BUILDING EXTRACTION AND DAMAGE ESTIMATION BASED ON INSTANCE SEGMENT UTILIZING THE POST-EVENT AERIAL PHOTOS OF THE 2016 KUMAMOTO EARTHQUAKE

Yihao Zhan (1), Wen Liu (1), Yoshihisa Maruyama (1)

[1] Graduate School of Engineering, Chiba University,
1-33 Yayoi-cho, Inage-ku, Chiba, 263-8522, Japan
Email: zireael19andre@chiba-u.jp; wen.liu@chiba-u.jp; ymaruyam@faculty.chiba-u.jp

KEY WORDS: Damage evaluation, Object detection, Remote sensing, Mask R-CNN

ABSTRACTS: Remote sensing is an effective method to evaluate the damage situation after a large-scale nature disaster. Recently, deep learning algorithms have been used for the damage assessment from remote sensing images. A series of earthquakes hit the Kyushu region, Japan in April 2016, and caused severe damage in Kumamoto and Oita Prefectures. Numerous buildings were collapsed by the continuous strong shaking. In this study, the authors modified the Mask R-CNN model to extract residential buildings and estimate their damage levels. The Mask R-CNN model employs a two-stage instance segmentation algorithm which maintains a Convolutional Neural Network backbone and a Region Proposal Network with a ROI Align head. The aerial images captured on April 29, 2016 (two weeks after the main shock) in Mashiki Town, Kumamoto Prefecture, were used as the training and test sets. Comparing with the damage report of the field survey, the accuracy for the building extraction was 92%. As for the damage estimation, the precision and recall of the collapsed buildings achieved approximately 72% and 95%.

1. INTRODUCTION

In last decade, natural disasters occurred frequently over the world. It is important to grasp the damage situation immediately after a disaster. Although field surveys could provide detailed information, they require huge manpower and take much time. Under such circumstances, remote sensing technology is an effective tool to collect damage information. In recent years, deep learning algorithms as one category of the machine learning methods, have attracted widespread attention in the field of image recognition. This method does not require the operator to design the functions for the target detection. Instead, the computer would learn the image features automatically and output the results. Therefore, the time and workload required for the discrimination could be reduced. Meanwhile, highly convolutional neural network algorithms (CNN) have been able to identify object categories with higher accuracy than humans (Ishii et al., 2018). For these reasons, artificial intelligence neural networks have been widely used for

remote sensing images.

Khryashchev et al. (2018) used the high-definition satellite imagery dataset UC Merced to classify and identify common aerial objects, such as roads and agricultural fields. Yang et al. (2018) developed a model to improve the detection accuracy of the vehicles using aerial images. Furthermore, the deep learning algorithms were also applied for the damage assessment. Duarte et al. (2018) proposed three multi-resolution CNN feature fusion approaches to perform the image classification of building damage induced by earthquakes. Naito et al. (2020) compared the two training models using the traditional machine learning and the deep learning to identify the damage levels of buildings using the aerial images after the 2016 Kumamoto earthquake. Miura et al. (2020) developed a CNN model to classify the collapsed, no-collapsed and blue tarp-covered buildings from the post-event aerial images of the Kumamoto earthquake.

This research employs the neural network algorithm Mask R-CNN, which is mainly used for recognition of the large-scale object. This study performs object recognition and extraction from wide-area aerial images, and damage classification of residential buildings. The proposed model is applied to the aerial photos taken after the 2016 Kumamoto, Japan, earthquake. Building masks with collapse and non-collapse labels are generated automatically. The accuracy of this method is examined using the actual damage dataset after the earthquake, and its applicability is discussed in this paper.

2. DEEP LEARNING ALGORITHM AND PARAMETERS

The application of deep learning could be divided in three different fields: classification, object detection, and semantic segmentation. The instance segmentation is a recent approach including both the object detection and the semantic segmentation. In the instance segmentation, the object detection task detects the objects' classes with a bounding box predicted from an image. Then the semantic segmentation task classifies each pixel into the pre-defined categories. Thus, it enables us to detect objects and precisely segment the mask for each object instance in one structure. In this article, the authors modified the instance segmentation algorithms Mask R-CNN introduced by He et al. (2018), as an extension to Faster R-CNN (Ren et al., 2016), to recognize buildings from a mosaiced aerial image and classify their damage level.

