Smart Structures and Systems, *Vol. 31*, *No. 4* (*2023*) *383-392*
https://doi.org/10.12989/sss.2023.31.4.383

Automatic assessment of post-earthquake buildings based on multi-task deep learning with auxiliary tasks

Zhihang Li [1], Huamei Zhu*[1], Mengqi Huang [1], Pengxuan Ji [1], Hongyu Huang [2] and Qianbing Zhang [1]

[1] *Department of Civil Engineering*, *Monash University*, *Wellington Road Clayton*, *Victoria 3800*, *Australia*
[2] *Institute of Geotechnical Engineering*, *Zhejiang University*, *Hangzhou 310058*, *China*

(*Received September 7*, *2022*, *Revised December 4*, *2022*, *Accepted February 2*, *2023*)

Abstract. Post-earthquake building condition assessment is crucial for subsequent rescue and remediation and can be automated by emerging computer vision and deep learning technologies. This study is based on an endeavour for the 2nd International Competition of Structural Health Monitoring (IC-SHM 2021). The task package includes five image segmentation objectives – defects (crack/spall/rebar exposure), structural component, and damage state. The structural component and damage state tasks are identified as the priority that can form actionable decisions. A multi-task Convolutional Neural Network (CNN) is proposed to conduct the two major tasks simultaneously. The rest 3 sub-tasks (spall/crack/rebar exposure) were incorporated as auxiliary tasks. By synchronously learning defect information (spall/crack/rebar exposure), the multi-task CNN model outperforms the counterpart single-task models in recognizing structural components and estimating damage states. Particularly, the pixel-level damage state estimation witnesses a mIoU (mean intersection over union) improvement from 0.5855 to 0.6374. For the defect detection tasks, rebar exposure is omitted due to the extremely biased sample distribution. The segmentations of crack and spall are automated by single-task U-Net but with extra efforts to resample the provided data. The segmentation of small objects (spall and crack) benefits from the resampling method, with a substantial IoU increment of nearly 10%.

Keywords: building assessment; CNN; multi-task deep learning; semantic segmentation; small object detection

1. Introduction

Earthquakes accounted for around 1.87 million deaths in the past century with building collapses identified as the major fatal cause (Blaikie *et al*. 2014, Doocy *et al*. 2013). Secondary hazards can be detrimental as well. For example, a devastating tragedy of 44 fatalities including rescuers and journalists was caused by the aftershock in Van, Turkey in2011 (Mehdi and Nazmazar 2013). In a quick response to earthquakes, condition assessment of damaged buildings is crucial to provide timely rescue of lives, informed decision-making, and mitigation measures for secondary hazards. Manual inspection can be laborious, causing a delay in subsequent actions, and sometimes hazardous, resulting in worker casualties due to secondary effects like aftershocks. These concerns are calling for automatic alternatives for post-earthquake building assessments. Computer vision technologies bring new perspectives and incentives to automate such a task. Particularly, the emerging deep learning algorithms that can be deployed in Unmanned Aerial Vehicles (UAV) are making substantial contributions (Azimi *et al*. 2020, Bao *et al*. 2019, Spencer *et al*. 2019).

1.1 Literature review

Computer vision-based building assessment can be characterised by various detection scales ranging from image-wise, object-wise to pixel-wise. Image-wise detection aims to classify pictures of the monitored structures. For instance, a Convolutional Neural Network (CNN) was trained to sort input image patches as crack-containing or non-crack- containing (Cha *et al*. 2017). For object-wise detection, target categorization is intended. Guo *et al*. (2020) automated a multi-categorical classification task by a meta-learning-based CNN to recognize the image of a building façade as blistering, peeling, cracking, delamination, spalling, or biological growth. In addition, refined damage regions can be localized with bounding boxes by object detection methods, such as in Cha *et al*. (2018), Pan and Yang (2020). However, precise damage information is required to perform comprehensive condition assessments and make informed decisions. Semantic segmentation at the pixel level is thus adopted in some attempts (Choi and Cha 2020, Narazaki *et al*. 2020). Many pieces of research focused on a single task, especially damage recognition. However, in practice, the UAV-based building condition assessing process encompasses more than one task to sense the varying environments, plan out optimal routes and recognize targets from complicated backgrounds. Jointly training one model with related tasks has the proven ability to enhance the accuracy and make robust evaluations by leveraging multi-source information (Hoskere *et al*. 2018, 2020, Ruder 2017, Vandenhende *et al*.

*Corresponding author, Ph.D. Candidate,
E-mail: Huamei.zhu@monash.edu

ISSN: 1738-1584 (Print), 1738-1991 (Online)

2020). Not only because these tasks share a similar pattern but they also inter-correlate with each other, indicating the potential and necessity of multi-task learning methods to conduct multiple tasks in one shot.

1.2 Research background

This study is the attempt for Project 2 (computer vision-based post-earthquake inspections of buildings) of the 2nd International Competition of Structural Health Monitoring (IC-SHM) (Spencer and Li 2021). This project aims to automate UAV-based health estimation of post-earthquake buildings, which encompasses five pixel-wise sub-objectives. Included aims are to identify structural components, detect damaged regions (spall, crack, and exposed rebar), and assess damage states. Deep learning methods provide promising solutions to automate building damage estimation tasks. Quality data is the major booster to optimising any data-driven algorithms. Failing to collect adequate data is implicitly hindering interdisciplinary applications. State-of-the-art attempts are being made in potential fields, such as resorting to transfer learning (Gao and Mosalam 2018), or data augmentation (Lee *et al*. 2022, Li *et al*. 2022a). Innovatively, the IC-SHM 2021 committee generated synthetic data with desirable quality and quantity in a scientifically sound way, which demonstrates huge opportunities in interdisciplinary deep learning explorations (Narazaki *et al*. 2021).

