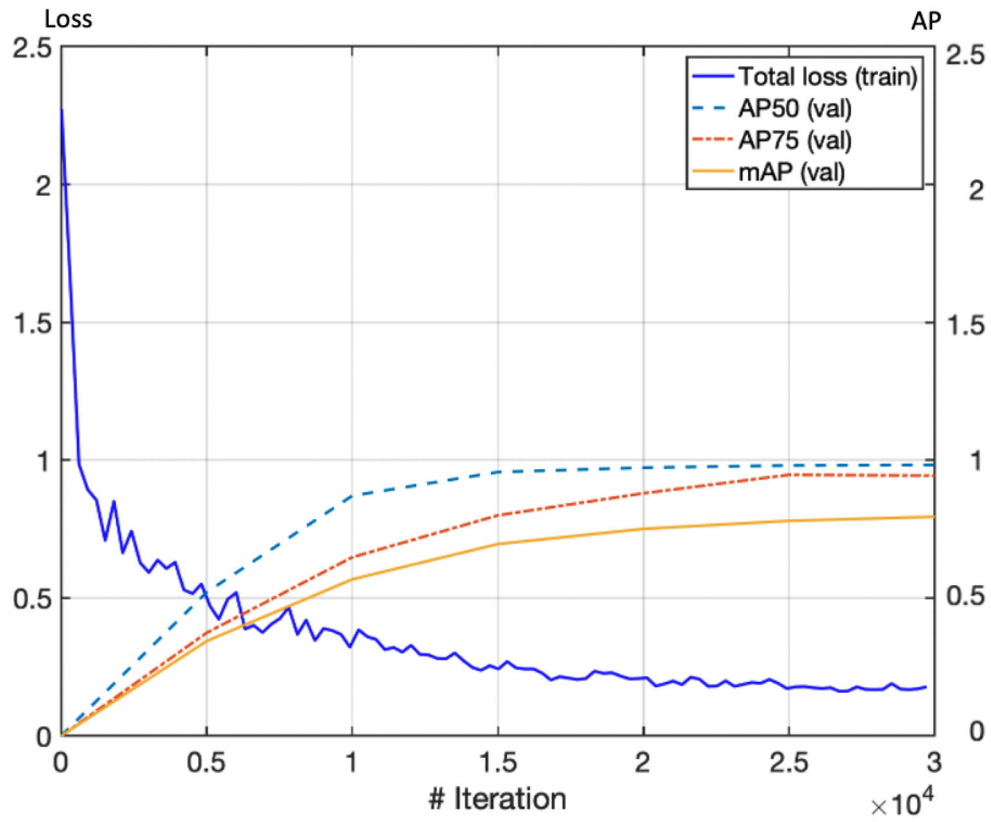
FIGURE 8 Total training loss and average precision on the validation set (AP50: average precision at IoU = 0.5; AP75: average precision at IoU = 0.75; mAP: mean average precision (IoU = 0.50:0.95))