Mask R-CNN adopts a two-stage procedure, with a Region Proposal Network (RPN) as the identical first stage. In the second stage, in parallel with predicting the class and box offset, Mask R-CNN predicts the class and box offset, and outputs a binary mask for each Region of Interest (RoI). **Figure 1** shows the structure of Mask R-CNN.

The default dataset for the valuation of Mask R-CNN is Microsoft COCO dataset (COCO Consortium, 2015). The images in this database are ground photos, and the target object often covers 40-70% of the full frame. On the contrary, aerial images and satellite images taken at a high altitude cover many small-scale objects. For this issue,

the authors modified the size of the anchor boxes for every feature map extracted from RPN. The large feature map size corresponds to the area of the small anchor boxes, which makes it easier to get more details order to detect small target objects. The authors tried two methods to modify the size of anchor boxes: halving and scaling. After testing several times on the anchor size parameters, the authors rescaled the anchor size from the original [32, 64, 128, 256, 512] to [8, 16, 64, 128, 256] as the final setting for training. This step improved the accuracy of the original model.

The optimization method of train strategy uses the Stochastic Gradient Descent (SGD) with Momentum. The momentum value was set to 0.9. The learning iteration is set to 90,000 times. Also, the initial setting of learning rate is 0.0025 with twice 0.0001 weight decay at 60000 and 80000 iteration. **Table 1** shows execution environment for Mask R-CNN and **Table 2** shows the various training parameters.

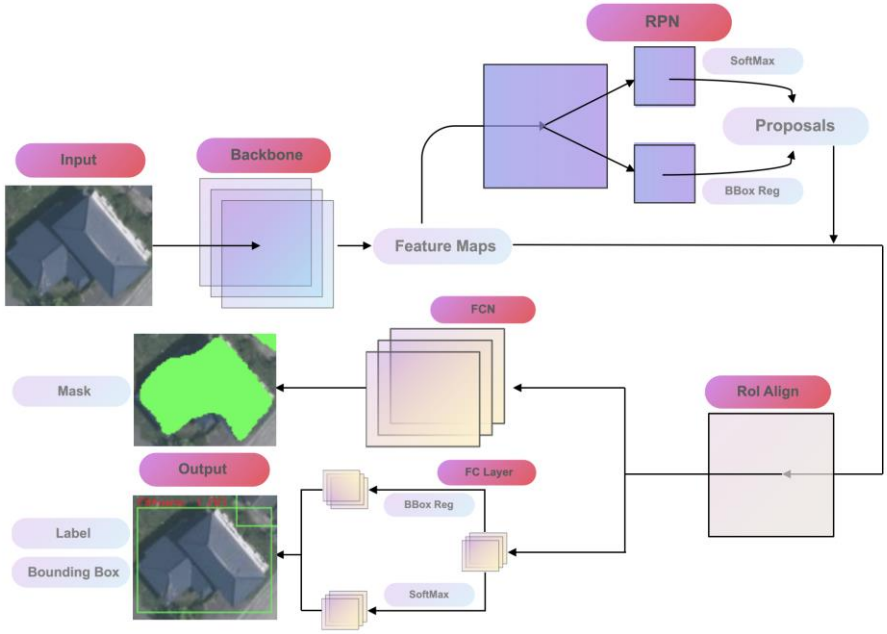


Figure 1 Structure of Mask R-CNN

Table 1 Execution environment

CPU	Intel Core i7-8700K CPU@ 3.70GHz × 12
GPU	GeForce GTX 1080Ti/PCIe/SSE2
Memory	32 GB
OS	Ubuntu 18.04
Programing Language	Python 3.7
Deep learning framework	Pytorch 1.4.0 dev20191016
Platform	CUDA 10.1

