AI ENHANCED SEISMIC ISOLATED BRIDGE DESIGN AND MAINTENANCE

A DISSERTATION

SUBMITTED TO THE COMMITTEE ON GRADUATE STUDIES IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF ENGINEERING

KATRINA MAE SANTIAGO MONTES

SUPERVISED BY

ASSOCIATE PROFESSOR JI DANG

DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
SAITAMA UNIVERSITY, JAPAN

SEPTEMBER 2023

PREFACE

This dissertation is submitted to the Department of Environmental Science and Civil Engineering and the Committee on Graduate Studies of Saitama University in partial fulfillment of the requirements for the degree of Doctor of Engineering. The work described in this dissertation has been conducted by the candidate during the period of 2020 to 2023 in the Structural Mechanics and Dynamics Laboratory in the Department of environmental science and civil engineering at Saitama University under the supervision of Associate Professor Dang Ji.

In accordance with the guidelines and policies of Saitama University governing the requirements of the degree of Doctor of Engineering, the candidate declares that the work presented in the dissertation is original unless otherwise referenced in the text.

Neither the dissertation nor any part thereof has been submitted or is being concurrently submitted in candidature for any degree at any other institution (except for own publications).

ACKNOWLEDGEMENT

First and foremost, the author would like to thank her supervisor, Associate Professor Ji Dang, for his unconditional support and very long patience over the past 3 years. He opens a lot of opportunities for her and guided her every step of the way. Through this journey, the researcher experienced different challenges, pressure, failures, etc. but thankfully her supervisor is there to guide /save her. The author has been inspired to explore research more by his continuous encouragement and guidance. Because of him, the author gained more knowledge, learned new skills like programming, improved her perspective on many things, and became a stronger person. The author will always keep all the learnings and continue to do research in the future.

The author would like to express her gratitude to her co-supervisors, Professor Okui, Professor Matsumoto, and Professor Saitoh, for their valuable suggestions on her research. To Kudo-san, the author sincerely appreciates her guidance and care. Also, the author expresses her appreciation to her research collaborators, Kawakin Core-Tech Co., Professor Igarashi, Dr. Yuqing Tan, and Associate Professor Chen. The author is greatly in debt by the kindness of Professor Nakamoto, for ensuring her safety during her stay at Saitama University. The author would also like to acknowledge Monbukagakusho (MEXT) Scholarship for providing with financial assistance.

To the author's close friends and laboratory group member who serves like a family during her stay at Saitama University, she would like to express her appreciation for their emotional support and help, especially Sal Saad, Christina, Aijia, Liu, Sania, Karma, Sanjeev, Arthur, and Samadi.

To Nida Maramara Sensei, the author would like to express her appreciation for teaching Nihongo and continuous emotional support which helps the author a lot in her everyday life in Japan.

The author would like to acknowledge her family, for their love, guidance, and help, especially her mother who unconditionally love her and encourage her to pursue her dreams from the start.

Lastly, the author would like to express her gratitude to Ryo Kawanoue, for the love, care, patience, understanding, and encouragement towards her which inspires her to do her best.

ABSTRACT

This study proposed various applications of artificial intelligence to enhance some processes in bridge seismic isolation design and vision-based bridge inspection and maintenance. For the bridge design phase, ANN-based enhanced nonlinear model selection, an optimization method for parameter identification, creating a nonlinear model library, bridge response evaluation, and design suggestions were proposed. The accuracy of the predicted nonlinear parameters from nonlinear models, equivalent stiffness, and equivalent damping, were then compared to the real HDR-S quasi-static loading and hybrid simulation experiment data. This will help the engineers to have an idea of what nonlinear model is suitable and will help with the initial parameter assumption which reduces the optimization process time and increase the chance of convergence.

For the bridge seismic isolated design performance evaluation and design suggestion, the trained ANN successfully classified the bearing's shear strain and determine whether it satisfy the design standard based on seven initial design input parameters. In addition to that, the ANN can predict the bearing's shear strain and suggest design parameters such as cross-section width and total rubber thickness of the HDR-S bearing. This reduces the trial-and-error process of initial design assumption and does not need to repeat the nonlinear simulation for performance evaluation.

Lastly, the proposed enhanced vison-based bridge damage detection and maintenance utilized the combination of different technologies such as UAV, deep learning method for corrosion segmentation, Mixed-Reality platform, and 3-D model reconstruction. The suggested UAV flight path in gathering the data of a steel bridge was proposed, then the 3-D model of the bridge with and without segmented corrosion was reconstructed using Structure-from-Motion (SfM). The localization of the segmented corrosion damage can be visualized from the whole bridge perspective. Then, the generated 3D model can be accessed through the mixed-reality platform for further evaluation which promotes digitalization. In continuation, the point cloud data were gathered for further 3D segmentation of bridge components and damage to better evaluate the cause and effect of the damages with the relationship of the component surface and the location and continuation of the damage.

TABLE OF CONTENTS

ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
BIM	Building Information Modelling
BL	Bilinear
BW	Bouc-Wen
CL	Clough
CNN	Convolutional Neural Networks
DL	Deep Learning
DT	Decision Tree
FPB	Friction Pendulum Bearing
HDR	High Damping Rubber
HDR-S	Super High Damping Rubber
JRA	Japan Railway Association
KNN	K-Nearest Neighbors
LDA	Linear Discriminant Analysis
LiDAR	Light Detection and Ranging
LR	Linear Regression
LRB	Lead Rubber Bearing
MAE	Mean Average Error
MBL	Modified Bilinear
MBW	Modified Bouc-Wen
MDOF	Multiple Degree of Freedom
ML	Machine Learning
MR	Mixed Reality
MSE	Mean Squared Error
NB	Naïve Bayes
ReLU	Rectified Linear Unit
RF	Random Forest
RMSProp	Root Mean Squared Propagation
SfM	Structure-from-Motion
SVM	Support Vector Machine
TR	Trilinear
UAV	Unmanned Aerial Vehicle
VGG-Unet	Visual Geometry Group Network Architecture
YOLO v3	You Only Look Once Version 3

LIST OF FIGURES

LIST OF FIGURES

LIST OF FIGURES

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 History of Bridge Seismic Isolation Design and Practice

The Great Kanto earthquake of 1923 was the motivation for the Japanese to develop their first own bridge seismic design to prevent the earthquake-related structural damage. The seismic design code was continuously improved [1][2][3] based on the observed damages occurred at 1978 Miyagi-ken earthquake. Design Specifications of Highway Bridges (JRA 1990) part V was the first seismic design code to include the design requirement for ductility capability to prevent shear failures for piers and columns. While the 1990 design code was being developed, numerous seismic isolation research investigations were ongoing. The Miyagawa Bridge in Shizuoka-ken and the Yama-age Bridge in Tochigi-ken were built as the first seismically isolated bridges using lead rubber bearings and high damping rubber bearings, respectively, in 1991 and 1992. Early in the 1990s, Lead Rubber Bearing (LRB), High Damping Rubber Bearing (HDR), and elastomeric bearings were also put into use. These bearings were used to distribute the inertial force to all substructures in a multi-span continuous bridges as well as seismically isolated bridges. Then, when the 1995 Kobe earthquake happened, severe damages to piers and bearings from foundations occurred, one example is the collapse of Fukake bridge as shown in Figure 1.

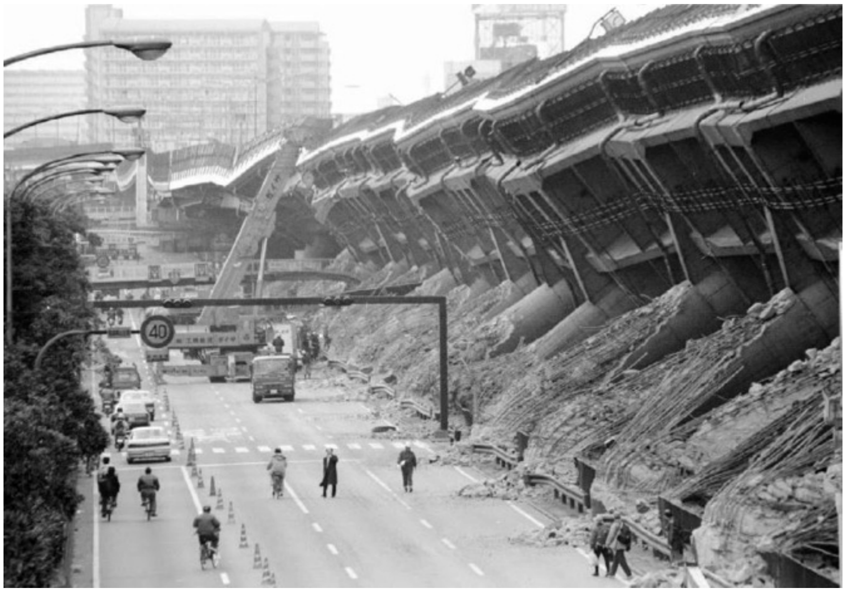


Fig. 1 Fukake Bridge Collapsed due to 1995 Kobe Earthquake

It was observed that some of the based isolated bridges with installed high damping bearing's had minimal damage. This was further noticed after the Great East Japan Earthquake of 2011, wherein the use of elastomeric bearings such as Lead rubber bearing (LRB) and High damping rubber bearings (HDR) significantly reduced the bridge damages compared to steel bearings as shown in Figure 2. Therefore, the application of elastomeric bearings starts to be a retrofit technique and installed to based isolated bridges [4][5]. Furthermore, the seismic design code for bridges has gone through significant improvements due to the incorporation of inelastic static analysis, increased ductility capacity of reinforced concrete and steel columns, design for residual displacement, seismic isolation, and the use of elastomeric bearings, such as lead rubber bearings and high damping rubber bearings, to replace the steel bearings [6].

