South Dakota State University

Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange

Electronic Theses and Dissertations

2021

Cascaded Deep Learning Network for Postearthquake Bridge Serviceability Assessment

Youjeong Jang
South Dakota State University

Follow this and additional works at: https://openprairie.sdstate.edu/etd

 Part of the Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Recommended Citation

Jang, Youjeong, "Cascaded Deep Learning Network for Postearthquake Bridge Serviceability Assessment" (2021). *Electronic Theses and Dissertations*. 5707.
https://openprairie.sdstate.edu/etd/5707

This Thesis - Open Access is brought to you for free and open access by Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of Open PRAIRIE: Open Public Research Access Institutional Repository and Information Exchange. For more information, please contact michael.biondo@sdstate.edu.

CASCADED DEEP LEARNING NETWORK FOR POSTEARTHQUAKE BRIDGE SERVICEABILITY ASSESSMENT

BY

YOUJEONG JANG

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Computer Science

South Dakota State University

2021

THESIS ACCEPTANCE PAGE

Youjeong Jang

This thesis is approved as a creditable and independent investigation by a candidate for the master's degree and is acceptable for meeting the thesis requirements for this degree. Acceptance of this does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department.

Kwanghee Won, PhD

Advisor Date

Mostafa Tazarv, PhD

Advisor Date

Siddharth Suryanarayanan, PhD
Department Head Date

Nicole Lounsbery, PhD
Director, Graduate School Date

ACKNOWLEDGEMENTS

A debt of gratitude is owed by my academic advisor Dr. Kwanghee Won for my master's degree program. He provided me a research opportunity and continuous guidance throughout this study.

I would like to express my special thanks of gratitude to Dr. Mostafa Tazarv for giving this research opportunity and continued support of this project.

The work presented in this thesis conducted with support from South Dakota State University and the National Center for Transportation Infrastructure Durability and Life-Extension (TriDurLE), a University Transportation Center (UTC) funded by the U.S. Department of Transportation. The contents of this report reflect the views of the authors, who are responsible for the facts and accuracy of the information presented. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Finally, I am thankful to my family for their encouragement and support during my study.

CONTENTS

ACKNOWLEDGEMENTS iii

ABBREVIATIONS vi

LIST OF FIGURES vii

LIST OF TABLES ix

ABSTRACT x

Chapter 1. Introduction 1

1.1 Introduction 1

1.2 Objectives and Scope 4

1.3 Document Outline 5

Chapter 2. Literature review 6

2.1 Convolutional Neural Network (CNN) 6

2.2 Image classification in deep learning 9

2.3 Image segmentation in deep learning 13

Chapter 3. Materials and method 16

3.1 Overview 16

3.2 Dataset 21

3.2.1 Crack data 22

3.2.2 Bridge column, spalling and exposed rebar data 22

3.3 Bridge column, spalling and exposed rebar detection 23

3.4 Crack detection 24

Chapter 4. Experimental and results 27

4.1 Damage Detection Results 27

4.2 Evaluation of Performance 40

Chapter 5. Summary and conclusions 44

Literature Cited 45

ABBREVIATIONS

CNN	convolutional neural network
FCN	fully connected network
PR	precision and recall
px	pixel
RC	reinforced concrete
SHM	structural health monitoring

LIST OF FIGURES

Figure 1. 1 Post earthquake bridge column damage examples 2

Figure 1. 2 Damage detection results using FCN [20] 4

Figure 2. 1 Visual concept of neuron 6

Figure 2. 2 Concept of convolution network 7

Figure 2. 3 Procedure of a two-dimensional CNN [27] 8

Figure 2. 4 Standard Convolutional Neural Network Architecture [28] 9

Figure 2. 5 MobileNet v1 convolution architecture [31] 11

Figure 2. 6 MobileNet v2 residual blocks [21] 12

Figure 2. 7 Fully Convolution Networks [35] 13

Figure 2. 8 Faster R-CNN architecture [36] 14

Figure 2. 9 The overall network architecture of Mask R-CNN [18] 15

Figure 3. 1 Column axial direction 17

Figure 3. 2 Samples of RC Bridge Column Condition at DS-1 [45], [46] 17

Figure 3. 3 Samples of RC Bridge Column Condition at DS-2 [45], [46] 18

Figure 3. 4 Samples of RC Bridge Column Condition at DS-3 [45], [46] 18

Figure 3. 5 Samples of RC Bridge Column Condition at DS-4 [45], [46] 18

Figure 3. 6 Samples of RC Bridge Column Condition at DS-5 [45], [46] 19

Figure 3. 7 Overview of proposed cascaded damage detection network 19

Figure 3. 8 Samples of data augmentation 21

Figure 3. 9 Sample image of crack dataset (a,b are non-crack and c, d are crack) 22

Figure 3. 10 Sample image of annotation 22

Figure 3. 11 Target deficiency object analysis 23

Figure 3. 12 Crack detection analysis 25

Figure 3. 13 (a) Segmented crack patches, (b) Measuring angle result 25

Figure 3. 14 (a) Segmented crack patches, (b) Measuring angle result 26

Figure 4. 1 Result sample of DS-1 (1) 29

Figure 4. 2 Result sample of DS-1 (2) 30

Figure 4. 3 Result sample of DS-2 (1) 31

Figure 4. 4 Result sample of DS-2 (2) 32

Figure 4. 5 Result sample of DS-3 (1) 33

Figure 4. 6 Result sample of DS-3 (2) 34

Figure 4. 7 Result sample of DS-4 (1) 35

Figure 4. 8 Result sample of DS-4 (2) 36

Figure 4. 9 Result sample of DS-4 (3) 37

Figure 4. 10 Result sample of DS-5 (1) 38

Figure 4. 11 Result sample of DS-5 (2) 39

Figure 4. 12 Intersection of Union (IoU) 40

Figure 4. 13 Real scene test result 42

Figure 4. 14 Samples of failed detection case 43

LIST OF TABLES

Table 3. 1 Proposed Damage States for RC Bridge Columns 16

Table 4. 1 Analysis of figure 4.1..... 29

Table 4. 2 Analysis of figure 4.2..... 30

Table 4. 3 Analysis of figure 4.3..... 31

Table 4. 4 Analysis of figure 4.4..... 32

Table 4. 5 Analysis of figure 4.5..... 33

Table 4. 6 Analysis of figure 4.6..... 34

Table 4. 7 Analysis of figure 4.7..... 35

Table 4. 8 Analysis of figure 4.8..... 36

Table 4. 9 Analysis of figure 4.9..... 37

Table 4. 10 Analysis of figure 4.10..... 38

Table 4. 11 Analysis of figure 4.11..... 39

Table 4. 12 Evaluation results of target deficiency detection for each component 41

Table 4. 13 Evaluation results of crack classification network 42

ABSTRACT

CASCADED DEEP LEARNING NETWORK FOR POSTEARTHQUAKE BRIDGE SERVICEABILITY ASSESSMENT

YOUJEONG JANG

2021

Damages assessment of bridge is important to derive immediate response after severe events to decide serviceability. Especially, past earthquakes have proven the vulnerability of bridges with insufficient detailing. Due to lack of a national and unified post-earthquake inspection procedure for bridges, conventional damage assessments are performed by sending professional personnel to the onsite, detecting visually and measuring the damage state. To get accurate and fast damage result of bridge condition is important to save not only live but also costs.

There have been studies using image processing techniques to assess damage of bridge column without sending individual to onsite. Convolutional neural networks (CNNs) have shown state-of-art results in object detection and image classification tasks. This study proposed cascaded deep learning network for post-earthquake bridge serviceability assessment. Major target deficiency components (crack, spalled area, transverse bar, and longitudinal bar) were used to determine the proposed damage states to assess serviceability of bridge. Cascaded network is composed by Mask R-CNN and MobileNet v2 which have been proved as powerful network for each instance segmentation and image classification.

In this study, proposed network successfully detected target deficiency components and measured each damage state by following 5 stages. Column area is detected as first step, and counting exposed bars, finding maximum distance in spalled region within column area are followed to decide damage state. To determine deficiency of crack in bridge column, crack patch classification module is attached in proposed network. Counting diagonal and horizontal cracks with angle measurement are used to analyze type of cracks.

CHAPTER 1. INTRODUCTION

1.1 Introduction

Damages assessment of bridge is important to derive an immediate response after severe events to decide serviceability. Especially, past earthquakes have proven the vulnerability of bridges with insufficient detailing. Even modern reinforcement concrete (RC) bridge columns, which are detailed properly to serve as the main source of ductility in a bridge, may exhibit cover spalling, exposure of transverse and longitudinal bars, and buckling of longitudinal bars. To completely assess a bridge column performance during an earthquake, both capacity of and demand on the columns are needed, which are usually in the form of displacements.

Due to lack of a national and unified post-earthquake inspection procedure for bridges, conventional damage assessments are performed by sending professional personnel to the onsite, detecting visually and measuring the damage state. Although human-based assessment procedure may be effective, this procedure can take a lot of time and days after events and can miss critical time for rescue operations. And the correctness and accurate records of the decision for damaged bridge may be different from subjectivity of the inspector. Also, several bridge structural health monitoring (SHM) [1-2] are capable to detect large-scale damages in structure but used to be required with sensors or other instrumentals which are hard to install and not cost efficient.

To get an accurate and fast damage results of a bridge condition is important to save not only lives but also costs. With the increasing demand of a computer vision-based method, automated damage detection has been developed more to help and make

decision faster than past. However, as figure 1.1 shows, the task is non-trivial because in most sites, the damage appears with different shape and size, and usually mixed with noisy background which is hard to detect the parts which are needed to decide damage level.