The well-labelled QuakeCity dataset was provided to foster the explorations of deep learning methods in automating the five tasks (Hoskere *et al*. 2022). Two challenges were identified, arising from the varied environments and sizes of tracking targets. First, the multiple classes to be predicted are highly correlated with each other. For example, some components of key structural importance such as columns and beams are more likely to be damaged in seismic events. Besides, the damage state estimation task, which is the goal of condition assessment, is dependent on the presence of defects. For instance, rebar only exposes at the premise of spalling and usually indicates the severe damage state of the region (Fig. 1). Second, the tremendously varied target sizes among sub-tasks pose another challenge of prediction bias for data-driven algorithms. The structural component can be captured easily even from remote sight. Whereas small-size damages like cracks are only visible when the camera is closely present. Specifically, Fig. 1 summarises the sample distribution for each detection objective. The amounts of foreground pixels in component segmentation and damage state estimation are relatively balanced, while that in defect segmentation (spall/crack/rebar exposure) are severely biased. For example, the crack sample in total shares less than 1% of the dataset. In addition, when identifying the rebar from the background, the proportion of positive (i.e., rebar) and negative (background) samples can be as high as 18,000:1. Therefore, component and damage state segmentation tasks are treated as the normal object segmentation task with their relatively mass positive samples. The spall and crack pixels are characterised by their severely non-balanced proportions and thus sorted into the small segmentation task, which requires special attention not only on the model structure but more on the data preprocessing. For instance, training deep learning models with the positive sample rate of 1/18,000 in the original rebar dataset tends to disable most error backpropagation algorithms without any pre-filtering. Rebar exposure detection is abandoned for now out of the extremely biased sample distribution.

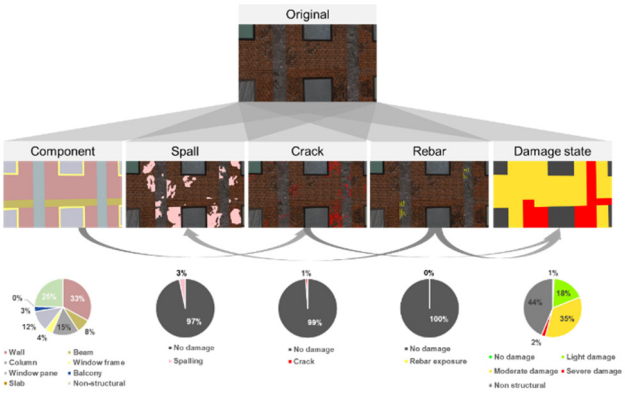


Fig. 1 Dependence among five sub-tasks: representative images and sample distribution of each sub-task (structural component, spall, crack, rebar exposure, and damage state estimation)

Aiming at the first challenge, a multi-task CNN model with auxiliary tasks was designed to identify structural components and estimate damage states. Particular attention was devoted to data augmentation to address the second challenge for spall and crack segmentation tasks. Afterwards, U-Net was adopted and optimised by spall and crack datasets pre-filtered from the raw images. Supported by the open-source synthetic dataset, explorations were made to find potential optimum solutions to automate the post-earthquake estimation task. In the following, Section 2 details the used methods; Section 3 shows an implementation of the experiments; Section 4 presents the outcomes and related discussions, and Section 5 concludes the study.

2. Methodologies

A multi-task CNN model with hard parameter sharing was designed for post-earthquake building estimation. The model takes structural component identification and damage state estimation as its major tasks and spall/crack/rebar exposure segmentation as the auxiliary tasks. Contributing modules of the model include Deep Residual Network (ResNet) (He *et al*. 2016), Spatial Pyramid Pooling (SPP), Atrous Spatial Pyramid Pooling (ASPP) (Chen *et al*. 2016) and Fully Connected (FC) layer.

The contributing modules are ensembled as in Fig. 2. Specifically, the network extracts feature from the input image using several preliminary blocks. After that, the feature map is further processed by a modified ResNet 50 backbone. The first modification of ResNet 50 here refers to the integration of ASPP to enhance its acquisition of size and view-varying objectives. Besides, skip connections were added within the backbone to minimise the feature

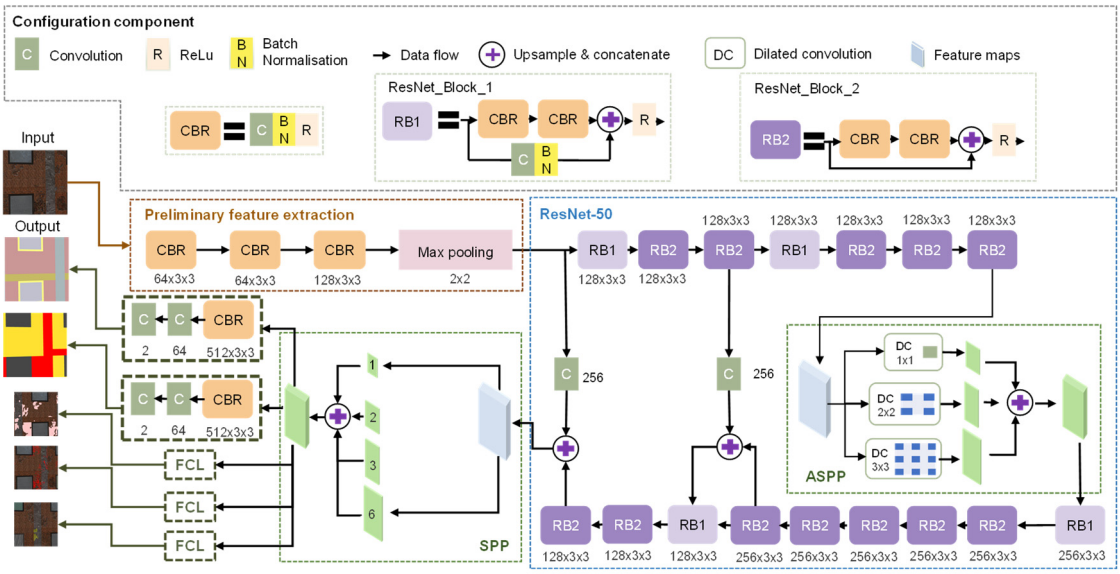


Fig. 2 Architecture of the proposed multi-task CNN model with auxiliary tasks: ensembled by (1) ResNet, (2) ASPP, (3) SPP, and (4) five independent task-specific FC layers

loss of fine targets such as cracks and rebars with overwhelmed proportions. Then, the feature map output from ResNet 50 is condensed and resized to fixed-length representations by SPP. Eventually, five task-specific FC-based decoders take in the representations and produce the final prediction mask for each task independently.