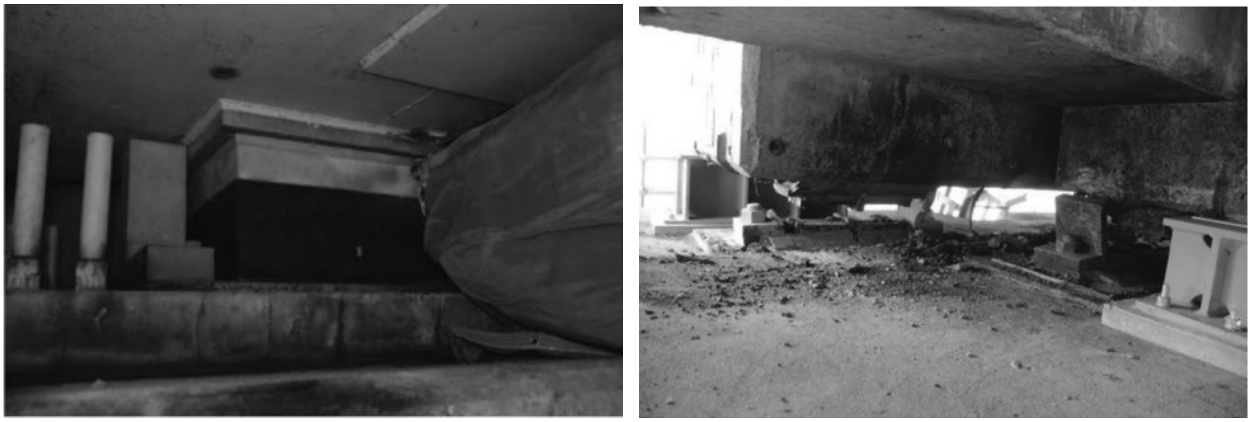
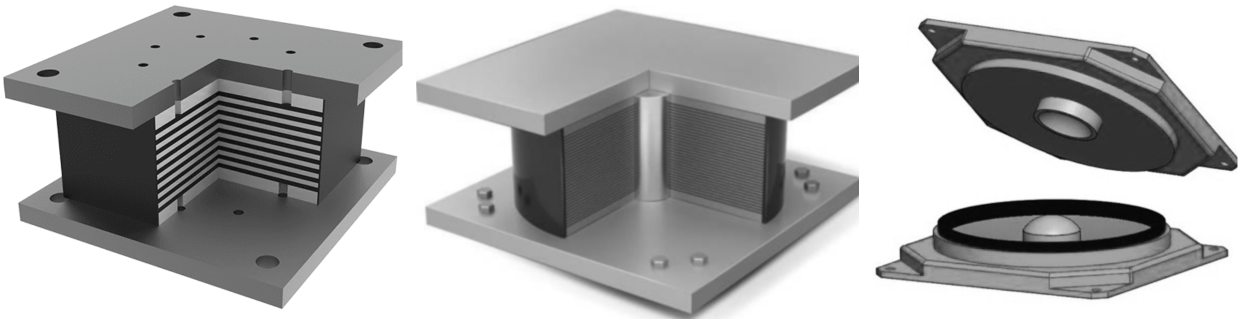


Fig. 2 (a) Undamaged Elastomeric Bearings and (b) Damaged Steel Pin Bearings

Seismic isolation bearings are usually installed between the superstructure and at the top of the substructure (piers and/or abutments). It increases the structure's flexibility in the event of a major earthquake by lengthening the structures period and increasing the energy dissipation. This makes it possible for the superstructure to vibrate at a lower frequency than its piers, which reduces or eliminates deformation of the substructure components out of their elastic range, especially to components which are difficult to inspect or repair [7]. It has been proposed that when applying seismic isolation to highway bridges and viaducts, the natural period should not be excessively increased but the emphasis should be placed on increasing the energy dissipation capacity and dispersing seismic lateral force of the deck to as many piers as possible. In addition to that, it has been recommended that the seismically isolated bridge's natural period should be nearly twice as long as that of a fixed-base bridge [8].

There are different types of isolation bearing devices such as friction pendulum bearing, high damping rubber bearing, and lead rubber bearing as shown in Figure 3. Lead rubber bearing (LRB) [9] primarily comprises of lead plug with laminated rubber layers reinforced by steel sheets. The rubber layers help to stabilize the structure's weight, offer lateral flexibility, and return the bearing to its initial location. Steel shims offers the vertical rigidity to sustain the weight of the superstructure and control the lead core deformation. The source of energy dissipation is a lead plug, which lessens the overall displacement. Friction pendulum bearing (FPB) [10] is a modification of sliding-type bearings that overcomes the lack of restorative capacity once distorted by adding a spherical sliding surface to provide restoring stiffness while the friction between the sliding interfaces helps in energy dissipation. High Damping Rubber bearing consists of alternating layers of synthetic rubber that are cured at high temperatures and pressures to attach to intermediate steel shim plates. The bearing additionally has a rubber cover to guard the steel plates and internal rubber layers against ozone attack and corrosion, respectively, and environmental deterioration [11].



(a) (b) (c)

Fig. 3 Different Seismic Isolation Bearing Devices (a) HDR (b) LRB and (c) FPB

1.2 Nonlinear Behavior and Models for High Damping Rubber Bearing

Although HDR-S have high performance as a seismic isolation device, it has complicated hysteretic behavior due to different factors. The behavior of HDR-S bearing was observed to have a significant increase in shear stress at low temperature compared to ambient temperature [12], Mullin’s effect was observed in every first loading at each amplitude [13], the cyclic loading response was rate dependent [14], and the deterioration of rubber should be taken account [15][16].

Due to these reasons, the nonlinear behavior of visco-elastic material such as HDR-S bearings requires a sophisticated nonlinear model with many parameters to accommodate these uncertainties. To ensure that the selected nonlinear model and parameters are appropriate, it needs to be peer evaluated. Some engineers are trying to use easy to use nonlinear models due to limited experience, however it might cause danger during large seismic events. The commonly used nonlinear models for elastomeric bearings behavior are Bilinear [17], Ramberg- Osgood [18], and Bouc-Wen model [19]. Other more complicated nonlinear models have been proposed by some researchers. Nguyen at al. proposed an updated Rheology hysteresis model that considers the rate-dependence of High Damping Rubber bearings [14]. Modified Park-Wen model includes stiffness degradation and pinching behavior of high-damping rubber bearings [20]. Consideration of the stiffening behavior of elastomeric bearings based on shear strain magnitude was suggested by the Kikuchi hysteresis model [21]. A nonlinear model with self-heating and temperature dependent was proposed [12]. It was further improved by combining the modified Bouc-Wen model with inner temperature dependent [22] of a laminated rubber bearing [23] and high damping rubber bearing. In continuation, Hybrid simulation [24] was conducted, in which the behavior of the seismic isolator was influenced by the structure's response to the input earthquake wave [25] and was used to further validate the nonlinear parameters finding. A nonlinear model was proposed with account of Mullin's effect, inner temperature, stiffness degradation, and pinching [26]. These nonlinear models with different nonlinear parameters can be use in bridge seismic isolation design instead of bilinear model.

Based on the Japan bridge seismic control design manual [27], the design flow and performance evaluation of seismic isolated bridges is shown in Figure 4. It starts from the assumption of the isolation devices specifications such as for HDR-S, the cross-section area, number of rubber layers, thickness of rubber, and the nonlinear model and parameters are needed. Then, the setting of design displacement is usually 250% based on the standard limitation. From that, the calculation of equivalent stiffness and damping constants are needed based on the design displacement and obtained nonlinear parameters using Bilinear Model and equivalent linearization method. However, Bilinear model does not include any temperature dependent parameters, stiffness degradation, Mullins effects, deterioration effect, etc., which was discussed as the problem on why HDR-S have complicated nonlinear behavior. Therefore, this study proposed a machine learning

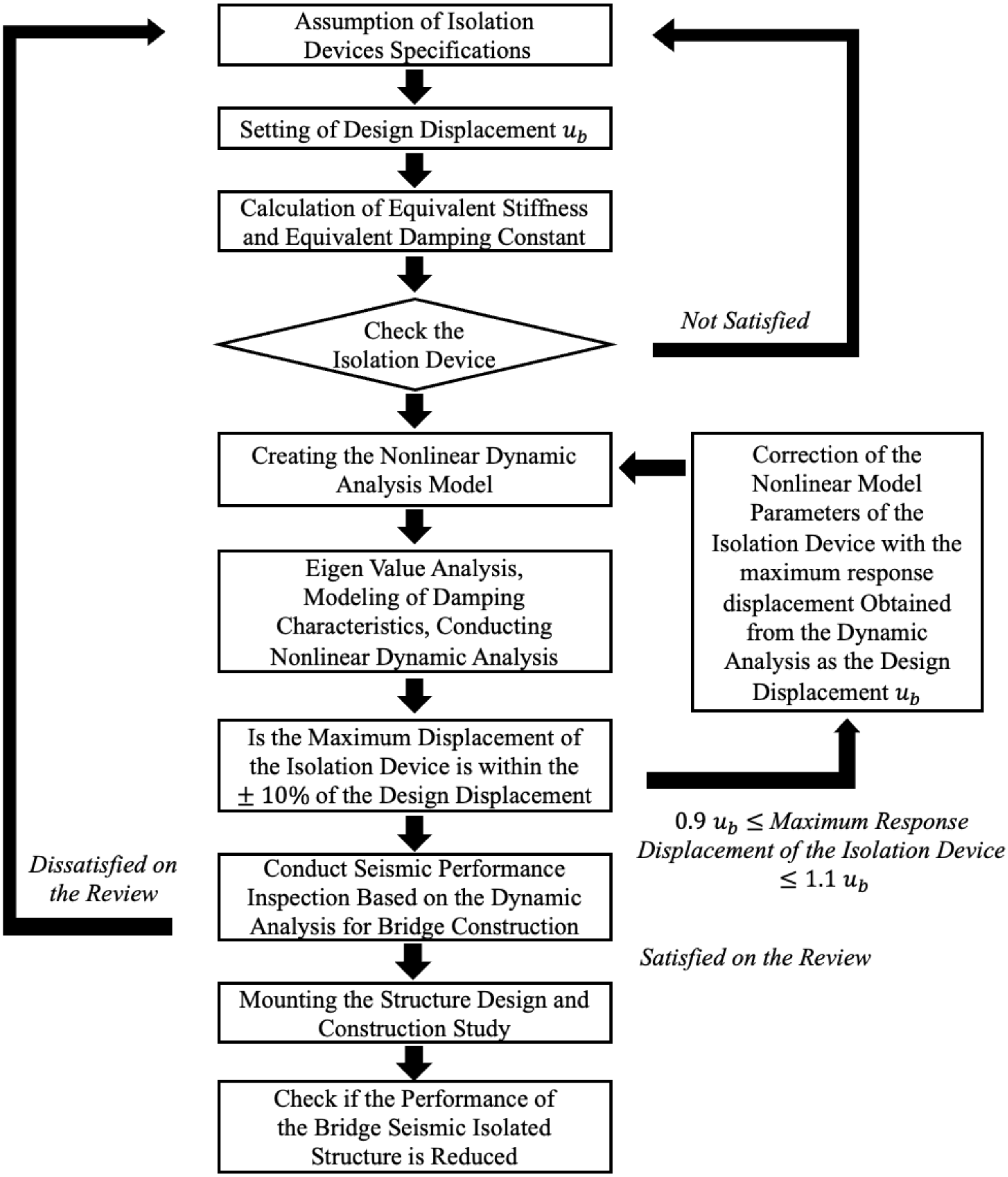

Assumption of Isolation Devices Specifications
Setting of Design Displacement u_b
Calculation of Equivalent Stiffness and Equivalent Damping Constant
Check the Isolation Device
Not Satisfied
Creating the Nonlinear Dynamic Analysis Model
Correction of the Nonlinear Model Parameters of the Isolation Device with the maximum response displacement Obtained from the Dynamic Analysis as the Design Displacement u_b
Eigen Value Analysis, Modeling of Damping Characteristics, Conducting Nonlinear Dynamic Analysis
Is the Maximum Displacement of the Isolation Device is within the ± 10% of the Design Displacement
$0.9\ u_b \leq$ Maximum Response Displacement of the Isolation Device $\leq 1.1\ u_b$
Dissatisfied on the Review
Conduct Seismic Performance Inspection Based on the Dynamic Analysis for Bridge Construction
Satisfied on the Review
Mounting the Structure Design and Construction Study
Check if the Performance of the Bridge Seismic Isolated Structure is Reduced