(A) Raw Image　　　　　　　　(B) Instance segmentation　　　　　　　(C) Ground truth annotation

(D) Raw Image　　　　　　　　(E) Instance segmentation　　　　　　　(F) Ground truth annotation

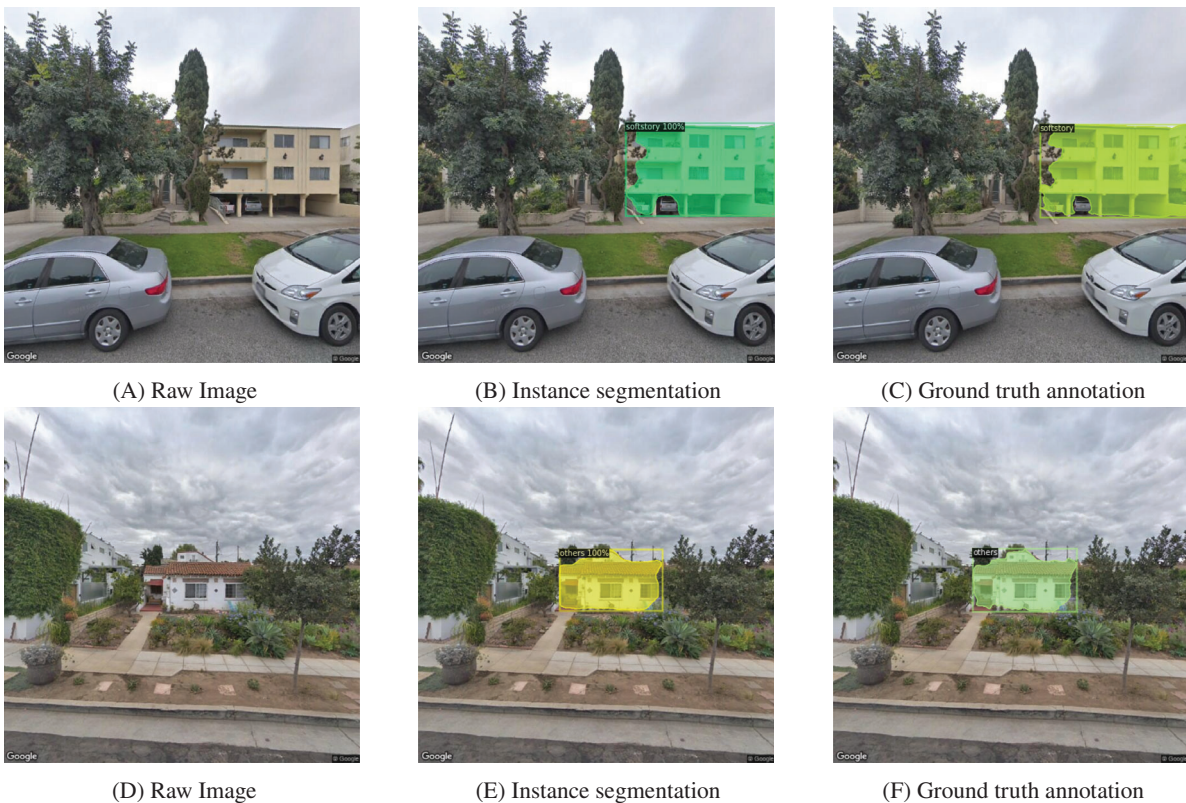**FIGURE 9** Segmentation examples of the soft-story detector (The first row is a SS building, the second row is a non-SS building.)

(A) Raw Image

(B) Instance segmentation

FIGURE 10 Small object example. The building far from the camera is detected and classified corrected.

(A) Raw image

(B) Instance segmentation

FIGURE 11 Example of heavy occlusions (trees and bushes occlude large parts of the building. The segmentation succeeds still in labeling a large coherent part of the building and the classifier predicts the correct label.)

Another problem often encountered is due to heavy occlusions, which are usually caused by trees. For example, the facade of the building in Figure 11A is occluded by several trees, leaving only a small part of it visible. As shown in Figure 11, the model can segment out this part and make the correct prediction of the class.

Shadows and bad lighting conditions are other conditions that may cause potential difficulties. Figure 12 shows an example of heavy shadows of the trees projected on the building facade. Though the segmentation is not perfect, it can still capture the building and correctly predict the class.

| (A) Raw image | (B) Instance segmentation |

FIGURE 12   Vegetation casts complex shadows onto the building. Even though this changes the building's appearance heavily, the approach succeeds in labeling sufficient pixel and the overall building class correctly, seen in B).

TABLE 1   Comparison of full image classification versus instance segmentation

	Method	Average acc.	Precision	Recall	F1
Previous study	Full image classification	87.72%	84.26%	92.39%	0.8814
This study	Instance segmentation	88.86%	84.61%	94.92%	0.8947

6 | MODEL TESTING AND APPLICATION

6.1 | Comparison with prior study

It is not suitable to perform a direct comparison of segmentation/detection and full image classification, since they are meant for different tasks. As shown in Figure 2, they should only be compared for specific scenarios. In the scenario of identifying SS buildings, the prerequisite for comparing their performances requires the building to occupy a major portion of the image, ideally centered in the image as examples shown in Figure 6. Therefore, we compare the performance of the model in this paper with an earlier study,[26] where the full image classification was used. In the comparison, we use the same benchmark data, with 395 street view images collected from two cities in northern California—Berkeley and San Jose, consisting 198 SS and 197 non-SS. The details of this benchmark can be found in Yu et al.[26] The result is shown in Table 1. Instance segmentation performs slightly better on this dataset than full image classification. It should be noted that one emphasis of this study on object detection/instance segmentation, which can provide explainable inference on images, and thus is more suitable for detecting interested features from wild street view scenes.