(a) (b)

Figure 1. 1 Post earthquake bridge column damage examples
(Failure due to spalling and exposed bars [3], (b) Failure due to spalling [4])

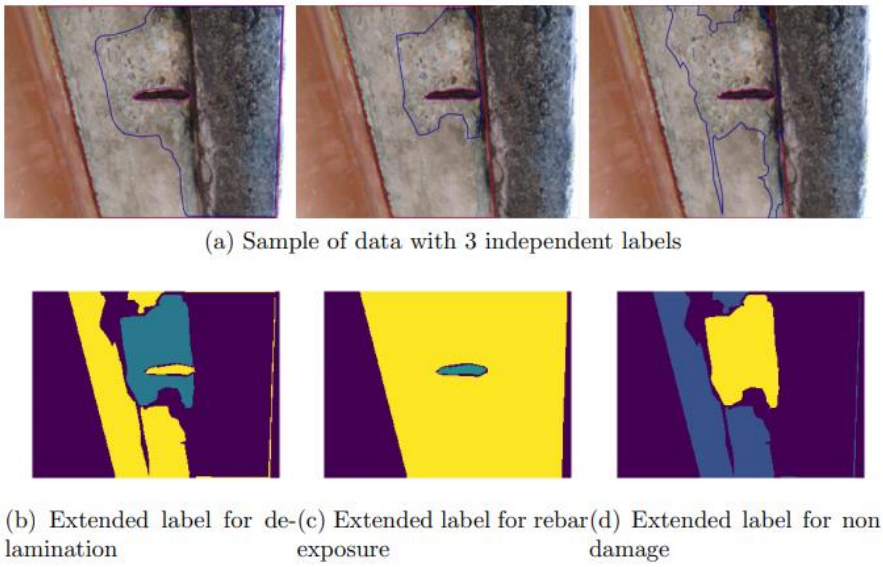
In an early stage, there have been studies with heuristic filters to detect objects. Image processing methods with edge detection [5-6], threshold methods [7] and traditional detectors [8-9] were very popular in object detection. Paal et al. [10] presented a computer vision-based method for determination of damage states of the column by localizing and quantifying each component (crack, spalling and exposed steel bar) properties of distinct textures of the region with Canny operator. Nishikawa et al. [11] applied the multiple sequential image filtering for estimating property and detection. Yeum et al. [12] used region localization of object detection and filtering to detect fatigue cracks in steel bar. But those heuristic methods are time-consuming and cannot be operated in noisy background image.

Due to those limitations, using deep learning-based techniques have been studied and applied to damage detection. Deep learning-based methods have been known for improving traditional vision-based damage detection by extending not only one component but also multiple defects. Image classification methods have been used for damage detection. Kim et al. [13] proposed the classification models using convolutional neural network (CNN) and speeded-up robust features (SURF) for crack detection. AlexNet [14] and GoogleNet [15] which both are state-of-art neural network models, have been applied to classify each crack and spalled image. Object detection methods have recently studied for a damage detection tasks. Object detection methods have improved an image classification tasks which classify entire images. Yeum et al. [16] used Regions with CNN features (R-CNNs) for detecting and indicating objects with bounding boxes. Cha et al. [17] used Faster R-CNN which developed with a region-based method for detecting different shape and size of delamination.

Besides those object detection methods, in our study, it is also important to quantify the damage to derive an accurate deficiency level. Semantic segmentation with object detection methods has been used to not only detect object with bounding rectangular box but also measure shape of the damage. Mask R-CNN [18] has been adapted for detection of cracks, spalling and exposed bars. This method not only segments detected objects, but also provides the exact location of each instance in image [19]. Using fully convolutional networks (FCN) [20] also have been adapted to segment damages and based method to implement Mask R-CNN.

We notice the importance of detecting damage in structural level since this task can be a mixture of classification, object detection and semantic segmentation tasks. And

deep learning-based methods have been studied to get state-of-art result for analyzing damages in bridge column. In this study, we proposed cascaded damage detection network with Mask R-CNN [18] for segmenting major damage components (column, spalling and exposed bars) and MobileNet v2 [21] for detecting and classifying cracks. Also, image augmentation techniques applied to enhance the network training and testing results will be explained detailed in following chapeter.



(a) Sample of data with 3 independent labels

(b) Extended label for de-lamination (c) Extended label for rebar exposure (d) Extended label for non damage

Figure 1. 2 Damage detection results using FCN [20]

1.2 Objectives and Scope

This research proposes a cascaded damage detection network with Mask R-CNN [18] and MobileNet v2 [21] for assessing post-event serviceability of RC bridge column. It is an important study to understand deep learning-based model and high performance of object detection and instance segmentation problem. Since object detection module with Mask R-CNN is not sufficient to detect small cracks in from the entire image and

there is no evaluation criteria of quantifying width of crack, we adapt MobileNet v2 [21] to improve performance results in detecting deficiency components.

This main product will accomplish this these objectives: (1) all the test datasets will be obtained by SDSU, (2) propose cascaded deep learning-based network to detect each deficiency components, and (3) analyzing object detection and instance segmentation results to determine final damage state of bridge serviceability.

1.3 Document Outline

In Chapter 1, there is presented the study and the scope of the work. Chapter 2 reviews the deep learning image segmentation and classification for our proposed cascaded network. In Chapter 3, detailed our cascaded network methods and data preparation will be addressed. And detection experimental, and results are shown in chapter 4 with each component (column, crack, spalling, longitudinal/transverse bars) detection phase and bridge column damage state determination. Analysis for those results and summary of this study will be discussed in Chapter 5. It is an important study for future damage assessment systems and object detection and instance segmentation problem.

CHAPTER 2. LITERATURE REVIEW

In this chapter, major literatures which describe how convolutional neural network developed and worked in deep learning, are reviewed. And deep learning-based segmentation and classification models including Mask R-CNN [18] and MobileNet v2 [21], which are two main components of our proposed cascaded model.

2.1 Convolutional Neural Network (CNN)

Convolution Neural Network (CNN) has been making a great achievement in many applications. CNN is a class of deep neural network in deep learning and has been used to solve not only computer vision tasks but also other tasks like natural language processing or time-series forecasting. CNN has emerged from Artificial Neural Networks (ANN) [22], which proposed concept of neurons.

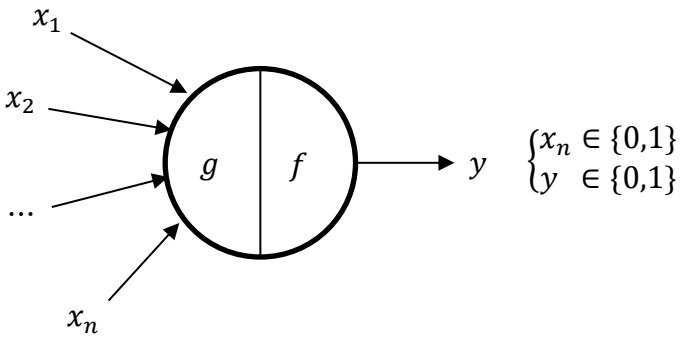


Figure 2. 1 Visual concept of neuron

Figure 2.1 shows the concept of neuron, and g takes an input $x_1 \cdots x_n$ and performs calculation which aggregates input values while f is decision function deriving y value between 0 to 1. Neural network is consisted of multiple neurons and functions of aggregation and decision. Single-layer perceptron network [23] and multi-layer

perceptron network [24] have been introduced to develop more complex networks to solve problematic tasks. CNN has been constructed based on those developments and many studies. LeCun et al. [25] showed performance of CNN in classifying hand-writing digit dataset and the term “convolution” was first used. LeNet-5 [26] is one of the earliest CNNs and the network was shallow model which only has 1 or 2 hidden layers. After advent of LeNet-5 [26], many state-of-art networks has been inspired and created to solve complex classification and object detection tasks. Chapter 2.2 and 2.3 will introduce various version of CNN in classification and object detection tasks. To help understanding future chapters, figure 2.2 and figure 2.3 show visualized concept of a two-dimensional CNN which has been developed from basic concept of neuron in figure 2.1.

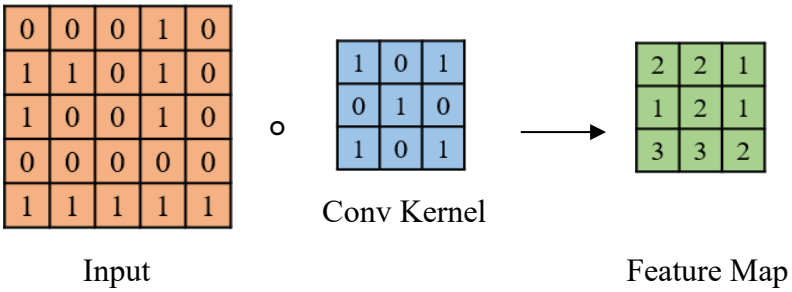


Figure 2. 2 Concept of convolution network

CNN is a feedforward network to extract features with convolutional formations. In figure 2.2 above, there is a general 3×3 convolutional kernel and 5×5 input image. CNN performs element-wise multiplication with input and convolution kernel and the results is called as a feature map. CNN kernels represent a different receptor that extract and derive useful features from the input source. Li et al. [27] stated many advantages of using CNN. First, there is a local connection which is different from previous multi-layer

networks and these connections are very cost and time efficient by reducing the number of parameters. Second, a group of local connection shares the same weights, which accelerates to reduce calculation process. And lastly, pooling layer after feature map can reduce dimension of feature maps. Down-sampling in pooling layer can reduce less-important data and remains only useful information.

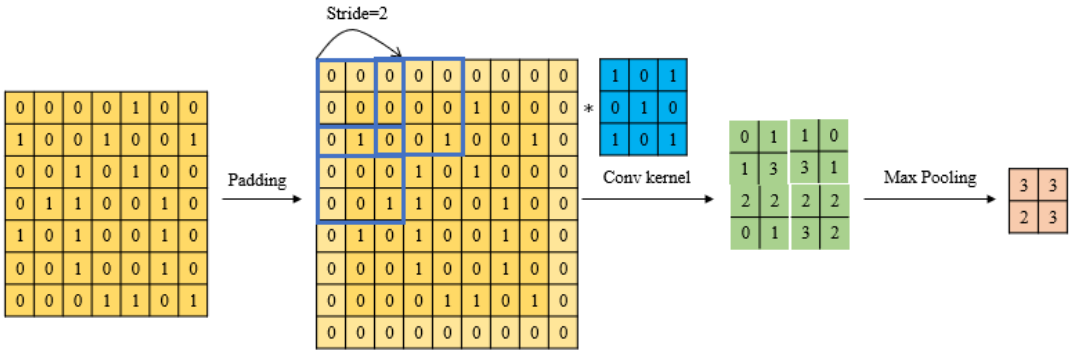


Figure 2. 3 Procedure of a two-dimensional CNN [27]

Figure 2.3 above is a sample of procedure of a two-dimensional CNN from [27]. When we set the certain size of kernel, the border information can be lost. So, padding with certain value (in this example is 0) is applied to keep the border information and stride is applied to reduce calculation steps in convolving. After each convolution operation, it derives high dimension of features. But these feature maps can cause overfitting which has possibility of ending up where network only works with training data. So, pooling layer (down-sampling) is used to reduce overlapping information and in this example, max pooling which only keeps the maximum value in 2×2 window is introduced. And those hyperparameters (kernel size, max pooling window size, etc.) are designed and adjusted by each network configuration.

2.2 Image classification in deep learning

CNNs have achieved a great performance in image classification tasks. In figure 2.4, He et al. [28] described standard convolutional neural network architecture. Image as an input source goes through the first convolutional layer and reduce the dimensionality with pooling layer. As explained in previous section, the number of layer and dimension of convolutional kernel can be different and adjusted. Also depending on which architecture chose, the number of convolutional layers is different. Figure 2.4 shows the standard convolutional neural network architecture. There are two convolutional layers and two pooling layers, and the multiplication operation steps are same as explained in section 2.1 with figure 2.3. Feature maps from last pooling layer, are transferred to fully connected layer which flattens those feature maps to make one dimension before going to activation (output) layer. In output layer, there are a number of states which corresponds the number of classes and activation functions like sigmoid or softmax are applied to determine final value of the input.