Table 2 Training Parameters

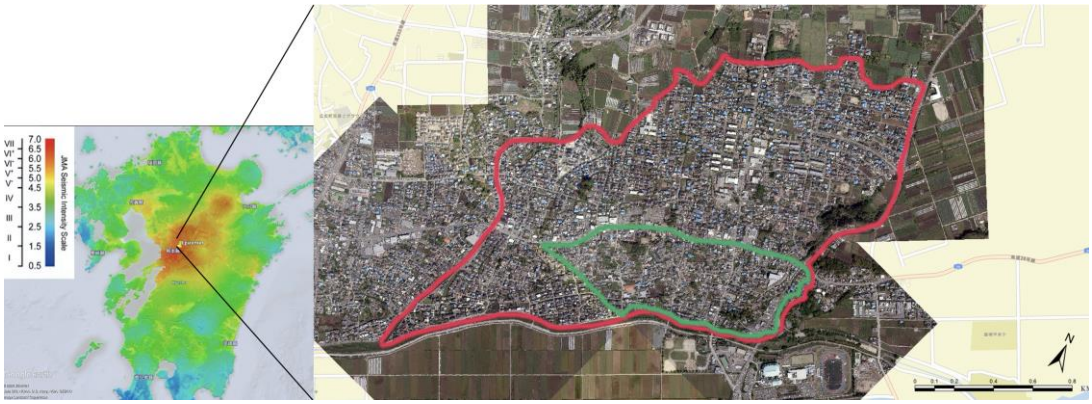
Anchor aspect ratio	[0.5, 1, 2]
Horizontal flip prob train	0.5
Vertical flip prob train	0.5
Warmup iteration/method	500/Constant
Batch size train	2
Batch size test	1

3. DATASET

A series of earthquakes affected Kumamoto Prefecture, Japan, in April 2016. Due to the continuously strong shaking, more than 8,000 buildings collapsed and about 30,000 buildings were severely damaged. Geospatial Information Authority of Japan (GSI) took aerial photos in the severe damaged regions several times after the earthquakes. In this study, the aerial photos of Mashiki Town taken on April 29, 2016 by GSI were used to train and test our proposed model. Mashiki Town was one of the most affected areas, where severe ground motions with the Japan Meteorological Agency (JMA) seismic intensity scale of 7 were observed twice. The distribution of the JMA seismic intensity of the main shock in Kyushu region is shown in **Figure 2(a)** (GSJ, 2016).

The aerial photos used in this study were taken by UltraCamX, with 200 pixels/mm spatial resolution (Microsoft Photogrammetry, 2016). Six photos covered the main area of Mashiki Town. The basic parameters of the aerial photos are shown in **Table 3**. The mosaiced image is shown in **Figure 2(b)**. Then we cut the mosaiced image to 500×500 pixels images. Finally, the authors obtained totally 223 images for training and valuation. This format was also following that of the Microsoft Common Object in Context (COCO) dataset.

These images were labeled manually by the tool LabelMe (Wada, 2015) into two damage categories. The building damage categories were cited from the work of Yamada et al. (2017). They performed visual interpretation of building damages from the post-event aerial photos and field surveys, according to the damage grades developed by Okada and Takai (1999). The damage grades were set from D0 to D5. Here we merged the damage grades D4 and D5 into “Collapsed” label, including collapse and torsion of first floor, partial collapse, total crushing, and beam (roof) fracture. The damage grades D0 to D3 were merged to the label “Others”. The detail description of the damage classification is shown in **Table 4**. Several samples of the labeled buildings are shown in **Figure 3**.



(a) Seismic intensity (b) Mosaiced aerial image

Figure 2 Study area of Mashiki Town, Kumamoto Prefecture, Japan: (a) JMA seismic intensity for the mainshock on April 16, 2016 (GSJ, 2016); (b) the mosaiced aerial image taken on April 29, 2016 by GSI (2016).