2.1 Residual network

Theoretically, deeper neural networks can extract more complex features while under the increasing risks of gradient vanishing/exploding and accuracy degradation. Deep residual network (He *et al*. 2016), referred to as ResNet, was proposed to solve such problems in deep networks. The basic unit of ResNet is the residual block (Fig. 2). Compared with the conventional plain network structure, residual networks add skip connections between every two layers enabling the consecutive layers to learn residuals directly from the previous layers. This structure of layers constitutes a deep residual network and solves the problem of decreasing accuracy during model training. There are two mappings in ResNet. One is identical mapping, which refers to the input data itself and is represented as a curve in the figure. The other is residual mapping, which refers to the rest of the network. The advantage of ResNet is the skip connection architecture so that the deep network can be trained adequately without inducing a vanishing gradient. Hence, a CNN can be extended to incorporate more functional layers without risking the error rate.

2.2 Spatial pyramid pooling (SPP) network and atrous spatial pyramid pooling (ASPP) network

Spatial Pyramid Pooling Network (SPP-Net) was proposed to enable arbitrary size inputs and generate fixed-length features (He *et al*. 2014). SPP block is positioned after the feature map extraction and before the fully

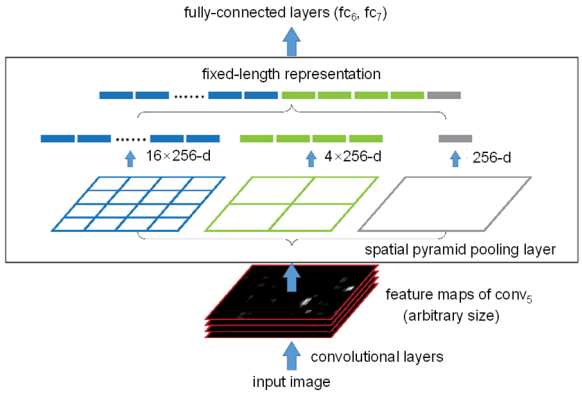


Fig. 3 The SPP layer that divides the feature maps into a fixed number of bins with sizes proportional to the image size to achieve a fixed-length representation (from He *et al*. 20144)

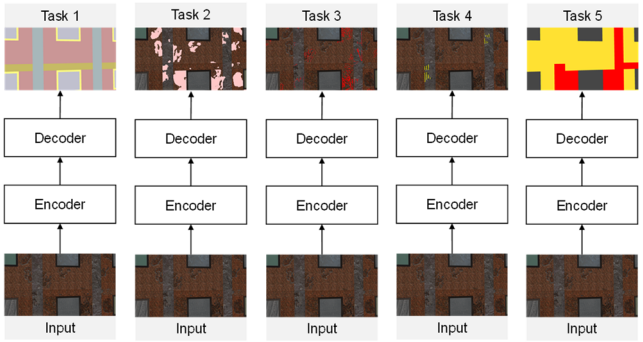
connected layers (Fig. 3).

Several spatial bins are implemented for pooling to maintain spatial information from the feature maps. The outputs are concatenated to form a fixed-length representation that will be fed into the fully-connected layers.

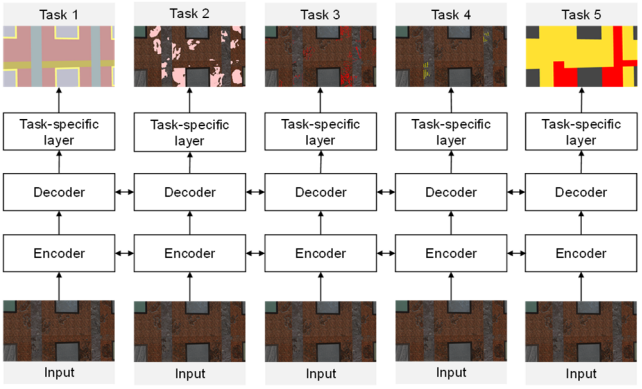
The SPP structure can effectively prevent incomplete clipping and shape distortion caused by the R-CNN algorithm; more importantly, it solves the problem of repetitive feature extraction by the convolutional neural network. To avoid background interferences and allow strong target features to be extracted, a deep network is adopted in this study. However, due to the intensive feature accumulation operation of the traditional convolution kernel method, there may be overlapping between the receptive fields, which increases the complexity of semantic information and results in a waste of computation resources (Li *et al*. 2018). The dilated convolution technique is thus inserted in the proposed network, which only places

weights in certain positions, and fills other spaces in the kernel with zeros (Yu and Koltun 2015). By geometrically increasing the dilation rate in continuous convolution layers, the receptive field can be extended while ensuring coverage.

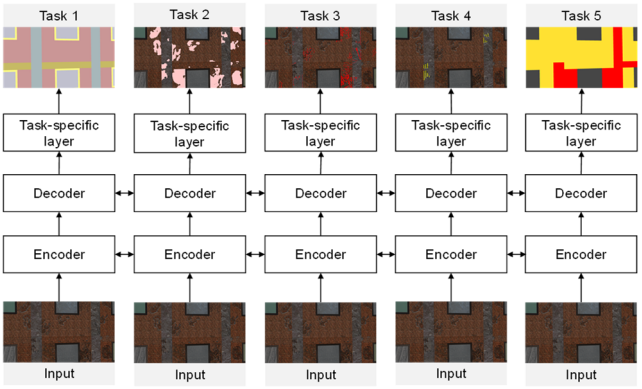
The ASPP (Yang *et al*. 2018, Zhao *et al*. 2017) is a further improvement of dilated convolution. It combines dilated convolution with Spatial Pyramid Pooling (SPP) module. The Pyramid Pooling Module is inspired by the superiority of the R-CNN SPP method, which demonstrates accuracy and efficiency in classifying a region of arbitrary scale by resampling the convolution features extracted from a single scale.



(a)



(b)



(c)

Fig. 4 Schematics of deep learning networks: (a) single-task network, (b) multi-task network with soft-sharing parameters, and (c) multi-task network with hard-parameter sharing

2.3 Multi-task learning architecture

Deep learning applications are predominantly associated with a single task which forms the mainstream single-task architecture (Fig. 4(a)). Whereas, in the case of more than one task, particularly when the tasks are interrelated with each other, the single-task solution may cause heavy computation burden and the waste of shared information. For instance, to automate the five segmentation tasks involved in this study, five dense networks must be optimised independently if adopting the single task path. In such cases, multi-task deep learning networks are increasingly mentioned to cope with multiple tasks.