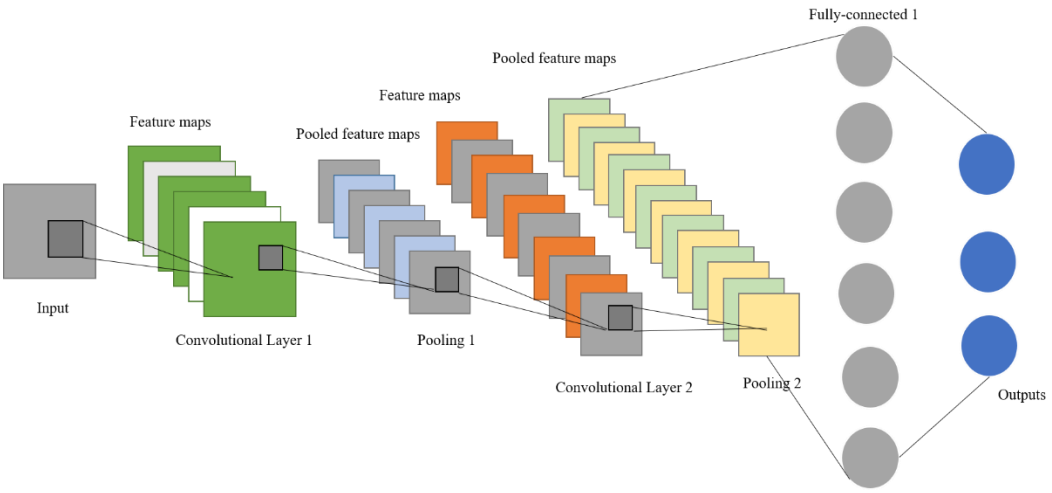


Figure 2. 4 Standard Convolutional Neural Network Architecture [28]

AlexNet [14] has been proposed in 2012, which won the first place in the ImageNet2012 competition. This network is composed with 5 convolutional layers and three fully connected layers. Also, AlexNet addressed the gradient vanishing problem which resulted no learning of models and introduced dropout techniques to resolve this issue. VGGNets [29] won the ImageNet2014 competition by building deep convolutional neural network. VGGNets have multiple series of models, VGG-14, VGG-16 and VGG-19. Those numbers state the number of convolutional layers and showed improved performance by building deep layers. Inception network series [15, 30-32] introduced batch normalization to make network more stable and won the ILSVRC 2014 image classification algorithms. He et al. [33] proposed Residual Network in 2016 and outperformed the Inception network performances. Inception network and Residual network has contributed to Deep Neural Networks (DNN) and those proposed methods are outperforming previous shallow networks. But those DNN models have large calculations which can derive time consuming procedure.

To resolve these problems, MobileNets have been proposed as a lightweight models. MobileNet v1 [34] introduced depth-wise separable convolutions which decompose standard convolutions into depth-wise to reduce number of channels of each layer. Figure 6 shows MobileNet v1 proposed method and figure 2.5 (a) shows standard convolution filters. The standard convolving process is replaced by depth-wise convolution and pointwise convolution in figure 2.5 (b) and figure 2.5 (c). M indicates the number of input channels and N is number of output channels. And $D_k \times D_k$ is dimension of kernel and $D_F \times D_F$ is dimension of feature map. By writing down the equations for calculating the number of parameters used in convolution multiplication,

standard convolution has cost of (1) from figure 2.5 (a).

$$D_k \cdot D_k \cdot M \cdot N \cdot D_F \cdot D_F \tag{1}$$

$$D_k \cdot D_k \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F \tag{2}$$

But, by decomposing standard convolution with depth-wise and pointwise, the network computation cost can be (2) which can reduce number of computations significantly.

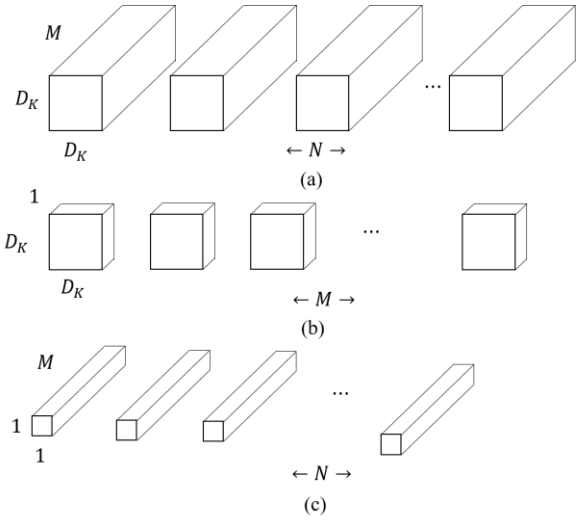


Figure 2. 5 MobileNet v1 convolution architecture [31]
(a) Standard convolution filters, (b) depth-wise convolution and
(c) point-wise convolution

MobileNet v2 [21] has improved this previous model by introducing inverted residual blocks. Figure 2.6. Shows how MobileNet v2 residual blocks are designed. These blocks widen the network using 1×1 convolution (pointwise) and following 3×3 depth-wise convolution reduces number of parameters. Afterwards, another point-wise convolution squeezes the network to match the initial number of channels. And residual connection prevents the performance of inverted block which can lose information from the activation function ReLU. So, the authors put a linear output where

last convolutional layer before adding with initial input. Both networks have achieved state-of-art results in ImageNet classification dataset by stating Top-1 accuracy as 71.8, 70.9 each (v1, v2) and Top-5 accuracy as 91.0, 89.9.

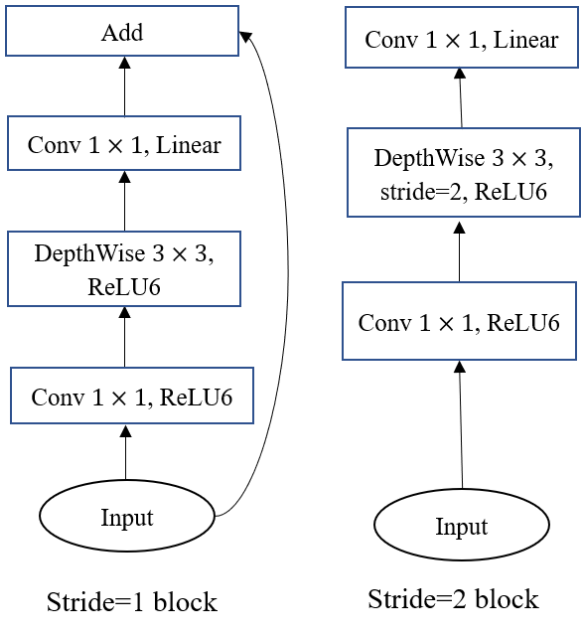


Figure 2. 6 MobileNet v2 residual blocks [21]

2.3 Image segmentation in deep learning

In this study, it is important to not only evaluate and classify those damage elements, but also to detect and quantify damages such as bridge column, spalling and rebar exposure. With the rapid development in deep learning, are introduced to address the problems existing in traditional architectures.

Fully Convolution Networks [35] was proposed for semantic image segmentation tasks. FCNs is composed of convolutional layers and can be applied to multiple size of images. Figure 2.7 shows the FCNs structure and how this network was trained for end-to-end pixelwise prediction and supervised pre-training. This method has been used in a variety of segmentation problems like brain tumor detection.

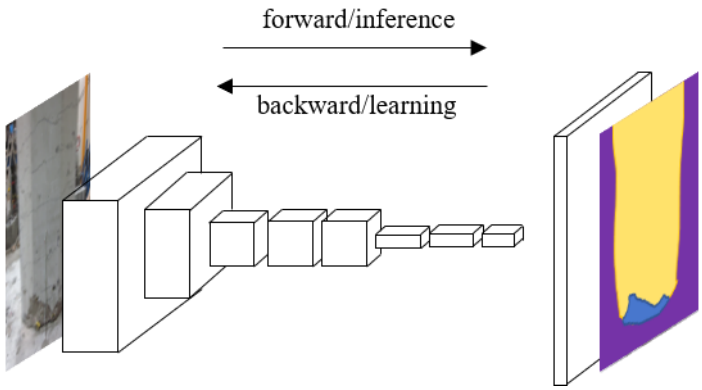


Figure 2. 7 Fully Convolution Networks [35]

Region-based convolution network (R-CNN) and its extensions (Fast R-CNN, Faster R-CNN and Mask R-CNN) have shown state-of-art results in object detection. Faster R-CNN [36] proposed region proposal network (RPN) to propose interest of region and derive bounding box candidates. This RPN network extract region of interest (RoI) and RoIPool layer in figure 2.8, calculates features from those RoIs and classify object and regress bounding boxes.

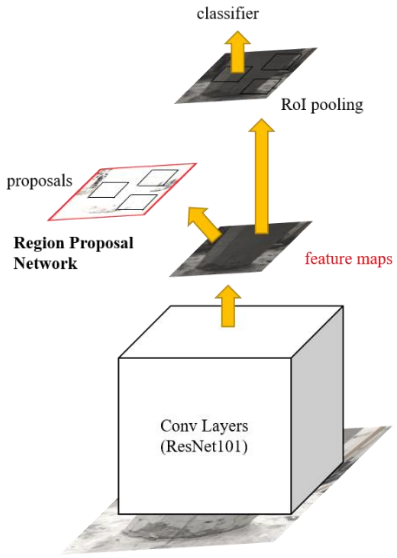


Figure 2. 8 Faster R-CNN architecture [36]

Mask R-CNN [18] showed state-of-art results in instance segmentation tasks which detect and classify each object of interest in image. Mask R-CNN are composed with 3 stages: region proposal, classification, and segmentation. In first stage, network takes an image as an input and extracts features with back- bone CNN network, high feature extractor ResNet101 [33]. The input image size is 1024×1024 and image can be resized with keeping ratio of original size. And the Region Proposal Network (RPN) select candidate areas for objects in an image from the features map which is the output of backbone network. And the selected candidate areas are called “anchor boxes” and each box is extracted with different aspect ratios and scored. Scored boxes indicate likelihood of containing object. If the box does have low score which means less possibility to contain object, then the RPN refines size and ratio of anchor boxes so the network can find better fit of object. After scoring, the boxes go to the classification and

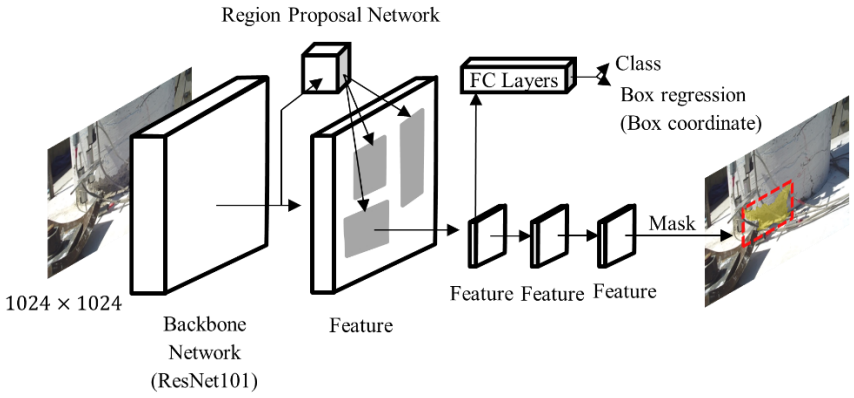


Figure 2. 9 The overall network architecture of Mask R-CNN [18]

bounding box regression module to classify each object in the box and get coordinate of box. The classification module classifies object into $n + 1$ classes which n is the number of class and background class. And the box regression module operates similar as box refinement in RPN network, but it is more fitted to each object and detailed to get exact bounding coordinate which states location of each object in an image. Lastly, the mask network takes selected and classified boxes from the classification network and generate masks to indicate each instance (object) in the image and estimate shape of object from the previous procedure. And mask network process pixel-to-pixel classification since this network should represent estimated shape and outline of the object. According to original paper [30], this branch works regardless of classification network accuracy, which means it is not affected by instance's class.