Table 3 Acquisition conditions of six aerial photos taken by UltraCam-X

Number of pixels	Number of scenes	Rotation (°)	Division	Number of channels
7215×4710	4	270	Training set	R, G, B
4710×7215	2	140	Valuation set	

Table 4 Building damage classifications

Damage classifications	Features in photographs	Damage grades (Okada and Takai, 2000)	Color
Collapsed	• Distortion of the entire building, destruction or collapse	D4, D5	Red
Others	• No damage • Some roof tiles collapsion • Lots of roof tiles collapsed, or some walls have fallen	D0, D1, D2, D3	Green

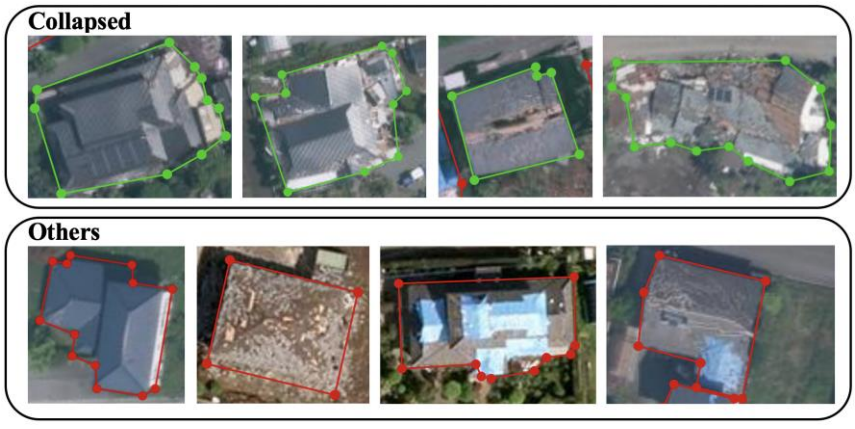


Figure 3 Examples of the buildings labeled in two different damage levels

4. TRAINING AND VALUATION

The 223 images cut from the mosaiced aerial image enclosed by the red line in **Figure 2(b)**, were applied to the proposed model. 44 images cover the area enclosed by the green line were used for both testing and valuation. The results of multiple backbone models were compared: ResNet-101, ResNet-50 (He et al., 2015), Deep Cross Network (DCN, Wang et al., 2019), and Facebook-Berkeley-Network (FBNet, Wu et al., 2019).

In object detection problems, Intersection over Union (IoU) evaluates the overlap between ground-truth (GT) and predicted-result (PR). It is calculated by **Eq. (1)**.

$$IoU = \frac{area(GT \cap PR)}{area(GT \cup PR)} \tag{1}$$

Mask R-CNN uses the default Microsoft COCO dataset Evaluation for the detection task (He et al. 2018). The evaluation is based on the precision and the recall. Precision is the ability of a classifier to recognize only relevant objects. It reflects the proportion of true positive detections. Meanwhile, recall measures model's ability to detect all the GTs (that is the proportion of true positives detected in all GTs). The authors also use the confusion matrix for further evaluation of the classification results shown in **Table 5**. The evaluation indices: Overall Accuracy, Precision, Recall, F-measure, are defined by **Eqs. (2-5).**

$$Overall\ \ Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} = \frac{TP}{all\ \ detections} \tag{3}$$

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{all\ \ GT} \tag{4}$$

$$F - measure = \frac{2Recall \cdot Precision}{Recall+Precision} \tag{5}$$

Further on, the graph of the relationship between precision-recall rate function is called precision-recall rate curve (PR-C). It shows the trade-off between the two metrics to change the confidence value of model detection. Average Precision (AP) is the area under the PR curve. Mathematically, AP is defined by **Eq. (6)**.

$$AP = \int_0^1 p(r)dr \tag{6}$$

Table 6 shows six main metrics of the COCO evaluation, including mAP (mean AP) and multiple AP with different α values. Mask R-CNN reports AP (mAP), AP50, AP75 and AP at different scales (small, medium, large). **Table 7** shows the comparison of the accuracies obtained by the multiple backbone models. According to the comparison, the authors found that ResNet-101 obtained the best mAP among all the backbone models, reaching approximately 42.1%. The mAP of Mask R-CNN using the default COCO dataset is 38.2%.