Fig. 4 Design Flow of Seismic Performance Evaluation of Bridges using Isolation Devices

based nonlinear model classification and parameter identification which will be discussed in the next chapters. Then after that, the creation of nonlinear model, eigen value analysis, and conducting nonlinear dynamic analysis are needed to check if the isolation's maximum displacement response agrees to the assigned design's displacement with a tolerance of $\pm 10\%$. If it does not satisfy the given condition, the design displacement will be changed to the isolation's maximum response and the process from the creation of nonlinear dynamic model will be repeated until the isolation's maximum response will satisfy the condition. This process is trial and error, therefore, this study also proposed a machine learning based seismic isolation performance evaluation and design suggestion. Lastly, there is a need to conduct performance inspection on the dynamic analysis result for bridge construction.

1.3 Machine Learning Application in Seismic Design

Machine learning demonstrated its effectiveness in learning the weights of input to output relationship utilizing a large amount of data. If the output value is discrete, the application task could be classification, or if the output is continuous, it could be regression [28]. Naive Bayes (NB) [29], Linear Discriminant Analysis (LDA) [30], Logistic Regression (LR) [31], K-Nearest Neighbors (KNN) [32], Support Vector Machine (SVM) [33], Decision Tree (DT) [34], Random Forest (RF) [35], Artificial Neural Network (ANN) [36], etc. are some examples of different machine learning methods. According to previous studies, time-series data can be classified using machine learning in a variety of applications [37], [38].

One type of machine learning is Artificial Neural Network (ANN) which was used in this study. The typical ANN model's architecture is depicted in Figure 5, the input layer's, bias, and weighted input variable sums are first transmitted to the hidden layer from the input layer. Based on activation functions, they are then assessed as shown in Figure 6. The variables from the hidden layer are then further weighted, and the activation functions are once more used to evaluate the hidden layer, bias, and weighted variables. They are then moved to the output layer to evaluate the predictive model.

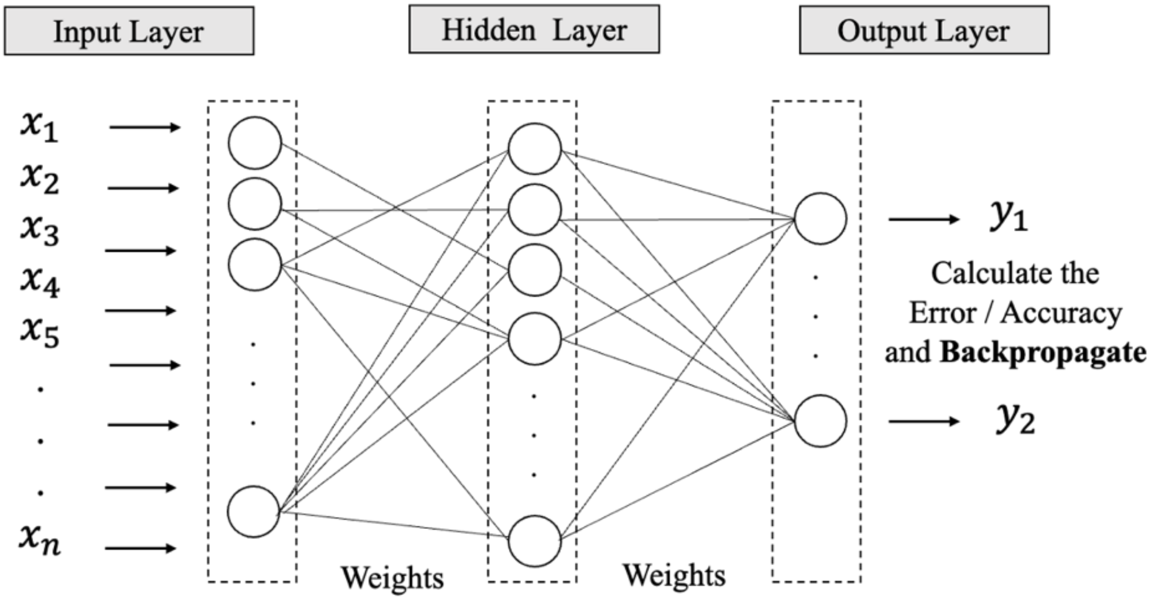


Fig. 5 Artificial Neural Network

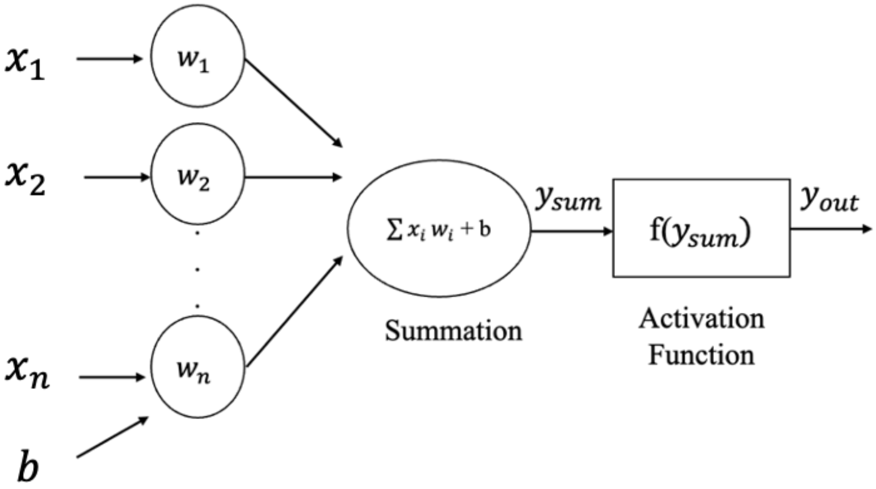


Fig. 6 Artificial Neuron

By altering the weights between the linked neurons, the network is trained and updated using backpropagation [39] techniques to reduce the difference between the output from the predictive model and the target value. The back-propagation process error during regression problem is often measured using the mean square error (MSE) [40] or mean average error (MAE) [41]. Iterative backpropagation is used to reduce the predictive model's MSE or MAE.

By finding patterns in the data gathered from multiple sources, machine learning models have been demonstrated to be helpful for forecasting and evaluating structural performance, identifying structural condition, and directing preventative and recovery decisions [42]. One of the engineering challenges includes the nonlinear hysteretic systems. For structural design, estimating the maximum response of a nonlinear hysteretic system under stochastic excitations is crucial. Although a nonlinear time history analysis is the most accurate way to predict the outcomes in many circumstances, the approach's adoption in standard engineering practice is hampered by high computational costs and modeling time. Therefore, application of machine learning to structure design and performance evaluation was conducted by various research.

Back in the 1990's, Anderson et al (1997), trained an artificial neural network model to predict the bi-linear moment-rotation characteristics of minor axis steel connections. The minor-axis-beam-to-column connections behavior of steel frames is more critical compared to buckling of columns, however there is no precise method to approximate this phenomenon. The ANN model was trained using the parameters gathered from actual conducted experiments [43]. Matsuda et al (1998), proposed a machine learning (ML) model for predicting the maximum lateral displacements of a hypothetical three-story base-isolated moment frame building subjected to earthquakes. Three characteristics that describe the behavior of isolation systems such as the ratio of initial to post-yield stiffness, normalized characteristic strength, and post-yield period as well as the average spectral accelerations at five chosen periods were considered as the input variables [44].

Then, Fahmy et al (2016), proposed a method using artificial neural network to suggest design components of orthotropic steel bridge deck with 0.008 training error loss. Eight different types of orthotropic decks with different dimension combinations were considered as design parameters. Then, eight design performance and safety evaluation were conducted to prepare the dataset. It was proven that the proposed method reduces the time and effort of engineers to determine the most economic and safe dimension of the orthotropic bridge deck [45]. Kim et al (2020), proposed a deep neural network using convolutional neural networks to predict the structural response of SDoF model system with influence of earthquake excitation. The proposed method effectively predicts the response without performing the nonlinear time history analysis [46].

By simulating multiple artificial earthquakes, an ANN model for forecasting the seismic responses of buildings based on the correlation between ground motion and the structures response was proposed. The ANN model is used to predict the seismic response of four multi-degree-of-freedom (MDOF) system with various natural periods [47]. A study proposed using artificial neural network to predict the seismic isolator's restoring force based on some input variables. The isolator was positioned between the superstructure and substructure using a two degree of freedom (2DOF) method. The trained ANN model's prediction of the HDR bearing's restoring force exhibits an excellent fit to the experimental data [48].

Hu et al (2022), proposed a machine learning based method to conduct probabilistic residual displacement-based design method of steel-moment-resisting frames using self-centering braces to improve the maintenance after an earthquake. A three-story MRF with self-centering braces was retrofitted using the proposed ANN design method approach and the seismic response was evaluated and found out to be effective [49].

1.4 Vision-Based Bridge Enhanced Structural Health Monitoring

After the bridge design phase and construction, it is important to conduct periodic maintenance to ensure its safety and extend the serviceability period. Japan's bridges are rapidly aging, out of over 700,000 bridges, 43% in Japan will have served for more than 50 years by 2023, in which the enhanced structural health monitoring techniques are needed [50]. In addition, 7.5% of the 617,000 bridges in the United States as of 2019 are in bad condition, and 42% have been in built for at least 50 years [51]. For other nations with significant amount of infrastructure built in recent decades that are experiencing aging and deterioration, effective and proper bridge maintenance has emerged as one of the most crucial issues. As bridges age, they may become less able to withstand catastrophic occurrences, which could have an adverse effect on both human safety and the economy.