CHAPTER 3. MATERIALS AND METHOD

3.1 Overview

In this study, we propose cascaded damage detection network with Mask R-CNN [18] and MobileNet v2 [21] for assessing post-event serviceability of RC bridge column. To successfully assess and quantify serviceability of the bridge, we design and define damage states for RC bridge column in Table 1. Based on the review of past studies on RC column damage definitions and available RC column performance database [37-43], a new damage state (DS) definition but consistent with past studies [40-41, 44] is recommended for RC bridge columns to be used in our proposed model.

Table 3. 1 Proposed Damage States for RC Bridge Columns

Damage State	**Qualitative Performance Description**	**Quantitative Damage Description for Computer Vision**
1	Hairline cracks	Horizontal cracks each with an angle ($\|\theta\| > 80°$) (**Fig. 3.2**)
2	Theoretical first yielding of longitudinal bars	At least three diagonal cracks each with an angle of $\|\theta\| < 70°$ (**Fig. 3.3**)
3	Extensive cracks and spalling	Length of spalled region in any direction at any direction at any column face is greater than $0.1D_{col}$ but smaller than $0.3D_{col}$ (**Fig. 3.4**)
4	Visible transverse and/or longitudinal reinforcement	Length of spalled region in any direction at any column face is greater than $0.5D_{col}$ and detect one transverse bar and/or one longitudinal bar (**Fig. 3.5**)
5	First buckling and/or rupture of longitudinal bar(s), crushing of core concrete	Detect the first buckling and/or rupture of longitudinal bar(s), and/or detect at least two longitudinal bars and three transverse bars (**Fig. 3.6**)
6	Total collapse in which the permanent drift ratio exceeds 20%	The angular change of the line connecting the column ends with respect to the column initial position exceeds 10° ($\|\alpha\| > 10°$)

Note:

α = The angle between the column axial directions before and after the deformation (see the figure 3.1 below)

θ = The angle between the crack and the column axial direction (see the Figure 3.1 below)

D_{col} = The undamaged column diameter or the largest side dimension

= Inspected = Limited Use = Unsafe

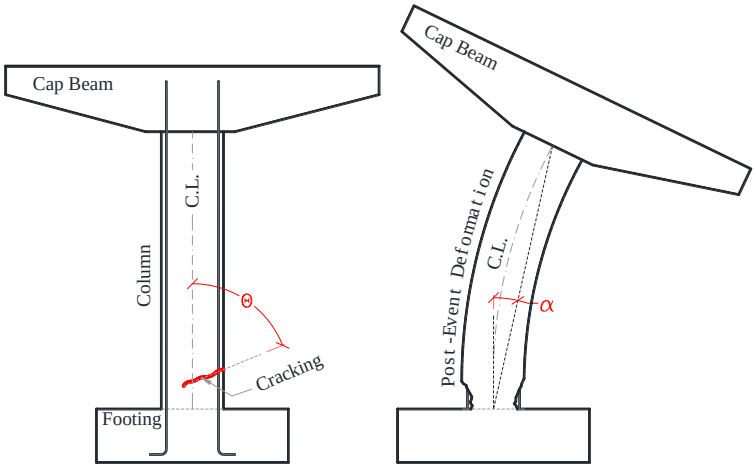


Figure 3. 1 Column axial direction

From figure 3.2 to 3.6, each damage state and target deficiency component has been described. Figure 3.2 indicates damage state 1 (DS-1) by showing horizontal cracks which are greater than 80 degrees. Also, figure 3.3 has horizontal cracks, but diagonal cracks have been found closed to horizontal cracks and counted more than 3. Figure 3.4 shows damage state 3 (DS-3) by indicating spalled regions which is less than 30% of column width. Damage state 4 (DS-4) are described in figure 3.5 with larger spalled regions compared to DS-3. Damage state (DS-5) shows exposed bars (longitudinal, transverse) in figure 3.6.

Figure 3. 2 Samples of RC Bridge Column Condition at DS-1 [45], [46]

Figure 3. 3 Samples of RC Bridge Column Condition at DS-2 [45], [46]

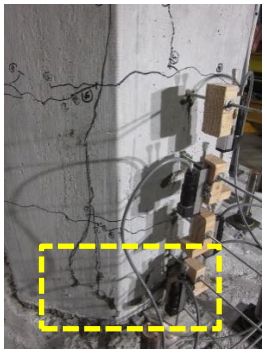


Figure 3. 4 Samples of RC Bridge Column Condition at DS-3 [45], [46]

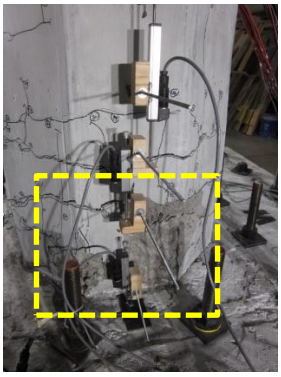


Figure 3. 5 Samples of RC Bridge Column Condition at DS-4 [45], [46]

Figure 3. 6 Samples of RC Bridge Column Condition at DS-5 [45], [46]

Before going over next sections, Figure 3.7 describe the overview of cascaded damage detection network. From stage 1 to stage 4, Mask R-CNN is a main network for instance segmentation for detecting target deficiency, column, longitudinal bars, transvers bars and spalled area. For last stage 5, MobileNet v2 is a main network to classify and segment crack from image. In figure 3.7, crack detection uses different network

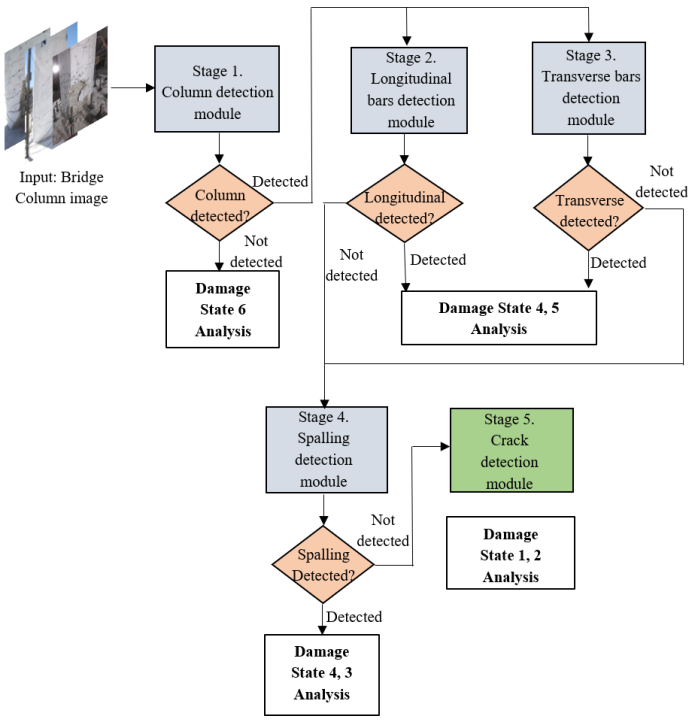


Figure 3. 7 Overview of proposed cascaded damage detection network

compared to other deficiency target detection network. In this study, crack deficiency is evaluated with number of horizontal and vertical cracks and their angles. But another deficiency module which is based on Mask R-CNN, focuses on detecting each component and dimension of column, spalled area, which is not necessary in crack evaluation. So, proposed network shows two different modules for crack and other deficiencies to determine accurate damage state.

In this section, detailed study about proposed model following the steps in figure 3.6 will be covered. First, we explain how the data for training and evaluating the proposed model is prepared in section 3.2, and second, two main modules and their processing are described in section 3.3.

3.2 Dataset

In the general computer vision tasks, a large amount data and high-resolution images are required to drive state-of-art result. An insufficient number of data and low-quality images can lead poor performance and give hard time to train network model. In this study, approximately 216 images were used for training and evaluation purpose of proposed network and divided into 80:20 ratios. Total training image number is 170 and the number of evaluation image is 46. Each image has size is different, but the image resolution is at least 2000×1980 and contains each deficiency component (column, spalling, rebar, and crack). In addition, for generating extra data, data augmentation technique has been adapted to increase network training. Each image was augmented with left/right-side flip and Gaussian blur operation of standard deviation value 0.5. Figure 3.8 shows the samples of image in training dataset and how we augmented those images to increase the number of images.

(a) (b) (c)

Figure 3. 8 Samples of data augmentation

(a) original image, (b) right-side flip, (c) Gaussian blur with standard deviation $\sigma = 0.5$

3.2.1 Crack data

For MobileNet v2 [21] classification module, each crack patch was generated by cropping dataset image with size of 64×64. For training classification model, both non-crack (figure 3.9 a, b) and crack patches (figure 3.9 c, d) were generated. As stated above, entire dataset images were counted as 216 and cropped crack patches from those images were counted as total 20,458 of 11,458 crack patch and 9,000 non-crack patches.

(a) (b) (c) (d)

Figure 3. 9 Sample image of crack dataset (a,b are non-crack and c, d are crack)

3.2.2 Bridge column, spalling and exposed rebar data

To instance segmentation module, it is important to annotate each instance and classify before feeding to Mask R-CNN module. The outline of each target deficiency was labeled by polygon shape and the coordinate information of polygon was saved in annotation file. And each instance was annotated into 4 classes (column, spalling, transverse bar, and longitudinal bar).