We used the weight obtained in the training process for the valuation images to detect buildings and predict damage levels. In the valuation region enclosed by the green line in **Figure 2(b)**, there are a total of 628 buildings. After the training step, our model generated 657 bounding boxes as buildings. The 603 buildings were successfully extracted and masked, whereas 25 buildings were missed. Thirteen out of the 25 missing buildings were collapsed. The overall accuracy of the building detection reached approximately 91.8%, whereas 95.5% of the collapsed buildings were identified. **Figure 4** shows two examples of the results of the building detection and the damage prediction.

The authors considered the two major reasons for the mis-extraction of buildings: boundary line missing and other distractions. The evaluation region enclosed by the green line area in **Figure 2(b)**, was the most severe damaged area with many fully collapsed buildings. Several collapsed buildings split to half or even smaller parts. Thus, our model failed to identify them as buildings, and caused certain mis-extraction. Some small structures, such as garages, warehouses, were also the distraction factors of the processing. They caused the misrecognition of buildings.

Furthermore, the result of the damage classification for the extracted 603 buildings is shown in **Table 8**. The overall accuracy of the classification in our model was approximately 85.6%. The recall of the collapsed class reached approximately 95.3%, and the precision was approximately 72.4%. The F-measure was approximately 82.3%.

One main reason for the misclassification was the observation angle of the aerial photos. Several damage patterns, such as the pancake collapse or damages on the walls, were difficult to be observed from the top. Besides, the customed roof design, the scattered rubble or the abnormal shaped building would also lead to a misclassification.

Table 5 Confusion Matrix

		Predicted labels (Mask R-CNN)	
		Positive	Negative
Actual labels (Visual interpretation)	Positive	TP	FN
	Negative	FP	TN

*TP: true positive; FN: false negative; FP: false positive; TN: true negative.

Table 6 Average Precisions definded in the COCO Evaluation

Average Precision (AP)	
AP (mAP for COCO)	%AP at IoU .50: .05: .95 (Primary challenge metric)
$AP^{IoU=.50}$	%AP at IoU= .50 (PASCAL VOC metric)
$AP^{IoU=.75}$	%AP at IoU= .75 (Strict metric)
AP Across Scales:	
AP^{small}	%AP for small objects: area < 32^2
AP^{medium}	%AP for medium objects: 32^2 < area < 96^2
AP^{large}	%AP for large objects: 96^2 < area

Table 7 Comparison of the accuracies of the multiple backbone models

Model	AP (mAP)	AP50	AP75	AP_s	AP_m	AP_l
ResNet-101 (our model)	**42.1**	**69.1**	**44.3**	**16.2**	**43.9**	**26.5**
ResNet-101	41.7	67.9	43.5	16.1	42.1	26.6
ResNet-50	39.2	67.3	41.8	13.1	41.6	24.9
DCN	37.2	63.5	40.0	16.5	38.9	24.3
Fbnet	30.5	47.3	30.7	31.4	29.9	27.4

Figure 4 Examples of the results of building detection and damage prediction

Table 8 Results of the damage classification

		Predicted labels (Mask R-CNN)				
		Collapsed	Others	Precision	Recall	F-measure
Actual labels	Collapsed	202	77	72.4%	95.3%	82.3%
	Others	10	314			

5. CONCLUSION AND FUTURE STUDY

In this paper, the authors modified the deep learning network framework Mask R-CNN to extract buildings and classify their damage level. Six aerial photos which captured Mashiki Town, Kumamoto Prefecture, Japan, were used as the training and valuation data. The damaged buildings due to the 2016 Kumamoto earthquake were divided into two categories: collapsed (the damage grade D4 and D5) and others (the damage grade D0-D3). After increasing the complexity of the data set and modifying the anchor size, the authors performed automatic mask generation for the building extraction and the damage classification.

By training 179 images after 90,000 iterations, the proposed model predicted the results of 44 images. Our model achieved the 42.1% mAP, and it was 3% higher than the default COCO dataset. The overall accuracy of the building detection reached 91.8%. 95.6% of the collapsed building could be identified successfully. Although our dataset has only 223 images, which were less than the other datasets used for deep learning, the overall accuracy of the classification achieves 85.6%. In the future study, the authors would increase the number of images to improve the model.