Multi-task deep learning networks are ensembled usually in two parameter-sharing ways: soft-sharing (Fig. 4(b)) or hard-sharing (Fig. 4(c)) (Ruder 2017). Soft -sharing multi-task architecture forces the independent models to assimilate their parameters by regularizing the parameter distance. Like the single-task in Fig. 4(a), the soft-sharing multi-task network has five separate models but with weak parameter sharing. In hard-sharing mode, five separate models are compacted into one, with only five independent task-specific layers. Hard parameter sharing is the mostly used multi-task layer method in neural networks. The hard sharing of parameters is defined as the hidden layer being shared among all tasks, and different output layers are formulated for different tasks. Most parameters are completely shared among the multiple tasks in the hard-sharing network (Fig. 4(c)).

Hard parameter sharing can reduce the risk of overfitting, since learning diverse tasks synchronically motivates the model to find universal representations over all tasks. Multi-task learning architecture output multiple independent predictions, which have been proven to be physiologically aligned with the human being. Deep learning could be identified as a method to extract useful information for a specific task from a large amount of data. For the output prediction of different tasks in the same dataset, the features extracted from the data are completely distinct (Ando and Zhang 2005). In addition, during the process of extracting features from the data set for a specific task, the unnecessary labels that exist in the dataset tend to generate different noise patterns and interfere with the output prediction (Zhang and Huang 2008), thus reducing the overall prediction accuracy. Moreover, the model is likely to learn the data and noise patterns (Zhang and Huang 2008), resulting in overfitting. In the same model, the data features of multiple different categories will be embedded into the same semantic space (Baxter 1997). But the specific features of each task retain their parameters to extract the same features during the output stage, which renders the model with the ability to deal with multiple tasks.

The benefits of multitask learning method are manifested during the training process, its effect is equivalent to averaging the noise corresponding to different tasks, weakening the influence of noise, and allowing the model to effectively extract useful features from different tasks for improved segmentation (Søgaard and Bingel 2017). With its unique internal parameter-sharing mechanism, a multi-task model, theoretically, has better

generalization ability and a lower risk of overfitting compared with single-task models.

2.4 Data filtering

The provided dataset consists of 3,805 annotated pictures with a size of 1,080×1,920 captured from damaged buildings by UAVs at different heights and views. 80% of original acquisitions were randomly extracted to form the training set and the remaining was used for model validations. Two dataset preparation modes were adopted in this study (Fig. 5) to tackle the sample imbalance problem for the sub-tasks. In plain mode, images of the original size were directly rescaled to 512×512 without any further manipulation. In filtering mode, where most information can be retained, each image was cropped to eight patches with the size of 512×512. For the small object segmentation tasks (spall and crack), subject to the non-equilibrium distribution, extra efforts were applied to filter out the overwhelming background. Cropped patches with a positive sample amount (spall and crack) less than the specified threshold value were discarded.

Sample distributions were considerably balanced after oversampling in the filtering mode, as shown in Table 1. The plain dataset was fed into the model with component and damage status estimation as the main tasks while the three balanced datasets of spall and crack were used to optimise the corresponding U-Net.

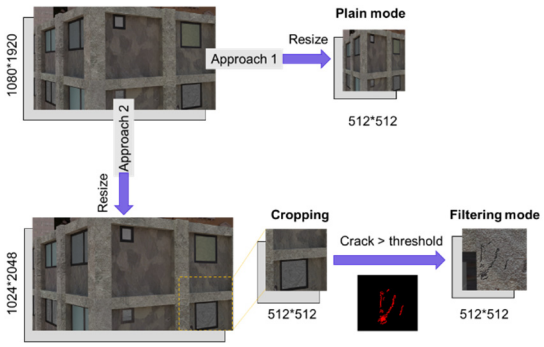


Fig. 5 Two dataset preparation approaches: 1 - plain mode and 2 - filtering mode

Table 1 Data balancing results of filtering mode

Category	Raw dataset		Filtered dataset	
	Number of patches	Proportion of positive samples	Number of patches	Proportion of positive samples
Spall	24,352	3%	15,000	10%
Crack	24,352	1%	3,700	3%

Table 2 Model definition and corresponding tasks

Model	Component	Damage status	Crack	Spall
Model A	√	√		
Model B			√	
Model C				√

3. Implementation and experiments

A total of 3 models were trained to complete the 4 tasks including structural component identification, damage state estimation, and spall and crack segmentation, as summarised in Table 2. The rebar exposure task was incorporated as an auxiliary task in the multi-task model but was abandoned as its own in this study because of the extremely imbalanced sample distribution. Efficient and timely remediation after an earthquake requires informative recognition of the component and conditions of damaged buildings to prioritise limited time and labour resources. Hence the structural component and damage state estimation tasks are identified as the priority among all sub-tasks.

Most research efforts were directed to Model A which used the proposed network (Fig. 2) with component segmentation and damage status estimation as its major tasks and damage segmentations (crack/spall/rebar) as its auxiliary tasks. It was optimised solely on the plain dataset (Fig. 5(a)) considering that its major task datasets are rather an equilibrium. For the damage segmentation tasks (crack/spall) where their highly-unbalanced datasets need to be customised individually, independent networks were trained to enable each task. Model B refers to U-Net which was trained on cropped patches selected containing crack samples. Similarly, Model C learned from the filtered spall dataset. It should be noted that Model A will be used to evaluate the multi-task learning network and Model B/C was trained mainly to explore the data filtering strategy.

3.1 Training platform and protocol

Specifications of the deep learning platform used for structural components and evaluate the damage state, leveraging auxiliary tasks of defect segmentation framework proposed by this paper are Windows 10, Python 3.7.4, and Keras 2.13; platform hardware configuration is CPU Intel i7 9800X with 32GB of memory; configuration for graphics card includes one NVIDIA RTX2080Ti, 11GB of video memory, CUDA10.0, and NVIDIA cuDNN7.4.2 are used for GPU acceleration. Training details can be found in Table 3, where all hyper-parameters were determined based on pretraining.