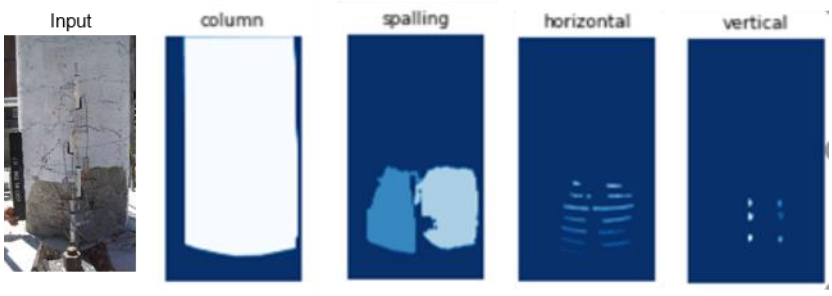


Figure 3. 10 Sample image of annotation

3.3 Bridge column, spalling and exposed rebar detection

As described in proposed network overview of figure 3.7, Mask R-CNN was used to detect deficiency components except crack. As reviewed Mask R-CNN [30] in chapter 2.3 and showed how this module works and derive a high performance of instance segmentation tasks. In this study, Mask R-CNN has been adapted to this proposed network to segment each target deficiency. Figure 3.11 is a detailed procedure of stage1 to stage4 in Figure 3.7.

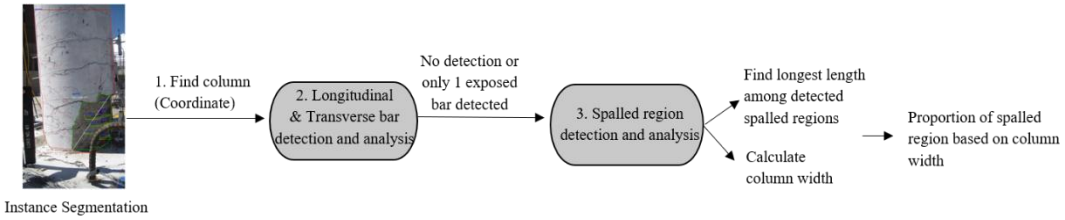


Figure 3. 11 Target deficiency object analysis

Once the image is fed to the Mask R-CNN module, each targeted deficiency object in an image is segmented and masked. From this result, proposed network will follow those analysis steps to determine damage state. The results from this module contain each target deficiency's location, class, and mask. First, column instances are analyzed. Column instances are important since all damage states are evaluated by measuring and quantifying deficiencies within column area. From column’s location (coordinate), it contains left-top x, y and right-bottom x, y and the area needed to inspect is narrowed by the box with that coordinate. Second, transverse and longitudinal bar analysis are followed by column detection phase. Proposed model counts the number of each detected bars and determine if counted number is matching with proposed damage

state DS-5 or DS-4. If the model cannot find any components which match DS-5 and DS-4, spalled region analysis step begins. In this step, the longest line inside of spalled region is used to determine damage state DS-4 or DS-3. For measuring longest line, mask information of spalled instance spalled region and measure the longest line inside of mask. For DS-3 and DS-4, column width is required to analyze those stages. Since there is a column mask information from the results of Mask R-CNN module, RANSAC [47] algorithm is used to calculate those two-vertical line of column. RANSAC algorithm is useful to fitting a line in two dimensions to a set of observations. Form masked column information, extract left-most-side coordinates and right-most-side coordinates. And RANSAC fit a line of each side and we can get a line equation for left and right side of column mask. And column width can be calculated by putting first y value of spalled region bounding box $[y_1\ x_1\ y_2\ x_2]$. And with this calculated width, the proportion between column width and longest distance is derived by simply dividing two values.

3.4 Crack detection

From the above procedure, if instance segmentation module cannot find those deficiency components, the network temp to find the last component crack. It is hard to detect crack in the image compared to other components which has relatively bigger instance area and more meaningful features and we only need the number of cracks and their angle value. So, we approach crack detection as a classification problem by cropping masked column area with 64×64 size patch.

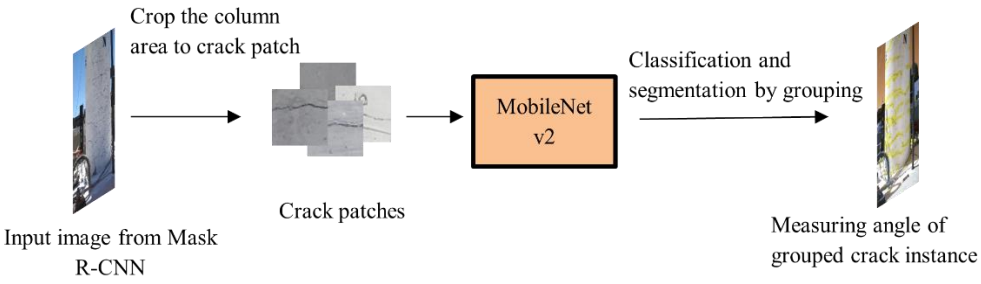
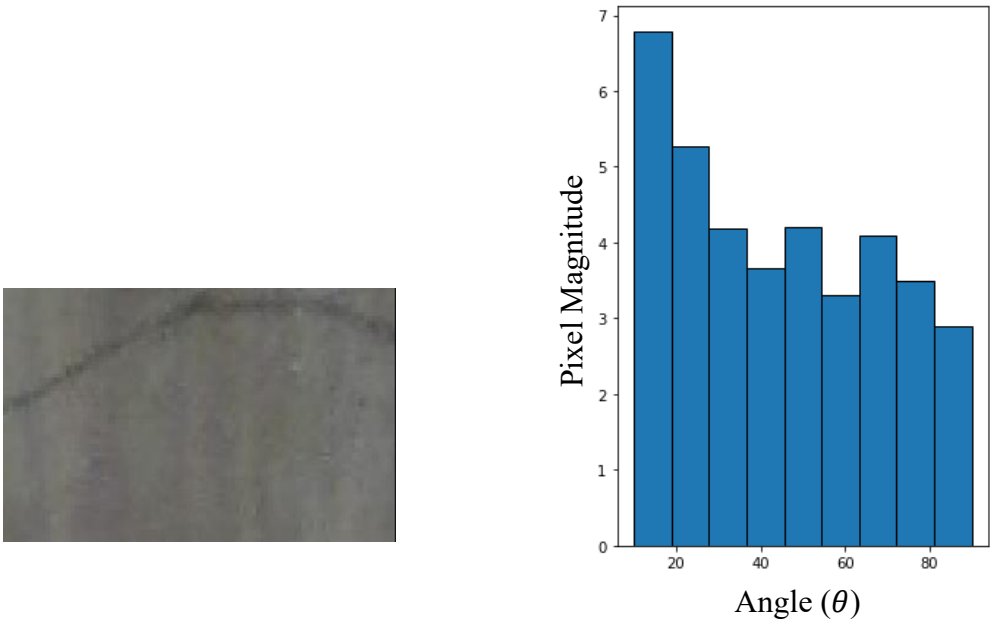


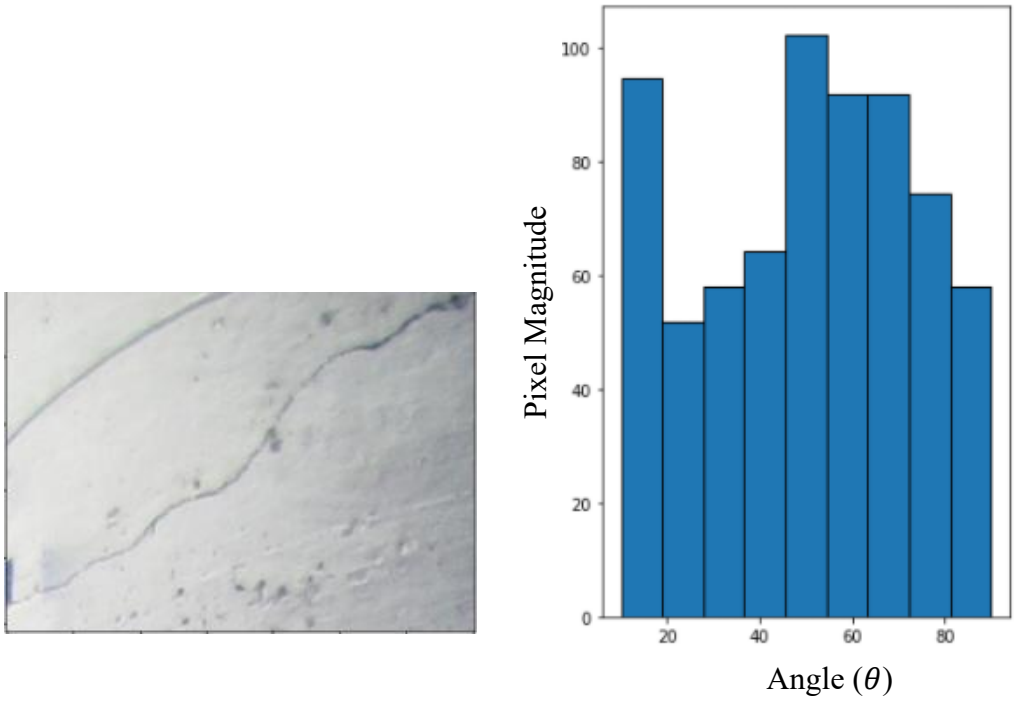
Figure 3. 12 Crack detection analysis

As MobileNet v2 state-of-art performance is described from section 2.2, MobileNet v2 has a light-weight computation compared to other existing models in same task but keep the high classification accuracy. MobileNet v2 will take cropped crack patches and determine either crack or non-crack. Once patch is classified as crack, the coordinate of patch is saved to segment patches which are connected in same crack.



(a) (b)

Figure 3. 13 (a) Segmented crack patches, (b) Measuring angle result



(a) (b)

Figure 3. 14 (a) Segmented crack patches, (b) Measuring angle result

Segmentation step decides that patches are corresponded to one crack if those two patches are close than patch size which is 64. In figure 3.13, segmented crack patch image is used to measure angle by histogram of oriented gradients (HOG) [9] algorithm. This algorithm computes a histogram of oriented gradients in each cell so we can find out which direction the cell's magnitude has been changed. And this step is followed by each segmented crack patches and pick the highest angle value to determine which direction crack is pointed out.

CHAPTER 4. EXPERIMENTAL AND RESULTS

From previous chapter, proposed cascaded deep learning model was explained detailed. In this chapter, training and evaluation of proposed model will be described and experimental and results of this study will be shown following by the steps in overview in Figure 3.7.