6.2 | Application

To demonstrate the application of the trained model, we used the trained model for detecting SS buildings in Central Berkeley, a neighborhood in California that is dense with population and buildings. A total of 1293 buildings were detected in street view images downloaded from Google Maps, where 168 of these buildings are SS. The detection result is merged with the building footprints for visualization in Figure 13. Similar to the Central Berkeley, there are many populated neighborhoods located in regions of high seismicity, where SS is a common construction type. The approach

FIGURE 13 SS building detection in the Central Berkeley neighborhood, California

demonstrated in this study provides a tool for rapid screening of this type of construction in a large region at very low costs.

7 | DISCUSSION

As mentioned in the literature review section, the technique of using machine learning for extracting useful information from street view images has been applied in engineering, economic, and sociological studies. To date, using this technique to obtain information for hazard mitigation starts gaining interest. The objective of this study is to develop a model, with minimum human annotation, for detecting SS buildings from street view images for assisting decision-making for earthquake preparedness. We demonstrated that with the proposed two-phase semiautomatic annotation strategy, a deep convolutional neural networks (mask R-CNN) can be trained to help detect SS buildings from street views. The performance of the trained model is good (with a mAP on a randomly selected testing data set of 79.5%), considering that the model is trained on a relatively small dataset that contained about only 1000 images. We expect that results can be improved even further by collecting more street view images for use in training.

Prior studies of applying DL to street view image classification have noted that the noises in the data pose difficulties. For the building recognition problem, the occlusions (e.g., vegetation and vehicles) blocking the image of the buildings are one of the major sources of noise. This is due to full image classification technique that train the model based on the whole image, without paying attention to the objects of interest in the image. Thus, not only the objects of the interest but also the noise are taken into consideration when deciding which class the image belongs to, requiring a huge amount of data to cover all the varieties in both the objects of the interest. This is labor-intensive and time consuming since all data must be labeled by hand.

Deleting occlusions from street view images is of significant importance for extracting building information; however, using full classification approach is problematic. Many of the existing literature uses traditional classification methods. This study builds on progress in the detection/segmentation domain for better building information acquisition from street view images and has demonstrated that this technique can overcome noise and occlusion problems.

8 | CONCLUSION

The present study aimed at developing a method for rapid identification of SS buildings from street view images. The method used is instance segmentation that is based on a deep CNN. Facing data scarcity, we developed a simple yet efficient strategy for annotating images in a semi-automatic way. We then annotated a dataset and trained a segmentation model, which performs the following tasks:

- Identify building instances from street view images
- Classify the identified instances of a building in said image into two categories: SS building and non-SS building.

When compared with the image classification method, one of the advantages of the present segmentation method is able to locate instances of buildings in an image. This leads to accurate classification of the building since the decision can be made on the building itself without interference from noise or occlusions in the image itself. This has significant implications for using machine learning to understand building attributes from images. The semiautomatic annotation strategy opens great opportunity for low-cost large-scale annotation to provide sufficient training data.

The presented approach provides an automated and inexpensive method for large-scale regional examinations that can benefit policymakers in determining seismic risk. Thought this method is not intended to replace more rigorous assessments of building performance using numerical simulations or statistical methods, it shows great potential for assisting the screening efforts.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID
Chaofeng Wang https://orcid.org/0000-0001-8534-9276
Sascha Hornauer https://orcid.org/0000-0003-0243-7647
Stella X. Yu https://orcid.org/0000-0002-3507-5761