Considering the graphic card of the deep learning platform is not designed for deep learning tasks of large data size, the input structure of the network, batch size, and GPU memory allowance in training was adjusted to prevent

Table 3 Training hyperparameters of three models

Hyperparameters	Model A	Model B	Model C
Initial learning rate	1e-4	1e-4	1e-5
Epoch		80	
Steps per epoch		2000	
Batch size	4	2	2
Optimiser		Adam	
Threshold		0.5	
Data augmentation		Vertical and horizontal flip	

the out of memory. Specifically, image patches with the size of 512×512 were compressed (Model A) and cropped (Model B, C) from raw data, and GPU memory allocation was set as growing adaptively with demand. Each deep learning model discussed in this study was trained for 80 epochs with 2,000 steps in each epoch. Learning rate decay was deployed in the training with a decay rate of 10 given that no improvement is recorded for five epochs.

Three annotated image datasets with different data preprocessed strategies are used for various tasks to train the models by a backward propagation algorithm. For model A, 3244 images of the original size were directly rescaled to 512×512 without any further manipulation. Among them, 3000 images act as training sets while the rest of the images are used as validation sets to validate the training effect at the end of each epoch. For models B and C, where most information can be retained, each image was cropped to eight patches with the size of 512×512. For the small object segmentation tasks (spall and crack), subject to the non-equilibrium distribution, extra efforts were applied to filter out the overwhelming background. Cropped patches with a positive sample amount (spall and crack) less than the specified threshold value were discarded. Output patches generated by models B and C were then merged into full size, to evaluate the model performance on full-scale images. The trained models were tested with 1004 images provided by the committed without annotation, while the output results of models are compiled into CSV table format by a specific algorithm and uploaded to the Kaggle platform to obtain the final evaluation scores.

3.2 Loss functions

The multi-task network was optimised by a joint loss function, summed from the five sub-tasks, as formulated in Eq. (1).

$$L = \sum_{i=1}^{i=5} \omega_i L_i \tag{1}$$

where i indicates the five sub-tasks; the loss weight ω_i is used to adjust the optimization priority of the loss L_i for task i. In this study, the loss weight was set as 1.0 for main tasks and 0.1 for auxiliary tasks. Categorical Cross Entropy loss was applied in component segmentation and damage status estimation tasks. As for the small object segmentation (spall and crack), Dice loss was adopted, which incorporates weights specialised to mitigate the biased sample distribution (Li *et al*. 2022b), as formulated in Eq. (2) (Milletari *et al*. 2016).

$$(A, B) = \frac{A \cap B + smooth}{A \cap B + \alpha|A - B| + \beta|B - A| + smooth} \tag{2}$$

where A is the predicted results and B is the ground truth. $|A\text{-}B|$ represents the FP (false positive) set and $|B\text{-}A|$ represents FN (false negative) set. The balance between these two clusters can be controlled by adjusting α and β, thus affecting the recall rate and other performance indicators. The original Dice loss was used in this study with $\alpha = \beta = 0.5$ and $smooth = 1.0$.

3.3 Evaluation metrics

Intersection over union (IoU) was used to evaluate segmentation performance, as defined in Eq. (3).

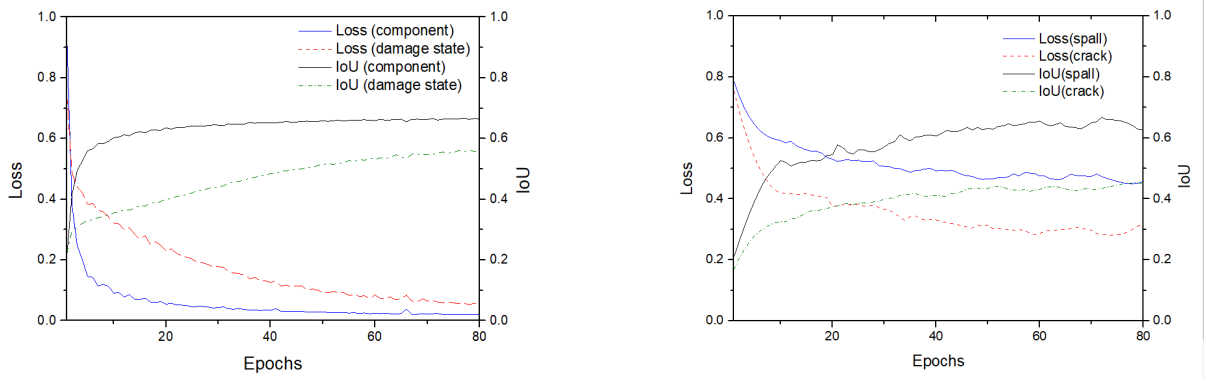
$$IoU = \frac{A \cap B}{A \cup B} = \frac{A \cap B}{A \cap B + |A - B| + |B - A|} \tag{3}$$

where the overlapping degree of prediction set A and ground truth set B is measured.

Mean IoU (mIoU) was also calculated by averaging the prediction accuracies among different categories, as defined in Eq. (4).

$$mIoU = \frac{1}{k}\sum_{i=0}^{k-1} \frac{p_{ii}}{\sum_{j=0}^{k-1} p_{ij} + \sum_{j=0}^{k-1} p_{ji} - p_{ii}} \tag{4}$$

where k equals the number of categories in each segmentation task where pixels can be classified as either 0 (background) or 1 (objective). The p_{ij} is the number of pixels belonging to class i but are mispredicted as class j, and p_{ii} and p_{ji} are also defined by that analogy. For example, p_{01} refers to the number of background pixels that are incorrectly identified as a target.



(a) (b)

Fig. 6 Training loss curves of (a) Model A (structural component and damage state) and (b) Model B (crack) and Model C (spall)

Table 4 Summary of task performance by mIoU values

Model	Component	Damage states	Crack	Spall
Model A	**0.9593**	**0.6374**	-	-
Model B	-	-	**0.3971**	-
Model C	-	-	-	**0.7010**
Single task in plain mode	0.9473	0.5855	0.3681	0.6054

4. Results and discussions

A total of three models were optimised based on different datasets with various main tasks, and the training records can be found in Fig. 6. In addition, single-task training of each sub-task in the dataset of plain mode was also carried out for comparative experiments.