4.1 Damage Detection Results

For training Mask R-CNN module, each image was labeled with 4 classes (column, spalling, transverse, and longitudinal bar) and 1 background class. And the weight was initialized by pre-trained weight of COCO dataset [48] with batch size of 2 to fine-tune the network with our own dataset. And the model was trained with NVIDIA GeForce GTX 1080 Ti and process 2 images per GPU. Backbone architecture is ResNet101, and train epoch is 70 with learning rate of 0.02. During the training, Mask R-CNN weight had been updated by multi-task loss function below. $\mathcal{L}_{cls}$ is the log loss function over two classes (classification loss) and $\mathcal{L}_{box}$ is difference between localization of ground truth and output result. And $\mathcal{L}_{mask}$ is defined as the average binary cross-entropy loss, only considering associated with the ground truth classes.

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{box} + \mathcal{L}_{masks} \tag{3}$$

MobilNet v2 for crack detection has been trained with our own dataset and this module takes input size of 224×224. As same as Mask R-CNN, MobileNet v2 resized the crack patches for feeding module and same GPU has been used with batch size of 96. Also, batch normalization was used between each layer and activation function is ReLU6. The loss function of this network is categorical cross entropy loss with learning rate of 0.0001.

In testing, three shapes of RC column (circular, octagonal and rectangular) were tested. Currently, rectangular column dataset does not have images for DS-1 to DS-3 so only DS-4 and DS-5 results are provided in this document. And octagonal column does not have image for testing DS-5. Figure 4.1 – Figure 4.11 shows detected component results from our proposed model and Table 4.1 – Table 4.11 recorded our analysis results from the damage definition table in Table 2.1. From DS-3 to DS-5, each detected component has been marked with specific colors (red=column, green=spalled region, blue=transverse bar, purple=longitudinal bar).

Table 4. 1 Analysis of figure 4.1

Analysis Component	Result
Number of Horizontal Crack	11
Number of Vertical Crack	2
Maximum length of spalled region (px)	N/A
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**1**

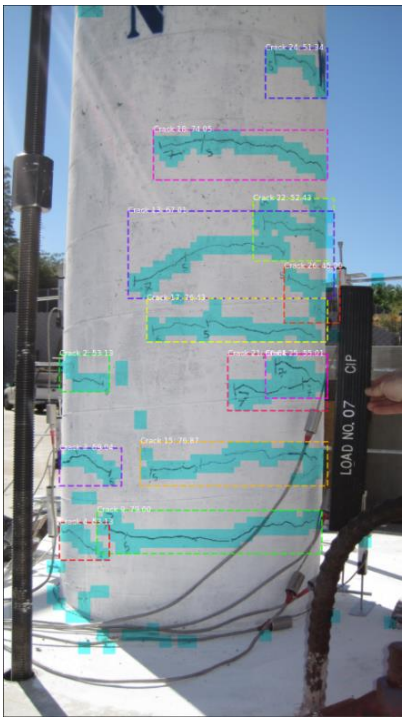


Figure 4. 1 Result sample of DS-1 (1)

Table 4. 2 Analysis of figure 4.2

Analysis Component	Result
Number of Horizontal Crack	5
Number of Vertical Crack	2
Maximum length of spalled region (px)	N/A
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	1

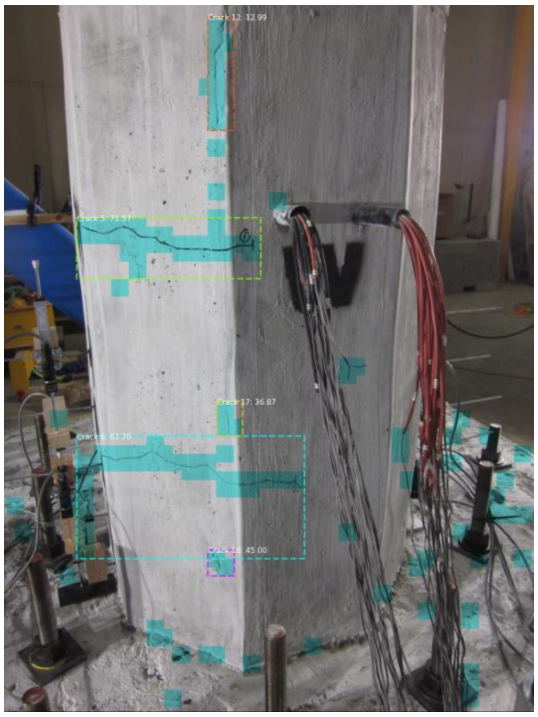


Figure 4. 2 Result sample of DS-1 (2)

Table 4. 3 Analysis of figure 4.3

Analysis Component	Result
Number of Horizontal Crack	9
Number of Vertical Crack	4
Maximum length of spalled region (px)	N/A
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**2**

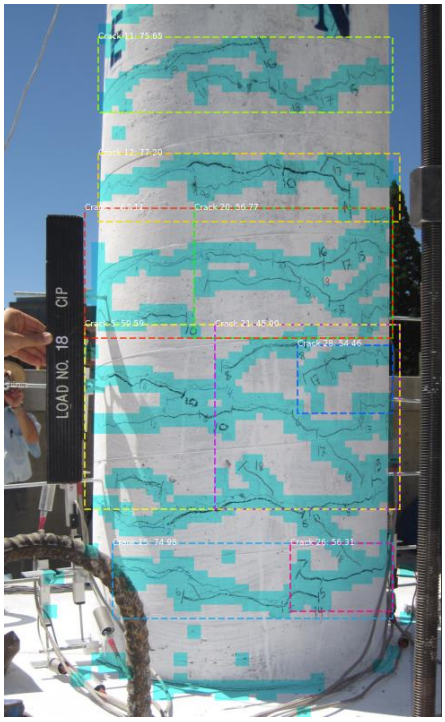


Figure 4. 3 Result sample of DS-2 (1)

Table 4. 4 Analysis of figure 4.4

Analysis Component	Result
Number of Horizontal Crack	2
Number of Vertical Crack	5
Maximum length of spalled region (px)	N/A
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**2**

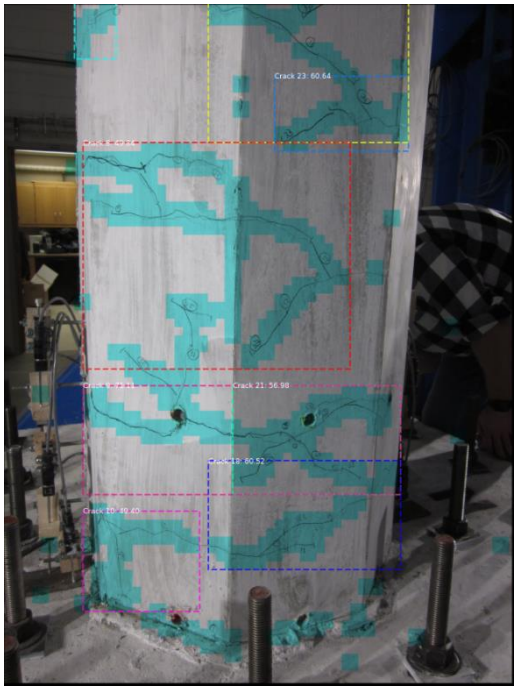


Figure 4. 4 Result sample of DS-2 (2)

Table 4. 5 Analysis of figure 4.5

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	537.23
Column Width (px)	1610.2
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	3

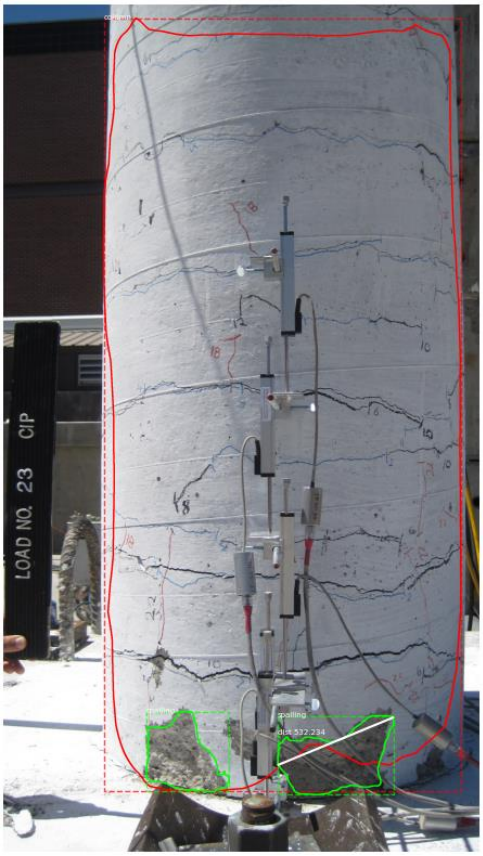


Figure 4. 5 Result sample of DS-3 (1)

Table 4. 6 Analysis of figure 4.6

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	491.62
Column Width (px)	1610.23
Number of Transverse (Horizontal) bar	N/A
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**3**

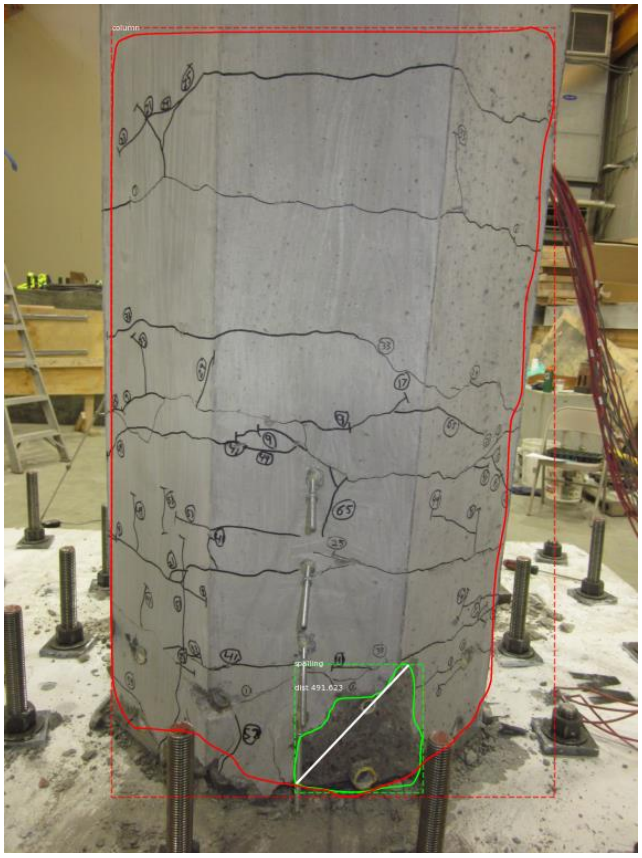


Figure 4. 6 Result sample of DS-3 (2)

Table 4. 7 Analysis of figure 4.7

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	1748.14
Column Width (px)	1809.67
Number of Transverse (Horizontal) bar	6
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**4**