Traditional bridge inspection is a labor-intensive, expensive, and resource-intensive operation that is conducted manually and relies heavily on equipment [52]. The development of inspection automation made possible by robotic imaging and machine vision has been expedited by recent breakthroughs in artificial intelligence (AI), and by various image-based deep learning (DL) techniques [53]. Researchers are continuously using cutting-edge technology to enhance the bridge inspection procedure. Recent developments in deep learning (DL) techniques that are based on machine vision have gathered a lot of interest from both researchers and practitioners.

Convolutional neural network (CNN)-based DL architectures [54] have shown improvements for damage segmentation and classification. It should be highlighted that most of these efforts use tuned or change existing architectures in order to embrace a data-centric strategy. For instance, a huge dataset of bridge inspection reports was utilized to identify different problems, such as corrosion, cracks, and spalling, on bridges [55]. To categorize various bridge defects including fractures and corrosion, YOLO v3 was used [56]. VGG-Unet model was recommended to segment the cracks in the rubber cover for bridge components, such as rubber bearings [57].

In addition to damage detection, damage quantification was also suggested by calculating the width of cracks in the structure element. Different convolutional neural network models were suggested to segment bridge components versus complicated scene in photos for bridge component recognition. Considerations included structural elements such columns, beams, and slabs as well as other structural parts and nonstructural elements [58]. Since there wasn't much genuine data available, synthetic data was created, and the parts and damages of the road bridges were retrieved and labeled. The components and bridge damage were then proposed to be segmented using a CNN model [59].

A method for bridge component segmentation using photos captured by a UAV was proposed in follow-up research [60]. The point cloud data was recreated using the photos, and the components were then categorized [61]. They trained a CNN model that can separate the bridge component from a point cloud data obtained by a laser scanner because the bridge component utilizing 3-dimensional data includes an extra dimension that can be used for further assessment. The deck, pier, and background categories were used to group the points in the point cloud data [62].

Structure from Motion (SfM) and other approaches, along with 3D model reconstruction, are helpful tools for visualizing the damage and its position on the bridge. To segment the components of a bridge's point cloud data using the DeepLab v3+ model, Chun et. al. proposed a method [63]. The segmented components could then be reflected to the original point cloud model. By using deep learning techniques to identify damages in 2D images and then project them back onto the 3D model, Yamane and Chun improved the process even further [64].

1.5 Proposed Methods and Objective of this Study

To overcome various challenges in the optimization and selection process during bridge seismic isolation design, machine learning-based approach can be used. Using the numerical data simulation from various nonlinear models, HDR-S quasi-static loading data, and hybrid simulation results, this work proposed an enhanced seismic design machine learning based approach.

The selection of nonlinear model for isolation bearing appears to be challenging for engineers because some nonlinear models are complex due to the inclusion of various parameters including temperature variations, deterioration, and stiffness degradation. After the nonlinear model selection, the initial parameter assumption using conventional optimization method highly influenced the convergence and dependent on the engineer’s knowledge and expertise. Engineers frequently select bilinear models or other straightforward method like equivalent linearization. Therefore, this study proposed a machine learning based approach for nonlinear model classification, nonlinear parameter optimization method using AI, creation of the nonlinear model library, and proposed numerical data generation.

The design parameter adjustment is a trial-and-error process which takes a lot of time and highly dependent on some initial parameters and engineer’s experience. Furthermore, this study proposed a machine learning based design parameter proposal and performance evaluation without repeating the nonlinear simulation.

After the design period, periodic inspection and maintenance is another important task. However, traditional bridge inspection needs a lot of manpower, equipment, and money in which enhanced method is needed by utilizing available technologies and innovation. Also, recent vision-based bridge damage detection focuses on 2-D damage classification and segmentation however further evaluation and localization should be explored. Therefore, this study proposed an enhanced bridge inspection and maintenance method by utilizing different technologies such as Unmanned Aerial Vehicle (UAV), Deep learning for damage segmentation, Structure-from-Motion (SfM), and Mixed Reality.

To visualize the corrosion from the RGB photos of a steel bridge, this study suggested a method using DeepLab v3+, SfM, and usage of UAV for data gathering. The 3D bridge model was then used to visualize the segmented corrosion damages. To determine the location of the damages, the feature points of the damage were projected onto a 3D bridge model that was recreated using structure from motion. The 3D model was also saved and is viewable remotely via a mixed reality platform. However, since the corrosion was divided into segments using 2D photos, SfM must be used to reconstruct the 3D bridge model, which can take some time depending on the technology. In continuation, point cloud data for structure's component and damage segmentation has been gathered for future studies.

CHAPTER 2: PROPOSED AI ENHANCED BRIDGE SEISMIC DESIGN

After the 1995 Kobe earthquake, Japan has increased the use of seismic isolation devices, such as high damping rubber (HDR-S) bearings for highway bridges [65]. Numerous new devices for structure's isolation and vibration control have been proposed over the last years. During the bridge isolation design process as shown in Figure 7, the research and development phase, design phase, and the associated steps are required before the installation of these isolation devices to the structure. The hysteretic behavior of seismic isolators has a significant impact on the seismic performance of seismically isolated structures. Standard experiments should be conducted to examine the device's hysteresis behavior during the R&D phase. Then, as a common method in design practice, engineers will choose some practical available hysteresis models.

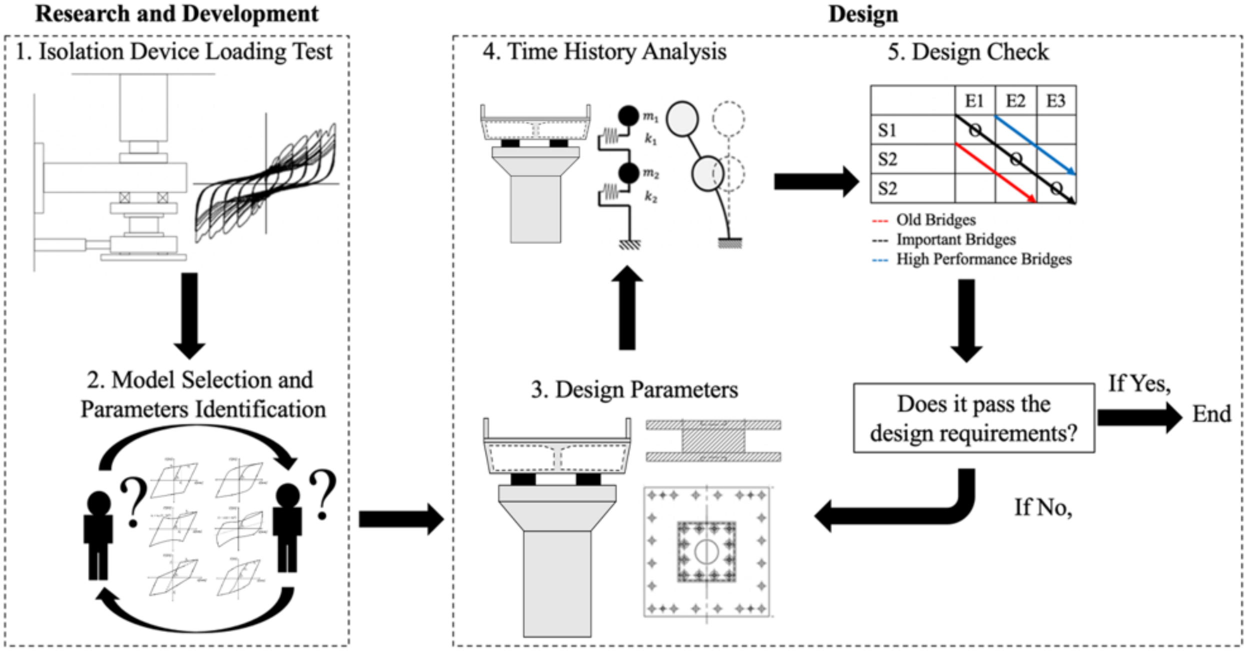


Fig. 7 Overview of Bridge Isolation Design

Since the equivalent stiffness can be directly observed and measured from the hysteresis, the parameters may be simply estimated using bilinear model. Although in some complex phenomena, such as the Mullins effect, stiffness degradation, temperature effect, etc., cannot be seen directly on the hysteresis, therefore, more sophisticated nonlinear model is necessary. This study proposed various application of artificial intelligence to enhance some process in bridge seismic isolation design and maintenance. Firstly, the researcher proposed a machine learning-based approach nonlinear model selection, wherein AI (Artificial Intelligence) model will be pre-trained to understand how the hysteresis loops of different nonlinear model looks like. Then, the trained ANN model will be use to classify the nonlinear model of some hysteresis loop of real experiment data.

Based in Japan bridge seismic control design manual [27], the equivalent stiffness and damping of the seismic device were specified using a bilinear model and the equivalent linearization approach. The list of isolation devices was limited, and newly developed seismic isolation devices and dampers were not included, thus, the identification of nonlinear model and parameters were highly dependent on engineer's expertise and experience. Additionally, by comparing the suggested model's nonlinear characteristics with the seismic isolator's experiment data, the nonlinear parameters should be evaluated before application. Developers should suggest new numerical models when the reproducibility of conventional models for those devices is insufficient. These new models are typically more complex, and it may be more challenging to identify their parameters. Some methods are used for parameter optimization like KH Method [66], Newtons method [67], Curve Fitting method [68], however the initial parameter selection is the critical part.

The design process will begin with reliable nonlinear models and design specifications. It is a process where the selection of the design parameters is heavily reliant on the expertise and experience of the engineer. To determine the initial design specifications, factors including bridge spans, pier counts, pier heights, and girder types will be taken into consideration. Engineers will also suggest preliminary parameters such as the kind of seismic devices, rubber quality, steel plate thickness, and elastomeric layers. Next, a time history analysis [69] of the bridge model subjected to design earthquakes should be conducted to assess its seismic performance in which a high level of structural dynamics expertise is needed. The design parameters will be continuously altered when the capacity exceeds the design needs and if the seismic performance of the structure could not meet the design requirement. Continuous design parameter adjustment is a trial-and-error procedure that takes a lot of effort and engineering expertise [70].

This study proposed a machine-learning based approach to address the problems of nonlinear model selection, the optimization method used for nonlinear parameter identification, and the trial-and-error process during seismic performance check to pass the required design standard. The proposed AI enhanced approach (AI) are shown in Figure 8.