Table 4 summarises mIoU values in the validation set from the proposed models and the results from the corresponding single-task counterpart. The proposed network achieved a high mIoU in component identification and damage state estimation and witnessed a performance booster when learning auxiliary tasks. For the small object segmentation tasks (spall and crack), training with the filtered datasets helped to increase the segmentation accuracy in small-size learning resources.

4.1 Multi-task learning with auxiliary tasks

Training with auxiliary tasks strengthened the main objectives. Notably, the damage estimation tasks benefited from the assistive learning of structural components and defect segmentation, with a mIoU enhancement from0.5855 of single-task training to 0.6374 of the proposed method.

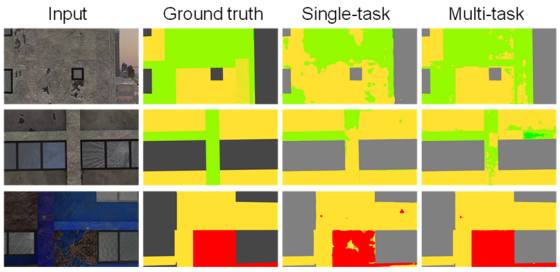


Fig. 7 Three representative results of damage state estimation outputted from single-task learning and multi-task learning with auxiliary tasks

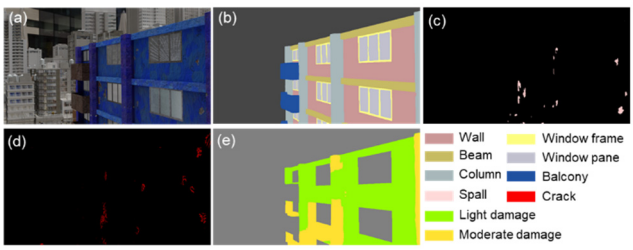


Fig. 8 Representative segmentation results of four sub-tasks: (a) input image, (b) structural component, (c) spall, (d) crack and (e) damage state

method. When evaluating the damage state, the model identifies structural components first. As indicated in Fig. 7, the proposed method can highlight the identified damage zone as integrity, while predictions from the single-task model tend to be coarse and disconnected. The estimation task further evaluates the injury degree based on a comprehensive acquisition of specific crack, spall, or exposed rebar. For instance, in Fig. 8, substantial areas of Fig. 8(e) overlap with that in component segmentation results in Fig. 8(b). Moreover, the region marked as moderate damage is closely associated with the presence of spalls and cracks (Figs. 8(c) and (d)).

Adding auxiliary tasks to single-task models, especially in soft parameter-sharing models, may occasionally slow down the prediction for the increment of parameter amount. To strengthen the performance with acceptable computation consumption, this study adopted hard sharing of parameters, where the decoders of assistive tasks diverged from the mainstream only at the end of the network. The trained models were implemented on the validation set and the average inference time was obtained by dividing the total processing time by the number of input images. The inference time of each input image (512×512) thus only increased on a neglectable scale (Table 5).

4.2 Data filtering

In the identification of building structural components, due to the building structural components occupying a large area in the overall photos, and the photos based on optical capture take buildings as the main capture objects, the difference between the foreground and the background is not obvious. However, in structural health monitoring, it is essential to detect the damaged area in the early stage of structural damage and prevent facilities from deficiency and further expansion of damages. The morphology of early damages is mainly composed of a series of small and micro features such as cracks, spalling, etc., which only account for a small proportion of the optical captured images, thus leading to extreme imbalance distribution of the sample. Apart from the three models shown above, single-task models in the plain mode were also trained as baselines to evaluate the data filtering strategy. U-Net was trained by the filtered spall and crack datasets, respectively, and outperformed the ones without data balancing. To reduce potential bias induced by the training process, hyperparameters of single-task models in plain mode are consistent with multi-task models.

Imbalanced sample distribution would bring bias to data-driven algorithms. The data filtering method was used to enhance the data equilibrium by discarding images containing many unwanted backgrounds. Apart from the three models shown above, single-task models in the plain

Table 5 Average inference time (ms) for single image with the size of 512×512

Model	Inference time (ms)
Multi-task model with auxiliary tasks	17
Single-task model	18

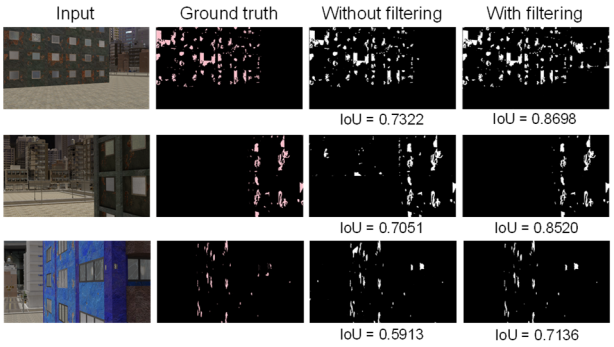


Fig. 9 Three representative spall segmentations of the models trained with and without data filtering

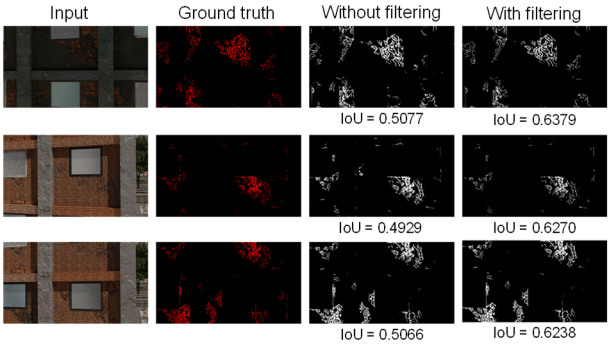


Fig. 10 Three representative crack segmentations of the models trained with and without data filtering

mode were also trained as baselines to evaluate the data filtering strategy. U-Net was trained by the filtered spall and crack datasets, respectively, and outperformed the ones without data balancing. To reduce potential bias induced by the training process, hyperparameters of single-task models in plain mode are consistent with multi-task models. Representative examples are plotted in Fig. 9 and Fig. 10, indicating the improvement of segmentation accuracy when training with filtered data. The crack segmentation IoU increased from 0.3681 to 0.3971 when using the filtered data. More remarkably, the spall prediction IoU was boosted from nearly 0.6054 to 0.7010 since the raw data possess a much higher sample imbalance.