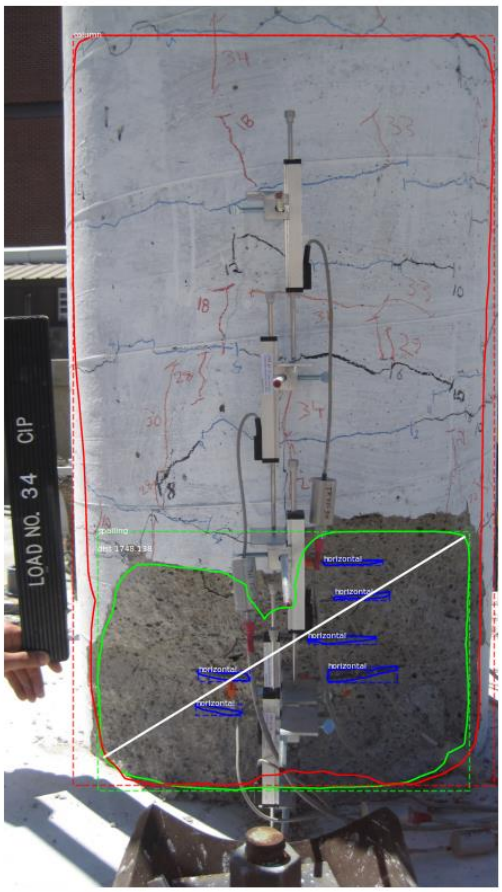


Figure 4. 7 Result sample of DS-4 (1)

Table 4. 8 Analysis of figure 4.8

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	870.13
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	8
Number of Longitudinal (Vertical) bar	N/A
Damage state (DS)	**4**

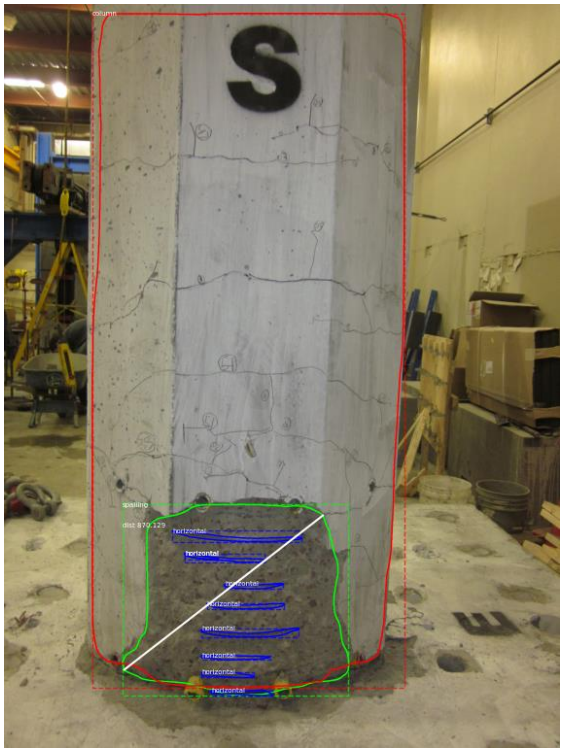


Figure 4. 8 Result sample of DS-4 (2)

Table 4. 9 Analysis of figure 4.9

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	1020.81
Column Width (px)	856.64
Number of Transverse (Horizontal) bar	3
Number of Longitudinal (Vertical) bar	1
Damage state (DS)	**4**

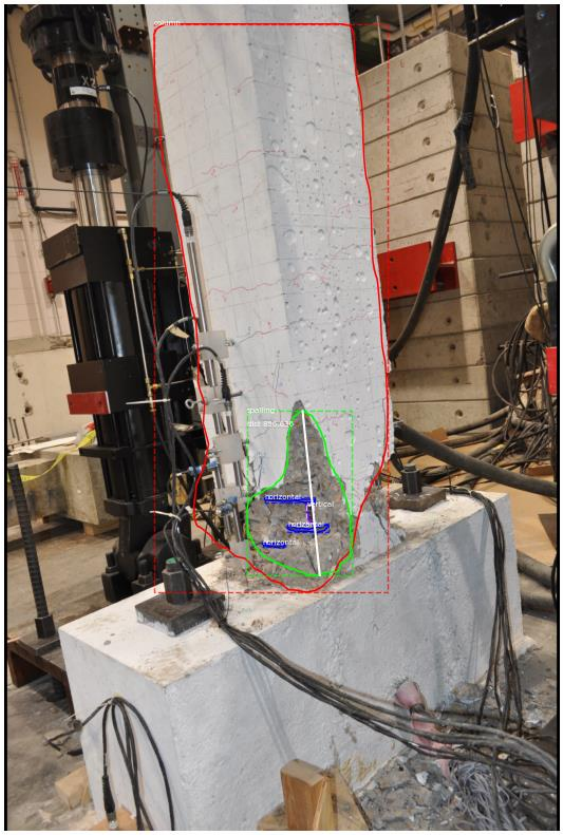


Figure 4. 9 Result sample of DS-4 (3)

Table 4. 10 Analysis of figure 4.10

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	1741.91
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	6
Number of Longitudinal (Vertical) bar	2
Damage state (DS)	**5**

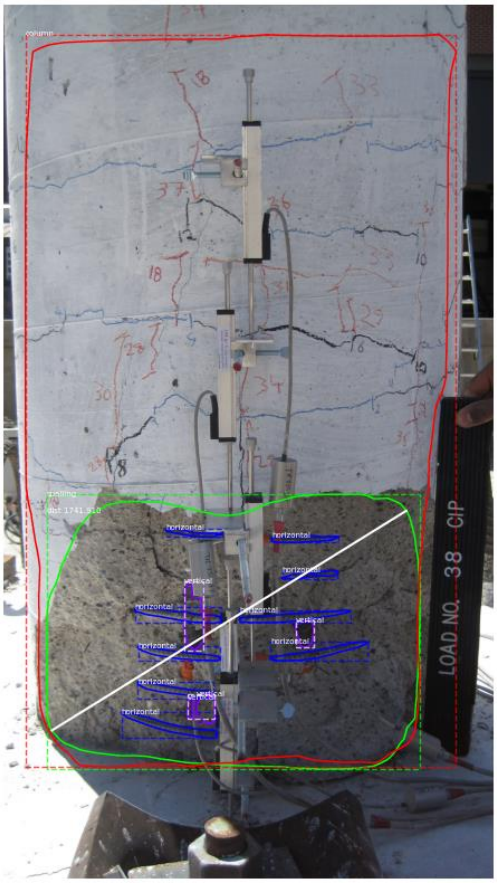


Figure 4. 10 Result sample of DS-5 (1)

Table 4. 11 Analysis of figure 4.11

Analysis Component	Result
Number of Horizontal Crack	N/A
Number of Vertical Crack	N/A
Maximum length of spalled region (px)	1317.08
Column Width (px)	N/A
Number of Transverse (Horizontal) bar	3
Number of Longitudinal (Vertical) bar	2
Damage state (DS)	**5**

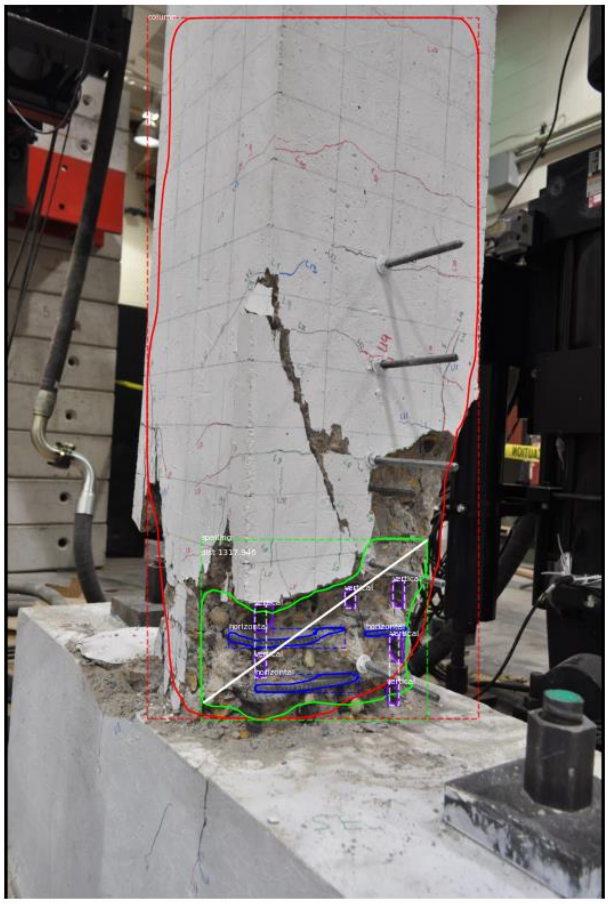


Figure 4. 11 Result sample of DS-5 (2)

4.2 Evaluation of Performance

The Mask R-CNN performance has been evaluated with precision and recall. Precision and recall (PR) is calculated by true positives (TPs), false positives (FPs), and false negatives (FNs). Precision is a ratio of true positive and overall positive results (true and false), and recall is a ratio of true positive and sum of true positive and false negative. Each number of true/negative is determined from overlapping between resulted masks and ground truth (our 42 validation images). Figure 4.12 shows the concept of Intersection over Union (IoU) and red box is ground truth and blue box is predicted output of each object. Each object is classified as TP over 0.5 and less than 0.5 as FP. When a network failed to detect the object stated in ground truth image it is classified as false negative (FN).

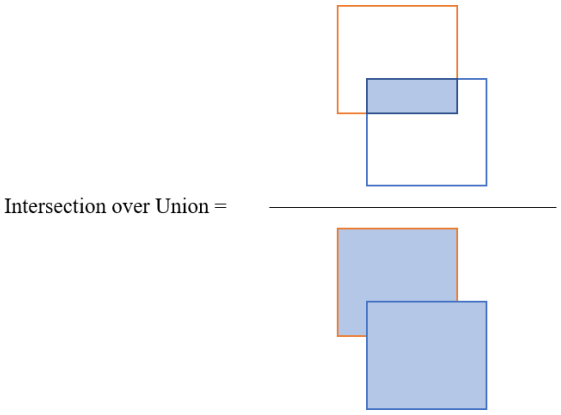


Figure 4. 12 Intersection of Union (IoU)

$$Precision = \frac{\#TPs}{\#TPs + \#FPs} \tag{4}$$

$$Recall = \frac{\#TPs}{\#TPs + \#FNs} \tag{5}$$

Instance segmentation module is analyzed of two parts of performance: localization and segmentation. Table 4.12 shows precision/recall results of average of each category (column, spalling and exposed bars). In testing images (42 validation images), there are 41 columns, 72 spalled areas, 56 transverse bar and 31 longitudinal bars. The PR table values were measured by overlapping pixels in each category. In this table below, the values of each component resulted above 90 percent for localization performance and have similar results for segmentation task but less than localization. The network performance for detecting target deficiency has above 90 percent overall but, column segmentation results are lower than any other components.