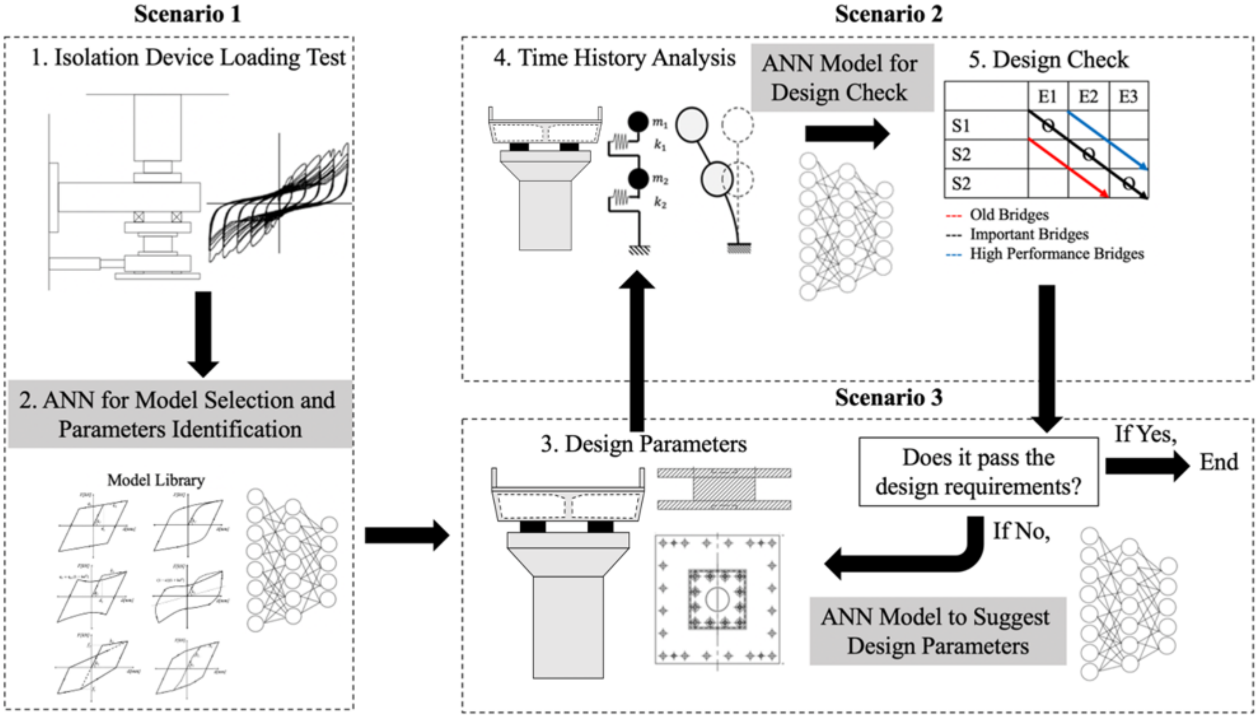


Fig. 8 Proposed ANN Models for Enhanced Bridge Seismic Isolation Design

This research proposed three situations in which the conventional design process could be improved by ANN approach. Under scenario 1, the HDR-S quasi-static loading data can be used to classify a nonlinear model and identify parameters using the trained ANN models. Before using additional optimization methods like the KH method [63], Newton's method [64], or curve fitting method [65], the trained ANN model will serve as an improved optimization method or support to recommend the initial parameter assumption. The creation of a library of nonlinear models with appropriate parameters will serve as a guide for engineers and be applicable to different kinds of seismic isolation devices aside from HDR-S.

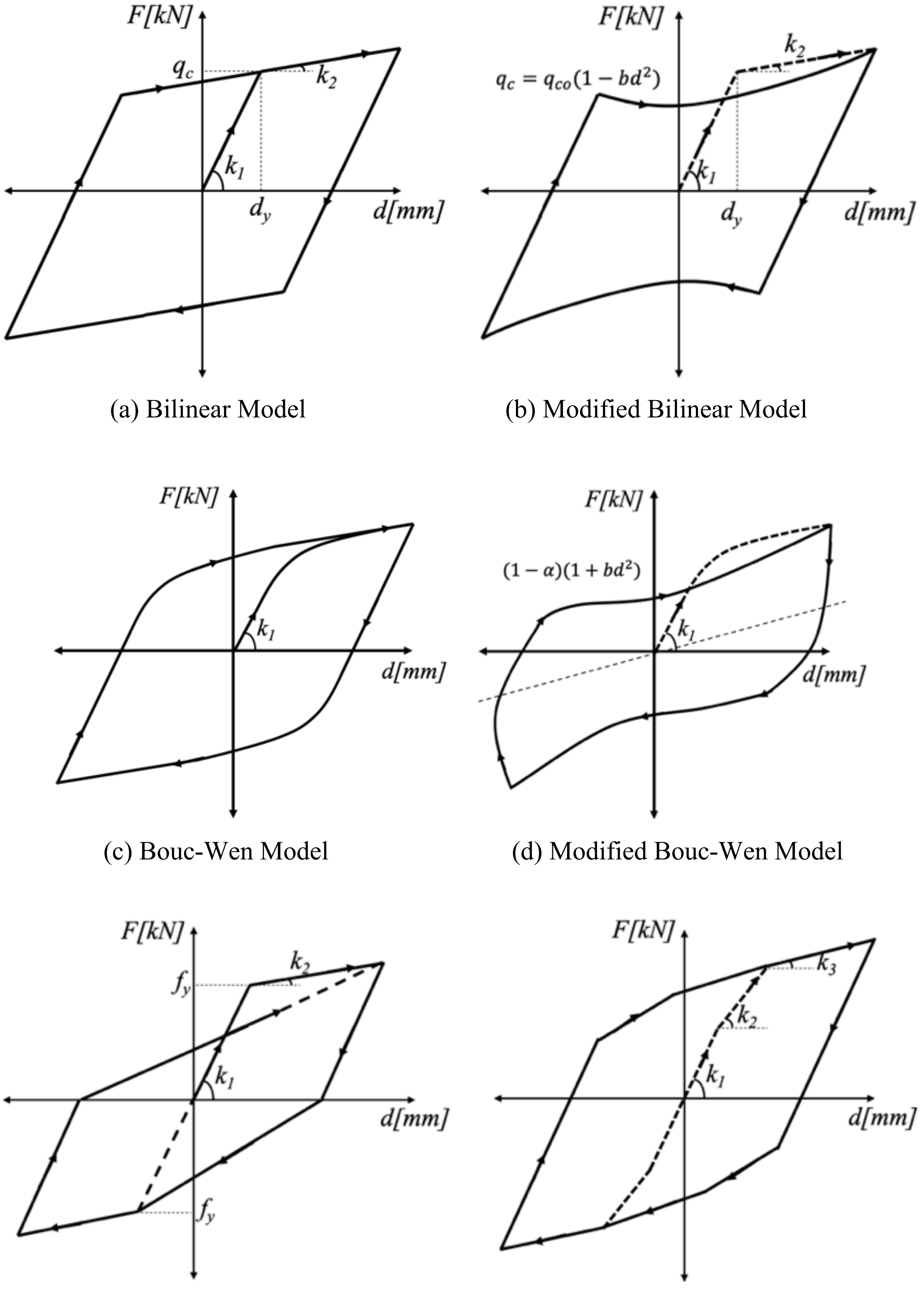
In scenario 2, an AI model was trained to determine if the design performance criteria could be automatically satisfied by the design parameters using only a few input factors, such as the structure and isolator's design specifications under three design earthquakes, without repeating the time history analysis. This ANN model was trained using simulations of a variety of seismic isolator specifications and big datasets of structural parameters. This will assist in solving the trial-and-error design performance testing procedure.

Under scenario 3, the ANN model will suggest design parameters after passing the design requirements. This is known as "inverse approach" after the design check. The seismic isolators' cross-sectional areas, rubber total thickness, and other essential design characteristics will all be included in this ANN model. The number of parameters considered, such as the complexity of the bridge structure design parameters, design earthquake data, and the design specifications of the seismic isolator affects how accuracy of the ANN model.

This work made a preliminary progress for each of the three proposed AI-enhanced components. Under scenario 1, which trained an ANN model to conduct model selection and parameter identification of the hysteretic behavior of the HDR-S bearing at three different temperatures. An advanced optimization strategy that uses the vast dataset offered to learn is the machine learning-based approach. This will make it much easier for engineers to choose which nonlinear model to use and will speed up the optimization process for identifying nonlinear parameters. The trained ANN model can be used for all other types of elastomeric bearings, not just HDR-S bearings, for both model selection and nonlinear parameter identification. Under scenario 2 and 3, the trained ANN model could check the bearing's performance, predict the bearing's strain response on the structure under three design earthquakes, and suggest bearing's design parameters. This will further be discussed in the following chapters.

2.1 Nonlinear Model Library

The numerical equations and hysteretic behavior of the seven nonlinear models taken into consideration for this study are covered in this chapter. Figure 9 represents the nonlinear models that were taken into consideration for this work, which includes Bilinear model [17], Modified Bilinear model, Bouc-Wen model [19], Modified Bouc-Wen model [20], Clough model [68], and Trilinear model [69]. The numerical equations of each nonlinear model were used to simulate the hysteretic behavior of the HDR-S bearing, and the correctness of the curve fitting was assessed by computing the contribution rate [73].



(a) Bilinear Model

(b) Modified Bilinear Model

(c) Bouc-Wen Model

(d) Modified Bouc-Wen Model

(e) Clough Model

(f) Trilinear Model

Fig. 9 Library of Different Nonlinear Models

2.1.1 Bilinear Model

The hysteretic behavior of HDR-S bearing and sliding bearing are frequently represented by this model. The restoring force during the elastic stage is dependent on the initial stiffness k_1 and displacement d. The plastic stage began at the second stiffness $k_2$when the yielding force q_c was exceeded.

Equation 1 states that the total restoring force F is equal to the sum of the elastic force F_e and plastic stage force F_p . Then, according to equation 2, α is the proportion between the secondary stiffness and the initial stiffness.

$$F = F_e + F_p \tag{1}$$

$$k_2 = \alpha k_1 \tag{2}$$

Every step will result in an update of the plastic force, as stated in equation 3. The condition of the elastic and plastic state changes, in which the yielding force q_c has a significant role in the yielding and unloading state. Figure 9-a shows the bilinear model's hysteresis.

$$F_{pn+1} = F_{pn} + (1 - \alpha)k_1 \Delta d_{n+1} \tag{3}$$

Table 1 displays the range of bilinear model parameters for data generation. Prior to ANN training, these parameter range were carefully chosen to produce realistic hysteresis curves. The parameters must be fit to the HDR-S quasi-static loading data, and they have a significant impact on the training process.