The results show that the preprocessing of training data by data filtering method can not only effectively reduce the impact of imbalanced sample distribution, increased the prediction accuracy for the training model with preprocessed data, but also boost the training efficiency and reduce the training time by discarding the images with an insufficient positive sample than the specified threshold value of the training set.

4.3 Limitations

This study presents positive results from the experiments on multi-task CNN and data filtering strategies. However, due to the timeline of IC-SHM 2021, there are some limitations to be further addressed in future study. First, the single-input multi-task learning architecture constrains the customisation of datasets for each task. For instance, the crack segmentation task would benefit from data filtering while training with ad-hoc crack samples would largely bias the structural component and damage state tasks. Likewise, a re-sampled crack dataset would mislead the model in the learning of spall features. This issue might be addressed by resorting to special training strategies such as parameter freezing or transfer learning. Moreover, the extremely imbalanced rebar sample distribution disables the data filtering strategy. According to our experiments of rebar detection task, the filtered-out positive samples (image patches with rebar exposure) are not sufficient and diverse enough to optimise a deep neural network to be accurate and robust. Hence the work of rebar exposure detection is omitted in this study. We would appreciate solutions from our peers. In addition, in this study, the models were evaluated with the samples we randomly selected, not the newly-released blind test data. The results slightly changed when the models were tested with the blind test data, but the main conclusions keep unaltered. For example, for the blind test set, mIoU values of the structural component task are 0.9675 and 0.9395 for Model A and the counterpart single-task model.

5. Conclusions

In this project, to address the main challenge posed by highly correlated tasks in post-earthquake damage and component identification, multi-task learning with auxiliary tasks was adopted and evaluated. The proposed network is featured by independent FC branches corresponding to the five segmentation sub-objectives: component, spall, crack, rebar, and damage states. Joint loss functions were weighted to enable the specified main tasks - component identification and damage state estimation, to govern during training. The strategy to associate the two main tasks with the related defect images boosted segmentation performance. Notably, a mIoU improvement from 0.5855 to 0.6374 was achieved for damage state segmentation, which is the goal of post-earthquake building condition assessment. The additional challenge lies in the sample imbalance of small objects. Cropping and filtering were applied in the non-equilibrium spall and crack dataset to adjust the sample proportion. The network improved in efficiency from the filtered samples by implicitly strengthening feature learnings from the augmented data. Specifically, both the spall and crack segmentation witnessed mIoU increases of nearly 10%, from that of single-task training. Furthermore, the increase in computational cost resulting from auxiliary tasks was estimated based on the average inference time. No significant increase in inference time was observed, indicating that learning with auxiliary tasks is a highly cost-effective method to improve the performance of closely-related multiple tasks.

Code availability

This project is based on the open-access QuakeCity Dataset (https://sail.cive.uh.edu/quakecity/). The authors

highly appreciate the open-sourcing efforts and would like to make all codes involved in this research public via https://github.com/Monash-Civil-CV-Team/IC-SHM2-2021.

Acknowledgments

This study was supported by Monash University for the scholarships and the high-performance computation platform sponsored by the 2022 AWS Cloud Computing Interdisciplinary Seed Project. The authors appreciate the organization committee of IC-SHM 2021, the University of Illinois at Urbana-Champaign, and the Harbin Institute of Technology, for generously providing the invaluable data. The authors also would like to thank the chairs of IC-SHM 2021, Prof. Billie F. Spencer Jr. and Prof. Hui Li, for leading this competition.

References

Ando, R. and Zhang, T. (2005), "A framework for learning predictive structures from multiple tasks and unlabeled data.", *J. Mach. Learn. Res.*, **6**, 1817-1853.

Azimi, M., Eslamlou, A.D. and Pekcan, G. (2020), "Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review", *Sensors*, **20**(10), 2778. https://doi.org/10.3390/s20102778

Bao, Y., Chen, Z., Wei, S., Xu, Y., Tang, Z. and Li, H. (2019), "The state of the art of data science and engineering in structural health monitoring" *Engineering*, **5**(2), 234-242. https://doi.org/10.1016/j.eng.2018.11.027

Baxter, J. (1997), "A Bayesian/information theoretic model of learning to learn via multiple task sampling", *Mach. Learn.*, **28**(1), 7-39. https://doi.org/10.1023/A:1007327622663

Blaikie, P., Cannon, T., Davis, I. and Wisner, B. (2014), *At risk: Natural Hazards, People's Vulnerability and Disasters*, Routledge.

Cha, Y.-J., Choi, W. and Büyüköztürk, O. (2017), "Deep learning-based crack damage detection using convolutional neural networks", *Comput.-Aided Civil Infrastr. Eng.*, **32**(5), 361-378. https://doi.org/10.1111/mice.12263

Cha, Y.-J., Choi, W., Suh, G., Mahmoudkhani, S. and Büyüköztürk, O. (2018), "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types", *Comput.-Aided Civil Infrastr. Eng.*, **33**(9), 731-747. https://doi.org/10.1111/mice.12334

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A.L. (2016), "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs", *IEEE Transact. Pattern Anal. Mach. Intell.*, **40**(4), 834-848. https://doi.org/10.1109/TPAMI.2017.2699184 Retrieved from: https://ui.adsabs.harvard.edu/abs/2016arXiv160600915C

Choi, W. and Cha, Y.-J. (2020), "SDDNet: Real-time crack segmentation", *IEEE Transact. Indust. Electron.*, **67**(9), 8016-8025. https://doi.org/10.1109/tie.2019.2945265

Doocy, S., Daniels, A., Packer, C., Dick, A. and Kirsch, T.D. (2013), "The human impact of earthquakes: a historical review of events 1980-2009 and systematic literature review", *PLoS currents*, 5. https://doi.org/10.1371/currents.dis.67bd14fe457f1db0b5433a8ee20fb833

Gao, Y. and Mosalam, K.M. (2018), "Deep transfer learning for image-based structural damage recognition", *Comput.-Aided Civil Infrastr. Eng.*, **33**(9), 748-768. https://doi.org/10.1111/mice.12363

Guo, J., Wang, Q., Li, Y. and Liu, P. (2020), "Façade defects classification from imbalanced dataset using meta learning-based convolutional neural network", *Comput.-Aided Civil Infrastr. Eng.*, **35**(12), 1403-1418. https://doi.org/10.1111/mice.12578