Table 4. 12 Evaluation results of target deficiency detection for each component

Component	Localization Precision (%)	Localization Recall (%)	Segmentation Precision (%)	Segmentation Recall (%)
Column	90.13	90.91	88.90	89.23
Spalled area	95.28	95.88	93.97	88.71
Transverse bar	95.27	95.82	92.71	93.14
Longitudinal	92.31	92.79	91.83	92.17
Average	93.24	93.85	91.10	90.81

Also, crack classification network is evaluated with same equation (4), (5). Total 4210 images (crack + non-crack) have been tested to get PR table and it has been reported with confusion matrix. Table 4.12 shows precision/recall results. PR score was measured for each classification (crack, non-crack). Total image is counted as 4,842 with 2,320 as crack and 2,522 as non-crack. And the network has slightly higher results in

detecting non-crack image. But overall results show this network high performance for classifying crack image with 97 percent.

Table 4. 13 Evaluation results of crack classification network

Component	Precision (%)	Recall (%)
Crack	96.32	95.28
Non-Crack	98.45	96.94
Average	97.38	96.11

Figure 4. 13 Real scene test result

As discussed, our overview of proposed network in figure 3.7, it is important to detect each component through the process. But if the input image is too close or too far to detect the column, the network cannot go over to detect rest of components. Figure 4.13 showed the test results from real scene. Proposed network can detect the object but the boundary is not accurate for column due to noise background like leaves around the spalled region. Also, Figure 4.14 shows few examples of failed detection. Those sample

images below are hard to detect entire column instance which is our first step in proposed network. So, it is important to make sure the entire image has bottom or top part of column border to see shape of column. Since neural network performance has dependent of dataset which has been trained only certain case. For example, the proposed network has been trained with our own dataset, so it is hard to derive state-of-art result if variety of dataset is not fed in model. So, data augmentation in figure 3.8 is important to derive high detection performance from the model.

Figure 4. 14 Samples of failed detection case

CHAPTER 5. SUMMARY AND CONCLUSIONS

This study presented cascaded deep learning network for post-earthquake bridge serviceability with our own analysis method. Mask R-CNN is composed of three stages (region proposal, classification, and segmentation) and MobileNetV2 has a light-weight residual block with depth-wise separable convolution. For instance segmentation of column, spalling and rebar, Mask R-CNN is trained with backbone network ResNet101, and MobileNetV2 is trained for crack detection and segmentation. Both networks have been trained with 216 images of 100 epoch each and 43 images for evaluation purpose. And analysis for each damage state has followed by postprocessing like measuring column width, counting diagonal/horizontal cracks and exposed transverse/longitudinal bars.

In future studies, the class or post-processing can be extended to buckling of bars, total collapse of structures which has been described in damage state 6. Also, crack patch generation can be modified to fit crack and column width ratio for pre-processing of data for crack detection phase so advanced classification results can be expected. In addition, Mask R-CNN has more computational cost than MobileNet v2, the backbone of Mask R-CNN can be replaced with cost-efficient network to reduce heavy computation during training and testing.

LITERATURE CITED

[1] Jeong, S., Hou, R., Lynch, J. P., Sohn, H., & Law, K. H. (2017). An Information Modeling Framework for Bridge Monitoring. *Advances in Engineering Software*. 114,11-31. doi: https://doi.org/10.1016/j.advengsoft.2017.05.009

[2] Zhou, G., Li, A., Li, J. & Duan, M. Structural Health Monitoring and Time-Dependent Effects Analysis of Self-Anchored Suspension Bridge with Extra-Wide Concrete Girder. (2018). *Appl. Sci*.. 8, 115, doi:10.3390/app8010115.

[3] Kawashima, K., Kosa, K., Takahashi, Y., Akiyama, M., Nishioka, T., Watan-abe, G., Koga, H., Matsuzaki, H. (2011). Damage of Bridges during 2011 Great East Japan Earthquake.

[4] Li, Y.-F., & Sung, Y.-Y. (2011, 02). Seismic Repair and Rehabilitation of a Shear-failure Damaged Circular Bridge Column using Carbon Fiber Reinforced Plastic Jacketing. *Canadian Journal of Civil Engineering*. 30, 819-829. doi:10.1139/l03-042

[5] Jahanshahi R, M. & Masri F, S. (2013). A new Methodology for Non-contact Accurate Crack Width Measurement through Photogrammetry for Automated Structural Safety Evaluation. *Smart Mater. Struct.* 22(3).

[6] Kim, J., Kim, A. & Lee, S. (2020). Artificial Neural Network-based Automated Crack Detection and Analysis for the Inspection of Concrete Structures. *Applied Sciences*, 10(22):8105. https://doi.org/10.3390/app10228105

[7] Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics.* 9(1):62-66. doi: 10.1109/TSNC,1979.4310076

[8] Ge, S., Yang, R., Wen, H., Chen, S., & Sun, L. (2014). Eye Localization based on Correlation Filter Bank. *In 2014 IEEE International Conference on Multimedia and Expo (ICME)*. p.1-5. doi: 10.1109/ICME.2014.6890249

[9] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1:886-893. doi: 10.1109/CVPR.2005.177.

[10] Paal, S., Jeon, J., Brilakis, I. and Desroches, R. (2015). Automated Damage Index Estimation of Reinforced Concrete Columns for Post-Earthquake Evaluations. *Journal of Structural Engineering*. 141. doi:04014228. 10.1061/(ASCE)ST.1943-541X.0001200.

[11] Nishikawa, T., Yoshida, J., Sugiyama, T. & Fujino, Y. (2012). Concrete Crack Detection by Multiple Sequential Image Filtering. *Comp.-Aided Civil and Infrastruct. Engineering*. 27, 29-47. 10.1111/j.1467-8667.2011.00716.x.

[12] Yeum, C., & Dyke, S. (2015). Vision-Based Automated Crack Detection for Bridge Inspection. *Comput. Aided Civ. Infrastructure Eng., 30*, 759-770.

[13] Kim, H., Ahn, E., Shin, M., & Sim, S. (2019). Crack and Noncrack Classification from Concrete Surface Images Using Machine Learning. *Structural Health Monitoring, 18,* 725 – 738.

[14] Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60,* 84 - 90.

[15] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9.

[16] Yeum, C., Dyke, S. & Ramirez, J. (2018). Visual data classification in post-event building reconnaissance. *Engineering Structures*. 155. 16-24. 10.1016/j.engstruct.2017.10.057.

[17] 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 Civ. Infrastruct. Eng*. 2018, 33, 731–747

[18] He, K., Gkioxari, G., Dollár, P. & Girshick, R. (2017). Mask R-CNN.

[19] Yein, L., Kim, B., & Cho, S. (2018). Image-based Spalling Detection of Concrete Structures Using Deep Learning. *Journal of The Korea Concrete Institute*, *30*, 91-99.

[20] Guillamón, J.R. (2018). Bridge Structural Damage Segmentation Using Fully Convolutional Networks.

[21] Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., & Chen, L. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510-4520.

[22] Mcculloch, W. S., & Pitts, W. H. (1942). A logical Calculus of Ideas Immanent in Nervous Activity. *The Bulletin of Mathematical Biophysics*. 5, 115-133.

[23] Rosenblatt, F. (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*. 368-408.

[24] Rumelhart, D., Hinton, G.E., & Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature, 323,* 533-536.

[25] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation, 1*, 541-551.

[26] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition.

[27] Li, Z., Yang, W., Peng, S., & Liu, F. (2021). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE transactions on neural networks and learning systems, PP*.

[28] He, Z. (2020). Deep Learning in Image Classification: A Survey Report. *2020 2nd International Conference on Information Technology and Computer Application (ITCA).* 174-177, doi: 10.1109/ITCA52113.2020.00043.

[29] Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR, abs/1409.1556*.

[30] Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ArXiv, abs/1502.03167*.

[31] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2818-2826.

[32] Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A.A. (2017). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *AAAI*.

[33] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.

[34] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv, abs/1704.04861*.

[35] Shelhamer, E., Long, J., & Darrell, T. (2017). Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*, 640-651.

[36] Ren, S., He, K., Girshick, R.B., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*, 1137-1149.

[37] Kawashima, K. (2000). Seismic Performance of RC Bridge Piers in Japan-An Evaluation after the 1995 Hyogo-ken Nanbu Earthquake. *Progress in Structural Engineering Materials*. 82-91.

[38] Kawashima, K. (2001). Damage of Bridges Resulting from Fault Rupture in the 1999 Kocaeli and Duzce, Turkey Earthquakes and the 1999 Chi-Chi, Taiwan Earthquake. *Seismic Fault-Induced Failures*, 171-190.

[39] Berry, M., Parrish, M., & Eberhard, M. (2004). PEER Structural Performance Database. *Pacific Earthquake Engineering Research Center, University of California, Berkeley*, 43

[40] Veletzos, M., Panagiutou, M., Restrepo, J., & Sahs, S. (2008). Visual Inspection and Capacity Assessment of Earthquake Damaged Reinforced Concrete Bridge Elements. California Department of Transportation Report No. CA08-0284, Sacramento, CA, 392

[41] Vosooghi, A., & Saiidi, M.S. (2010). Seismic Damage States and Response Parameters for Bridge Columns. *ACI Special Publication*, 271(2), 29-46.

[42] EERI Archive. (2020). Learning from Earthquakes Reconnaissance Archive. Retrieved on February 11, 2020 from < https://www.eeri.org/projects/learning from-earthquakes-lfe/lfe-reconnaissance-archive/>.

[43] Ghannoum, W., Sivaramakrishnan, B., Pujol, S., Catlin, A.C., Fernando, S., Yoosuf, N., & Wang, Y. (2015). NEES: ACI 369 Rectangular Column Database. https://datacenterhub.org/resources/255.

[44] Hose, Y.D. (2001). Seismic Performance and Flexural Behavior of Plastic Hinge Regions in Flexural Bridge Columns. PhD Dissertation, University of California, San Diego, 493.

[45] Haber, Z.B., Saiidi, M.S., & Sanders, D.H. (2013). Precast Column-Footing Connections for Accelerated Bridge Construction in Seismic Zones. Center for Civil Engineering Earthquake Research, Department of Civil and Environmental Engineering, University of Nevada, Reno, Nevada, 13-08, 612.

[46] Sjurseth, T. (2021). Mechanically Spliced Precast Bridge Columns. South Dakota State University, 197. https://openparairie.sdstate.edu/etd/5252

[47] Fischler, M. & Bolles R. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image analysis and Automated Cartography. *Communications of the ACM*, 24(6):381-395. https://doi.org/10.1145/358669.358692.

[48] Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. & Zitnick, C. (2014). Microsoft COCO: Common Objects in Context.