Table 1. Bilinear Model Parameter Range for ANN Training

α	$G_1(N/mm^2)$	q_c
0.05 – 0.12	8 -20	0.5- 2.0

2.1.2 Modified Bilinear Model

The modified bilinear (MBL), as depicted in Figure 9-b, contains the same parameters as the bilinear model but have an additional parameter b as shown in equation 4. The parameter b represents the pinching effect brought on by the abrupt decrease in stiffness during the change in displacement, and it have an impact on the initial yield force q_c. The total yielding force q_c is affected by the initial yielding force q_{co} and the pinching effect. Table 2 displays the adjusted bilinear parameter ranges that will be utilized for data generation before ANN training.

$$q_c = q_{co} + bd^2 \tag{4}$$

Table 2. Modified Bilinear Model Parameter Range for ANN Training

α	$G_1(\mathrm{N}/mm^2)$	*qc (N)*	b
0.05 – 0.12	8 -20	0.5- 2.0	0.1 – 0.5

2.1.3 Bouc-Wen Model

Rheological dampers and various base isolation devices, like HDR, frequently exhibit hysteretic behavior, which is typically represented by the Bouc-Wen model as shown in Figure 9-c. It represents softening brought on by deterioration of rigidity. The smoothness of the curve is greatly influenced by the plasticity parameter *z*, and the stiffness change had an impact on the plastic force. Equations 5 and 6 display the formula for this model.

$$F = \alpha kd + (1 - \alpha)kz \tag{5}$$

$$\dot{z} = A\dot{d} - \beta|\dot{d}||z|z - \gamma\dot{d}z^2 \tag{6}$$

In this equation, *F* is the restoring force, *d* is the displacement, *k* is the stiffness coefficient, and α is the ratio of initial to post yield stiffness. ,*β* and *γ* specifically governs the softening and hardening of the hysteresis. $\dot{d}$ and $\dot{z}$ are change in displacement and plasticity with respect to time. To observe the relationship of parameter β and γ, we divide both sides by $\frac{dt}{dd}$ as shown in equation 7 ,equation 8, and equation 9.

$$\frac{dz}{dt} = A\frac{dd}{dt} - \beta\left|\dot{d}z^2\right|z - \gamma \mathrm{d}z^2 \tag{7}$$

$$\frac{dz}{dd} = A - \beta\frac{|\mathrm{dd}|}{dd}\frac{|\mathrm{z}|}{z}z^2 - \gamma\frac{dd}{dt}z^2 \tag{8}$$

$$\frac{dz}{dd} = A - z^2(\beta\frac{|\mathrm{dd}|}{dd}\frac{|\mathrm{z}|}{z} + \gamma) \tag{9}$$

When z_n is maximum as shown in Figure 4, and given the assumption that $\frac{dz}{dd} = 0$ as shown in equation 10, the maximum plastic yielding parameter $d_y = z_n$ will be equal to equation 11.

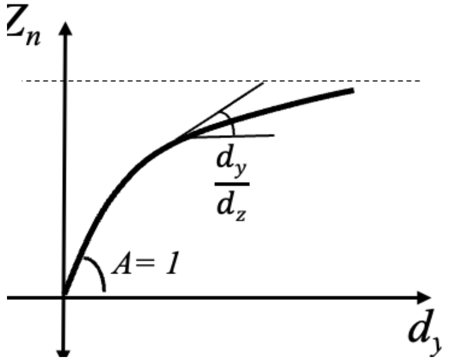


Fig. 10 Behavior of Plasticity Parameter

$$\left(\beta\frac{|\mathrm{dd}|}{dd}\frac{|\mathrm{z}|}{z} + \gamma\right)z^2 = A \tag{10}$$

$$z = \sqrt{\frac{A}{\beta + \gamma}} \tag{11}$$

Therefore, to avoid getting an error value for equation 10, the denominator $\beta + \gamma$ should follow the condition shown in equation 12.

$$\beta + \gamma > 0 \tag{12}$$

On the other hand, when A = 1, and $\frac{|dd|}{dd}\frac{|z|}{z}$ is equal to 1 or -1, this happens between the loading and unloading period as shown in Figure 11. The change of z with respect to the displacement are equal to equation 13 and equation 14 depending on the direction of loading.

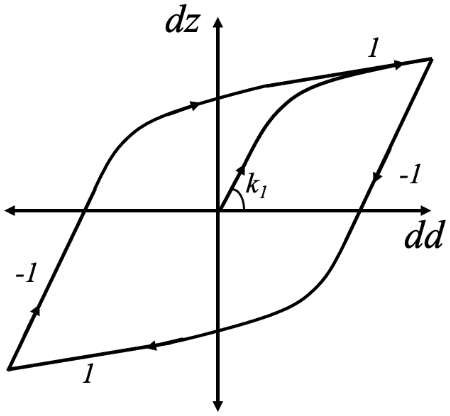


Fig. 11 Plasticity Behavior Effect on the Direction of Loading

$$\frac{dz}{dd} = 1 - (\beta + \gamma)z^2 \tag{13}$$

$$\frac{dz}{dd} = 1 - (-\beta + \gamma)z^2 \tag{14}$$

Given the condition of equation 13, the relationship of $\beta + \gamma$ is same as equation 12. However, using the condition of equation 13, the relationship of $\beta + \gamma$ is showin in equation 15. Table 3 displayed the Bouc-Wen model parameter ranges.

$$\beta - \gamma > 0 \tag{15}$$

Table 3. Bouc-Wen Model Parameter Range for ANN Training

α	$G_1(\mathrm{N}/mm^2)$	β	γ
0.1 – 0.6	0.5 - 9	0.1 – 0.5	0.1 – 0.5

2.1.4 Modified Bouc-Wen Model

Modified Bouc-Wen model as shown in Figure 9-d, includes pinching effect which can be observe on the parameter *b* and softening which represents by parameters *β* and *γ*. The restoring force can be computed using the equation 16. Then the parameter *z* as shown in equation 17, controls the smoothness of the curve same as the original Bouc-Wen model.

$$F = \alpha k d + (1 - \alpha) k (1 + b d^2) z \tag{16}$$

$$\dot{z} = A\dot{d} - \beta|\dot{d}\, z|z - \gamma \dot{d} z^2 (A = 1) \tag{17}$$

where *F* is the restoring force, $\dot{d}$ is displacement, *z* is the plastic component, *k* is the initial elastic stiffness, *b* is caused by the shearing deformation which result to the change of critical force and produce pinching effect. *β* and *γ* are based on the yield displacement, while A is generally equal to 1. The Modified Bouc-Wen model parameters ranges was shown in Table 4.

Table 4. Modified Bouc-Wen Model Parameter Range for ANN Training

α	G_1(N/mm^2)	β	γ	b
0.1 – 0.6	0.5 - 9	0.1 – 0.5	0.1 – 0.5	0.1 – 0.5

2.1.5 Trilinear Model

The behavior of a steel or reinforced concrete column subjected to external cyclic loading is commonly represented by the trilinear model [74], [75], [76] as shown in Figure 9-f. It shares the same backbone elements as the bilinear model, but it also has a third stiffness, k_3, that represents the strain hardening properties. The static response was idealized in the first loading while evaluating the behavior of reinforced concrete, and the subsequent behavior was considered by taking a set of rules into account.

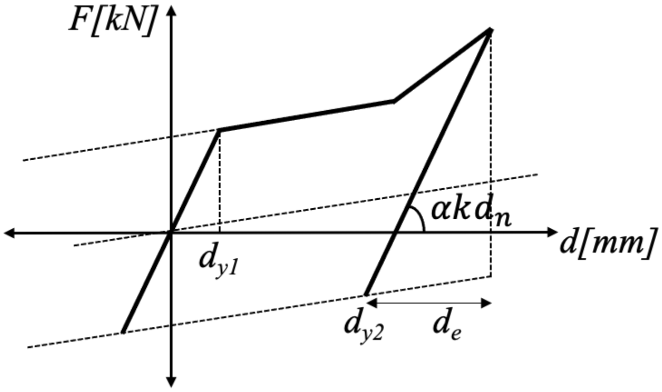


Fig. 12 Hysteretic Behavior of Trilinear Model

when the current displacement d is greater than the yield displacement d_y, the current restoring force F_{pn+1} is equal to equation 18. Where in $k_1$is the initial stiffness, α is the first stiffness ratio, and F_{pn} is the previous restoring force. Then, after the yielding displacement until the maximum displacement, the restoring force is equal to equation 19. Wherein k_3 is the third stiffness and μ is the ductility.

$$F_{pn+1} = F_{pn} + (d - d_y)\alpha k_1 \tag{18}$$

$$F_{pn+1} = (1 - \mu * \alpha - \alpha)k_1 d_{y1} + k_3|d| - \mu d_{y1} \tag{19}$$

The unloading will begin as depicted in Figure 6 when the displacement change turns negative. To compute the unloading point d_{y2} , the distance d_e which measures from the maximum displacement d_n to the unloading point should be computed first as shown in equation 23. This study considered two different scenarios, when $\beta = 0$ and $\beta > \alpha$. The range of parameters for this model was shown in Table 5.

$$d_e = \frac{k\, d_n - F_n}{(1 - \alpha\,)k} \tag{20}$$

Table 5. Trilinear Model Parameter Range for ANN Training

Trilinear Model	α	$G_1(\mathrm{N}/mm^2)$	d_y	μ	β
$\beta = 0$	0.1 – 0.6	3.0 – 8.0	0.3- 0.5	2.0 – 5.0	0
$\beta > \alpha$	0.1 – 0.6	3.0 – 8.0	0.3- 0.5	2.0 – 5.0	0.1 – 0.3

2.2.6 Clough Model

Clough model [77] as seen in Figure 9-e is being used to simulate the behavior of reinforced concrete columns when they are subjected to cyclic loading, such as earthquakes. The unloading stiffness is maintained equal to the initial stiffness and the maximum response point represents the stiffness degradation [78]. The steps of the hysteretic rules of this model, as depicted in Figure 7, are as follows:

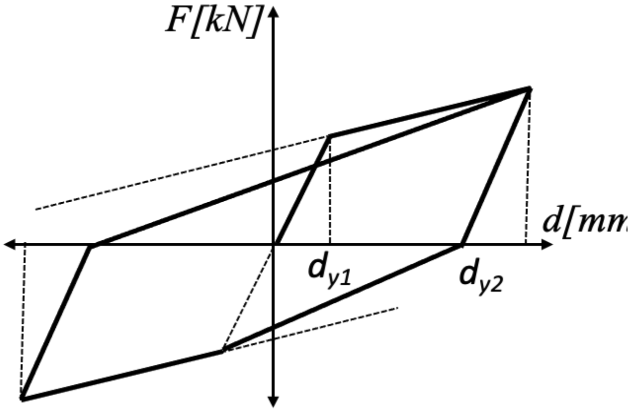


Fig. 13 Hysteretic Behavior of Clough Model

During the plastic stage, when the current displacement d is greater than the positive yield displacement d_{y1} , the current restoring force F_{pn+1} is equal to equation 21. Where k is the initial stiffness, α is the stiffness ratio, and F_{pn} is the previous restoring force.