He, K., Zhang, X., Ren, S. and Sun, J. (2014), "Spatial pyramid pooling in deep convolutional networks for visual recognition", *IEEE Transact. Pattern Anal. Mach. Intell.*, **37**(9), 1904-1916. https://doi.org/10.1007/978-3-319-10578-9_23

He, K., Zhang, X., Ren, S. and Sun, J. (2016), "Deep residual learning for image recognition", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Hoskere, V., Narazaki, Y., Hoang, T.A. and Spencer Jr, B.F. (2018), "Towards automated post-earthquake inspections with deep learning-based condition-aware models", arXiv preprint arXiv:1809.09195. https://doi.org/10.48550/arXiv.1809.09195

Hoskere, V., Narazaki, Y., Hoang, T.A. and Spencer Jr, B.F. (2020), "MaDnet: multi-task semantic segmentation of multiple types of structural materials and damage in images of civil infrastructure", *J. Civil Struct. Health Monitor.*, **10**(5), 757-773. https://doi.org/10.1007/s13349-020-00409-0

Hoskere, V., Narazaki, Y. and Spencer Jr, B.F. (2022), "Physics-based graphics models in 3D synthetic environments as autonomous vision-based inspection testbeds", *Sensors*, **22**(2), 532. https://doi.org/10.3390/s22020532 Retrieved from: https://www.mdpi.com/1424-8220/22/2/532

Lee, K., Lee, S. and Kim, H.Y. (2022), "Bounding-box object augmentation with random transformations for automated defect detection in residential building façades", *Automat. Constr.*, **135**. https://doi.org/10.1016/j.autcon.2022.104138

Li, Y., Zhang, X. and Chen, D. (2018), "Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes", *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June, pp. 1091-1100.

Li, J., Wang, Q., Ma, J. and Guo, J. (2022a), "Multi-defect segmentation from façade images using balanced copy–paste method", *Comput.-Aided Civil Infrastr. Eng.*, **37**(11), 1434-1449. https://doi.org/10.1111/mice.12808

Li, Z., Huang, M., Ji, P., Zhu, H. and Zhang, Q. (2022b), "One-step deep learning-based method for pixel-level detection of fine cracks in steel girder images", *Smart Struct. Syst., Int. J.*, **29**(1), 153-166. https://doi.org/10.12989/sss.2022.29.1.153

Mehdi, Z. and Nazmazar, B. (2013), "Van, Turkey earthquake of 23 october 2011, mw 7.2; an overview on disaster management", *Iran. J. Public Health*, **42**(2), 134.

Milletari, F., Navab, N. and Ahmadi, S.A. (2016), "V-net: Fully convolutional neural networks for volumetric medical image segmentation", *Proceedings of the 4th International Conference on 3D Vision (3DV)*, pp. 565-571.

Narazaki, Y., Hoskere, V., Hoang, T.A., Fujino, Y., Sakurai, A. and Spencer Jr, B.F. (2020), "Vision-based automated bridge component recognition with high-level scene consistency", *Comput.-Aided Civil Infrastr. Eng.*, **35**(5), 465-482. https://doi.org/10.1111/mice.12505

Narazaki, Y., Hoskere, V., Yoshida, K., Spencer Jr, B.F. and Fujino, Y. (2021), "Synthetic environments for vision-based structural condition assessment of Japanese high-speed railway viaducts", *Mech. Syst. Signal Process.*, **160**. https://doi.org/10.1016/j.ymssp.2021.107850

Pan, X. and Yang, T.Y. (2020), "Postdisaster image-based damage detection and repair cost estimation of reinforced concrete buildings using dual convolutional neural networks", *Comput.-Aided Civil Infrastr. Eng.*, **35**(5), 495-510.

https://doi.org/10.1111/mice.12549
Ruder, S. (2017), "An overview of multi-task learning in deep neural networks", arXiv:1706.05098.
https://doi.org/10.48550/arXiv.1706.05098
Retrieved from:
https://ui.adsabs.harvard.edu/abs/2017arXiv170605098R
Søgaard, A. and Bingel, J. (2017), "Identifying beneficial task relations for multi-task learning in deep neural networks", *arXiv preprint arXiv*:*1702.08303*.
https://doi.org/10.48550/arXiv.1702.08303
Spencer Jr, B.F. and Li, H. (2021), *The 2nd International Competition for Structural Health Monitoring* (*IC-SHM*, *2021*).
Retrieved from:
http://sstl.cee.illinois.edu/ic-shm2021/The%202nd%20International%20Project%20Competition%20-%20FINAL.pdf
Spencer Jr, B.F., Hoskere, V. and Narazaki, Y. (2019), "Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring", *Engineering*, **5**(2), 199-222.
https://doi.org/10.1016/j.eng.2018.11.030
Vandenhende, S., Georgoulis, S., Van Gansbeke, W., Proesmans, M., Dai, D. and Van Gool, L. (2020), "Multi-task learning for dense prediction tasks: A survey", *IEEE Transact*. *Pattern Anal*. *Mach*. *Intell*., **44**(7), 3614-3633.
https://doi.org/10.1109/TPAMI.2021.3054719
Retrieved from:
https://ui.adsabs.harvard.edu/abs/2020arXiv200413379V
Yang, G., Li, G., Pan, T., Kong, Y., Wu, J., Shu, H., Luo, L., Dillenseger, J.L., Coatrieux, J.L., Tang, L. and Zhu, X. (2018), "Automatic segmentation of kidney and renal tumor in ct images based on 3d fully convolutional neural network with pyramid pooling module", *Proceedings of the 24th International Conference on Pattern Recognition* (*ICPR*), Beijing, China, August, pp. 3790-3795.
https://doi.org/10.1109/ICPR.2018.8545143
Yu, F. and Koltun, V. (2015), "Multi-scale context aggregation by dilated convolutions", arXiv preprint arXiv:1511.07122.
https://doi.org/10.48550/arXiv.1511.07122
Zhang, C.-H. and Huang, J. (2008), "The sparsity and bias of the Lasso selection in high-dimensional linear regression", *Annals Statist*., **36**, 1567-1594. https://doi.org/10.1214/07-AOS520
Zhao, H., Shi, J., Qi, X., Wang, X. and Jia, J. (2017), "Pyramid scene parsing network", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July, pp. 6230-6239.