ACKNOWLEDGEMENT

The aerial photos used in this study were taken and owned by Geospatial Information Authority of Japan.

REFERENCES

COCO Consortium, 2015. Microsoft Common Objects in Context (COCO) dataset. from https://cocodataset.org/#home .

Duarte D., Nex F., Kerle N., VosSelman G., 2018. Multi-resolution Feature Fusion for Image Classification of Building Damages with Convolutional Neural Networks. Remote Sensing, 10(10), 1636; doi:10.3390/rs10101636.

Geological Survey of Japan (GSJ), National Institute of Advanced Industrial Science and Technology (AIST), 2016. Quick estimation system for earthQuake map triggered by observed records (QuiQuake). from https://gbank.gsj.jp/QuiQuake/QuakeMap/.

Geospatial Information Authority of Japan. GSI Map. 2016. from https://www.gsi.go.jp.

He K., Gkioxari G., Dollár P., Girshick R., 2018. Mask R-CNN. Computer Vision and Pattern Recognition (cs.CV), arXiv:1703.06870v3.

He K., Zhang X., Ren S., Sun J., 2015. Deep Residual Learning for Image Recognition. Computer Vision and Pattern Recognition (cs.CV), arXiv:1512.03385v1.

Ishii Y., Matsuoka M., Maki N., Horie K., Tanaka S., 2018. Recognition of Damaged Building Using Deep Learning Based on Aerial and Local Photos Taken After The 1995 Kobe Earthquake. Journal of Structural and Construction Engineering

(Transactions of AIJ), 83(751), pp.1391-1400.

Khryashchev V. V., Pavlov A. V., Priorov A., Ostrovskaya A. A., 2018. Deep Learning for Region Detection in High-Resolution Aerial Images. IEEE East-West Design & Test Symposium (EWDTS), Kazan, pp. 1-5.

Microsoft Photogrammetry, UltraCam-X Technical Specifications. 2016. from https://www.sfsaviation.ch/files/177/SFS%20UCX.pdf.

Miura H.; Aridome T.; Matsuoka M., 2020 Deep Learning-Based Identification of Collapsed, Non-Collapsed and Blue Tarp-Covered Buildings from Post-Disaster Aerial Images. Remote Sens, 12(12), pp.1924.

Naito S., Tomozawa H., Mori Y., Nagata T., Monma N., Nakamura H., Fujiwara H., Shoji G., 2020. Building-damage Detection Method Based on Machine Learning Utilizing Aerial Photographs of The Kumamoto Earthquake. Earthquake Spectra, 36(3), pp.1166–1187.

Okada S., Takai N., 1999. Classifications of Structural Types and Damage Patterns of Buildings for Earthquake Field Investigation. Journal of Structural and Construction Engineering (Transactions of AIJ), 64 (524), pp. 65-72.

Ren S., He K., Girshick R., Sun J., 2016. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Computer Vision and Pattern Recognition (cs.CV), arXiv:1506.01497v3.

Wada K., 2015. labelme: Image Polygonal Annotation with Python. from https://github.com/wkentaro/labelme .

Wang R., Fu B., Fu G., Wang M., 2019. Deep & Cross Network for Ad Click Predictions. Machine Learning (cs. LG); Machine Learning (stat.ML), arXiv:1708.05123v1.

Wu B., Dai X., Zhang P., Wang Y., Sun F., Wu Y., Tian Y., Vajda P., Jia Y., 2019. Keutzer K., FBNet: Hardware-Aware Efficient ConvNet Design via Differentiable Neural Architecture Search. Computer Vision and Pattern Recognition (cs.CV), arXiv:1812.03443v3.

Yamada M., Ohmura J., Goto H., 2017. Wooden Building Damage Analysis in Mashiki Town for the 2016 Kumamoto Earthquakes on April 14 and 16. Earthquake Spectra, 33(4), pp. 1555–1572.

Yang Y. M., Liao W., Li X., Rosenhahn B., 2018. Deep Learning for Vehicle Detection in Aerial Images. 25th IEEE International Conference on Image Processing (ICIP), Athens, pp. 3079-3083.