$$F_{pn+1} = F_{pn} + (u - d_{y1})\alpha \mathrm{k} \tag{21}$$

On the other hand, when the current displacement d is less than the negative yield displacement d_{y2}, the restoring force F_{pn+1} is equal to equation 22.

$$F_{pn+1} = F_{pn} + (d - d_{y2})\alpha\mathrm{k} \tag{22}$$

For the plastic stage, when the difference of the current and previous displacement is greater than zero, and the current displacement is larger than the positive yield displacement d_{y1}, the restoring force F_{pn+1} is equal to equation 23, wherein d_{t1} is the positive displacement the previous restoring force is positive. However, when the current displacement is less than the negative displacement d_{t2} and the previous restoring force is negative, the restoring force F_{pn+1} is equal to equation 24. When the negative current displacement d_{t2} at zero restoring force happens, the behavior will lead to the positive yield displacement d_{y1}. The range of parameters used for data gathering under clough model was shown in Table 6.

$$F_{pn+1} = \frac{F_{pn}}{d_{t1} - d_{y1}}(d - d_{y1}) \tag{23}$$

$$F_{pn+1} = F_{pn} + \frac{F_{pn}}{d_{t1} - d_{y1}}(d - d_{t2})\,\alpha \tag{24}$$

Table 6. Clough Model Parameter Range for ANN Training

α	$G_1(\mathrm{N}/mm^2)$	d_y
0.1 – 0.6	3.0 – 8.0	0.3- 0.5

2.2 HDR-S Loading Tests

This study considers two types of loading test, quasi-static loading [76] and hybrid simulation [77]. This segment will discuss the HDR-S specimen, quasi-static loading test specifications, hybrid simulation procedures, and input design earthquake considered.

2.2.1 High Damping Rubber Bearing Specimen

Using a quasi-static loading test, the nonlinear behavior of HDR-S bearings under different amplitude and three different temperatures were considered. Figure 14 illustrates the cross-sectional area of the HDR-S specimens used in this investigation, which is 0.0576 mm2. The cover is 5 mm thick on both sides, hence the cross section's actual size was 240 mm by 240 mm. The laminated elastomeric bearing has a total height of 96 mm and is made up of six layers of rubber that are each 30 mm thick and five inner steel plates. The vertical pressure is 6 Mpa, the first shape factor is 12 and the second shape factor is 8, and the rubber material is G12. The material properties of steel and rubber used were shown in table 7. Then, five-cycle amplitudes had values of 50%, 100%, 150%, 200%, and 250%. In order to exclude the impact of the cross-section and rubber thickness on the data production, the HDR-S hysteresis curve was expressed in terms of displacement and force, which were converted to shear stress and shear strain.

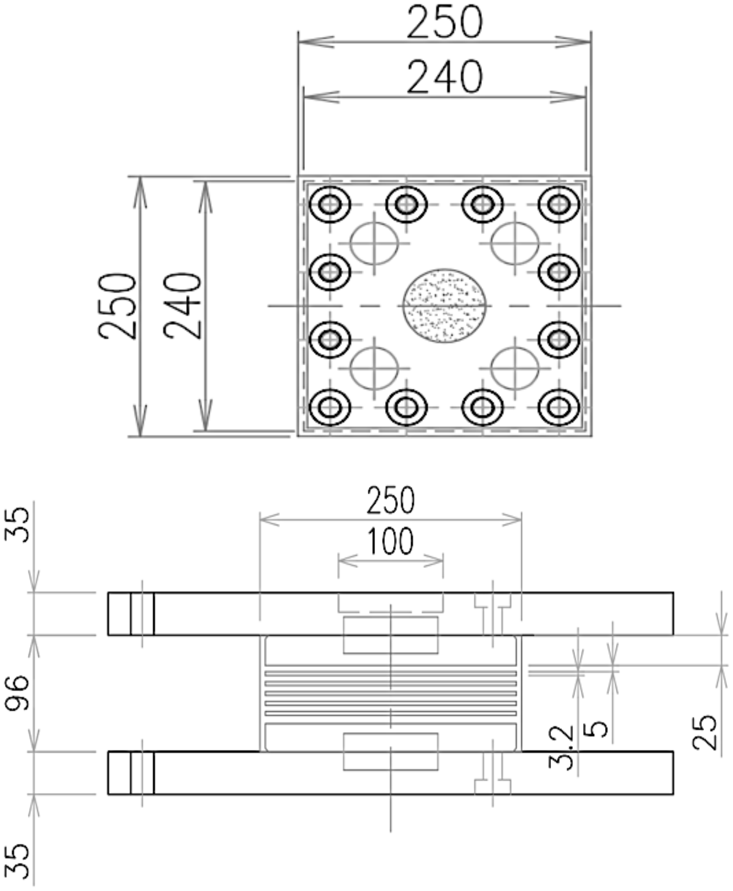


Fig. 14 HDR-S Specimen Specifications

Table 7. HDR-S Material Properties

Material	Density (kg/m^3)	Specific Heat Capacity (J/kg·K)
Rubber	1146	1732
Steel	7740	432

2.2.2 Quasi-Static Loading

Figure 15 shows the setup and equipment for quasi-static loading test. The control panel separates the signal for the load cell and measures the displacement and force while the loading system control PC records the displacement and force response from the HDR-S sensor. For the machine parts, the horizontal actuator simulated sine waves while the vertical actuator keeps the regular 6 Mpa vertical pressure. To maintain the desired temperature throughout the test, temperature chambers were assembled, and a cover was placed over the HDR-S specimen.

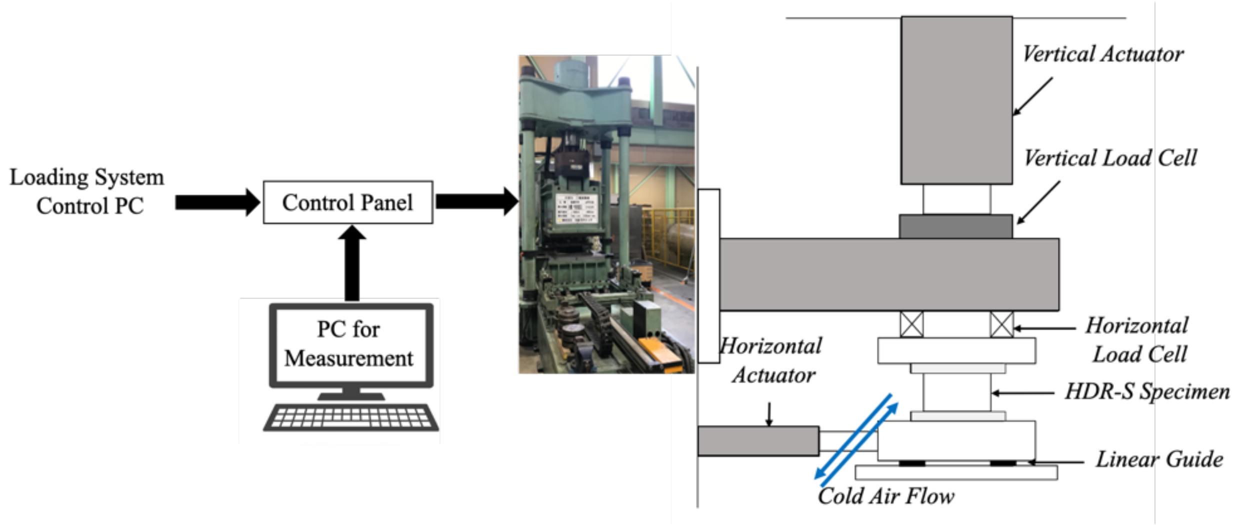
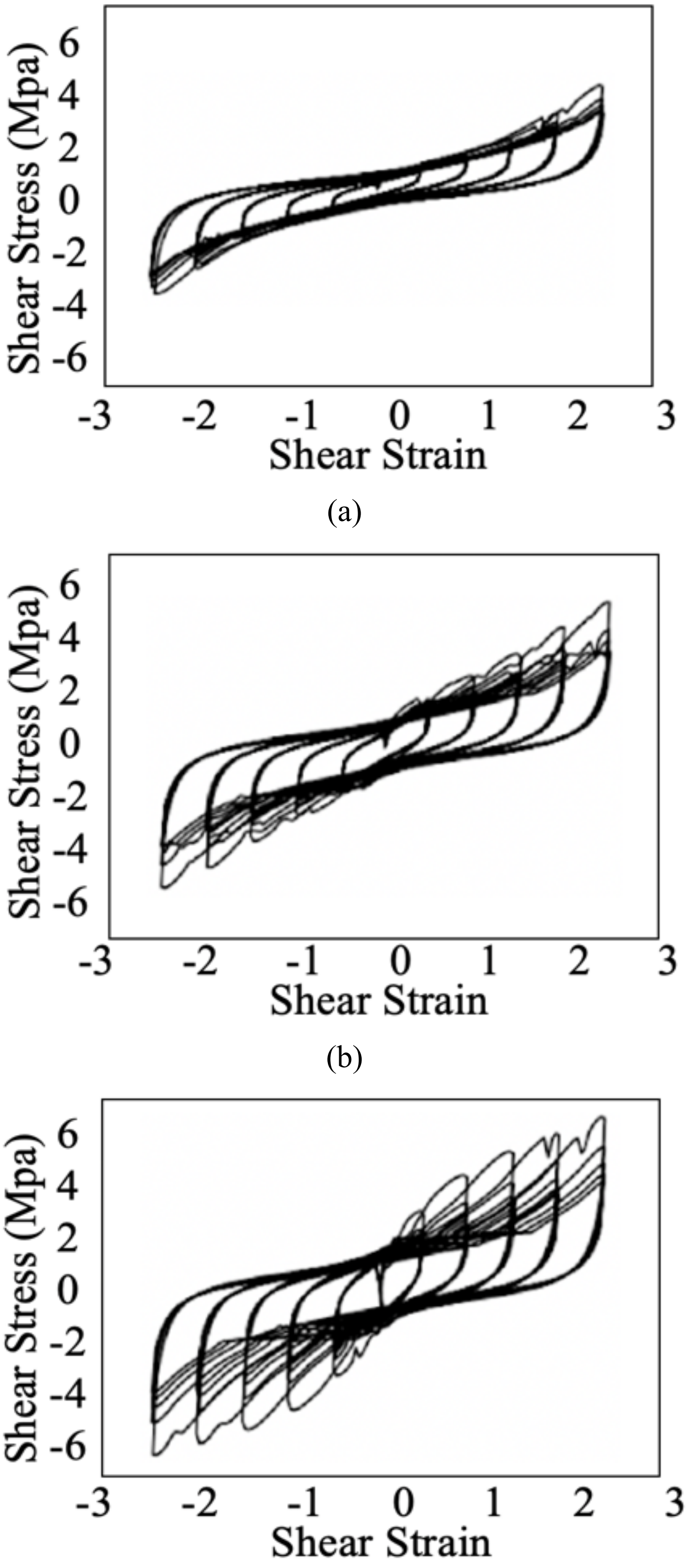


Fig. 15 Quasi-Static Loading Test Setup

Figure 16 displays the HDR-S quasi-static loading data at three different. Each of the five amplitudes range from 50% to 250% and has five cycles. Larger stiffness is visible at higher amplitudes. Furthermore, the first loop of each amplitude exhibits the Mullins effect.