REFERENCES
1. Deierlein GG, Zsarnóczay A. *State-of-art in Computational Simulation for Natural Hazards Engineering*. 2019:106-111.
2. Kappos AJ, Dimitrakopoulos E. Feasibility of pre-earthquake strengthening of buildings based on cost-benefit and life-cycle cost analysis, with the aid of fragility curves. *Nat Hazard*. 2008;45(1):33-54.
3. ATC. *Rapid Visual Screening of Buildings for Potential Seismic Hazards: A Handbook, FEMA 154*. Federal Emergency Management Agency; 1988.
4. Moseley V, Dritsos S, Kolaksis D. Pre-earthquake fuzzy logic and neural network based rapid visual screening of buildings. *Struct Eng Mech*. 2007;27(1):77-97.
5. Samant LD, Porter K, Cobeen K, et al. Mitigating San Francisco's soft-story building problem. In: Conference on Improving the Seismic Performance of Existing Buildings and Other Structures. 2010:1163-1174.
6. Gao Y, Mosalam KM. Deep transfer learning for image-based structural damage recognition. *Comput-Aided Civ Infrastruct Eng*. 2018;33(9):748-768.
7. Wang N, Zhao Q, Li S, Zhao X, Zhao P. Damage classification for masonry historic structures using convolutional neural networks based on still images. *Comput-Aided Civ Infrastruct Eng*. 2018;33(12):1073-1089.
8. Guo J, Wang Q, Li Y, Liu P. Façade defects classification from imbalanced dataset using meta learning-based convolutional neural network. *Comput-Aided Civ Infrastruct Eng*. 2020;35:1403-1418.
9. Cha YJ, Choi W, Suh G, Mahmoudkhani S, Büyüköztürk O. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Comput-Aided Civ Infrastruct Eng*. 2018;33(9):731-747.
10. Czerniawski T, Leite F. Automated segmentation of RGB-D images into a comprehensive set of building components using deep learning. *Adv Eng Inf*. 2020;45:101131.

11. Narazaki Y, Hoskere V, Hoang TA, Fujino Y, Sakurai A, Spencer Jr BF. Vision-based automated bridge component recognition with high-level scene consistency. *Comput-Aided Civ Infrastruct Eng*. 2020;35(5):465-482.

12. Kang J, Körner M, Wang Y, Taubenböck H, Zhu XX. Building instance classification using street view images. *ISPRS J Photogramm Remote Sens*. 2018;145:44-59.

13. Yu Q, Wang C, Cetiner B, et al. Building information modeling and classification by visual learning at a city scale. *arXiv preprint arXiv:1910.06391* 2019.

14. Yu Q, Wang C, McKenna F, et al. Rapid visual screening of soft-story buildings from street view images using deep learning classification. *Earthq Eng Eng Vib*. 2020;19(4):827-838.

15. Gonzalez D, Rueda-Plata D, Acevedo AB, et al. Automatic detection of building typology using deep learning methods on street level images. *Build Environ*. 2020;177:106805.

16. Rueda-Plata D, González D, Acevedo A, Duque J, Ramos-Pollán R. Use of deep learning models in street-level images to classify one-story unreinforced masonry buildings based on roof diaphragms. *Build Environ*. 2021;189:107517.

17. Pelizari PA, Geiß C, Aguirre P, Santa María H, Pena YM, Taubenböck H. Automated building characterization for seismic risk assessment using street-level imagery and deep learning. *ISPRS J Photogramm Remote Sens*. 2021;180:370-386.

18. Coburn A, Spence R. *Earthquake Protection*. John Wiley & Sons; 2003.

19. Lenjani A, Yeum CM, Dyke S, Bilionis I. Automated building image extraction from 360 panoramas for postdisaster evaluation. *Comput-Aided Civ Infrastruct Eng*. 2020;35(3):241-257.

20. Kalfarisi R, Hmosze M, Wu ZY. Detecting and geolocating city-scale soft-story buildings by deep machine learning for urban seismic resilience. *Nat Hazard Rev*. 2022;23(1):04021062.

21. Wang C, Antos SE, Triveno LM. Automatic detection of unreinforced masonry buildings from street view images using deep learning-based image segmentation. *Autom Constr*. 2021;132:103968.

22. He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: IEEE International Conference on Computer Vision. 2017:2961-2969.

23. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems. 2015.

24. Lin TY, Maire M, Belongie S, et al. *Microsoft COCO: Common Objects in Context*. Springer; 2014:740-755.

25. Caesar H, Uijlings J, Ferrari V. COCO-stuff: thing and stuff classes in context. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018:1209-1218.

26. Yu Q, Wang C, McKenna F, et al. Rapid visual screening of soft-story buildings from street view images using deep learning classification. *Earthq Eng Eng Vib*. 2020;19(4):827-838.

27. Rathje EM, Dawson C, Padgett JE, et al. DesignSafe: new cyberinfrastructure for natural hazards engineering. *Nat Hazard Rev*. 2017;18(3):06017001.