(a)

(b)

(c)

Fig. 16 Quasi-Static Loading Test at (a) 23 °C (b) 0 °C (c) -20 °C

2.2.3 Hybrid Simulation

Hybrid simulation was used to observe the HDR-S hysteretic behavior installed in a hypothetical bridge with 6 Mpa vertical pressure under design earthquake. The bridge response to the input earthquake had an impact on the HDR-S nonlinear behavior during hybrid simulation. The input earthquake as shown in Figure 17, which is Kobe Japan Railway Takatori North-South direction [81] under level 2 type 2 G2 according to Japanese standard. Due to the limits of the experiment facility, a scale factor of 1/6 was utilized, and the similarity rule was applied in the simulation of the required parameters. The mass of the superstructure was determined using the 6 MPa standard vertical pressure and the mass of the pier is 1/3 of the mass of the superstructure. Figure 18 illustrates the two-degree of freedom (2DOF) model of the three-span steel bridge cross section. Table 8 provides an illustration of the 2DOF steel bridge's mass, stiffness, and damping properties.

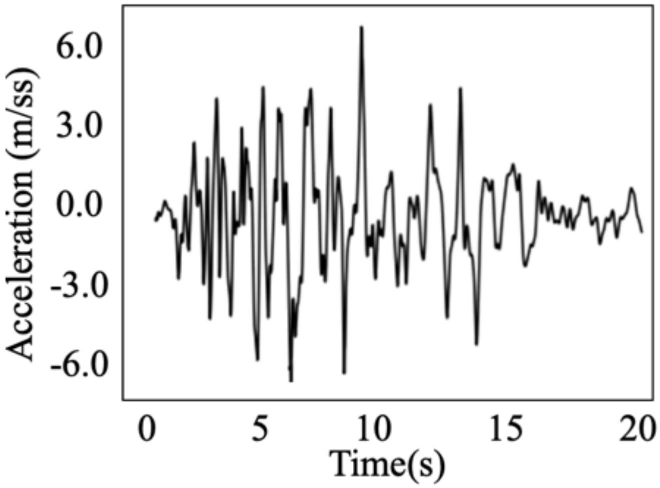


Fig. 17 Input Earthquake for Hybrid Simulation

Table. 8 2DOF Bridge Properties

Properties	Superstructure	Pier
Mass	m_1= 10156408.16 (Kg)	m_2= 10156408.16 (Kg)
Stiffness	$k_1 = 376320000$ N/m	$k_2 = 376320000$ N/m
Damping, h	$h_1 = 0.05$	$h_2 = 0.05$

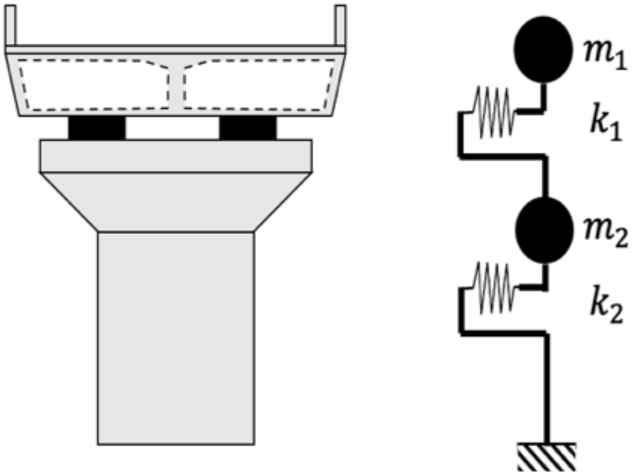


Fig. 18 Hybrid Simulation Bridge Model

Figure 19 illustrates the addition of a simulation computer, displacement regulator, and terminal block to the loading test apparatus for the hybrid simulation test. These three devices are crucial for simulating the response of a two-degree-of-freedom bridge when an earthquake wave is loaded in a horizontal direction. The structure's displacement and force were simulated and recorded for each time step of the loaded seismic wave. Figure 20 displayed the HDR-S hybrid simulation data.

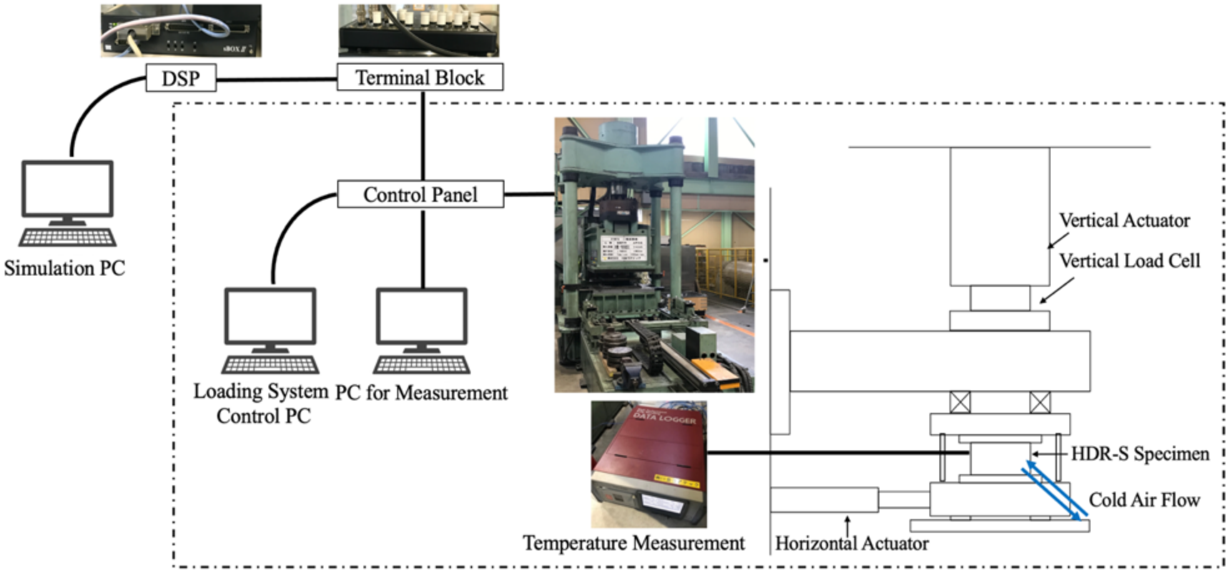
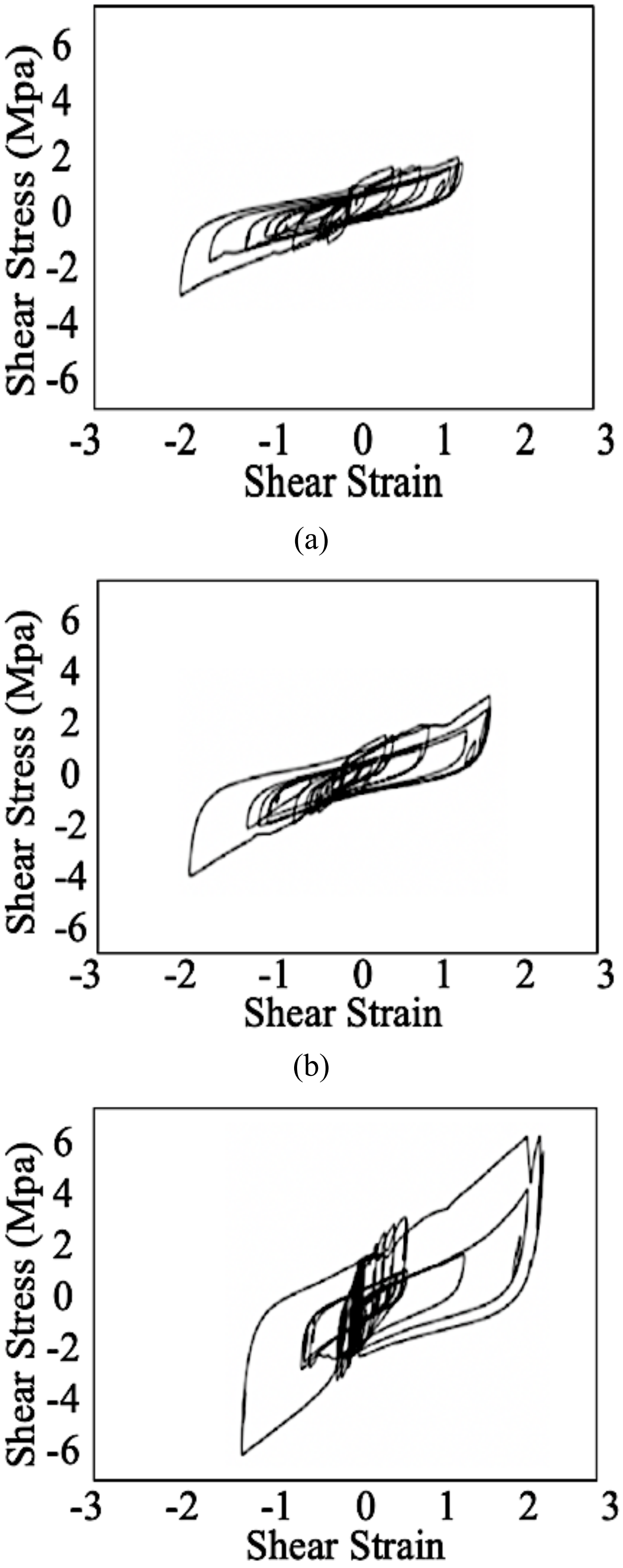


Fig. 19 Hybrid Loading Test Setup



(a)

(b)

(c)

Fig. 20 HDR-S Hybrid Simulation Result at (a) 23 °C (b) 0 °C (c) -20 °C

Chapter 3: Machine Learning Based Nonlinear Model Classification and Parameter Identification

This chapter includes the development of an advanced nonlinear classification and parameter optimization method based on machine learning that aims to classify the nonlinear model and suggest the nonlinear parameters of the HDR-S quasi-static loading data. Seven nonlinear models, including the Bilinear model (BL), Modified Bilinear model (MBL), Bouc-Wen model (BW), Modified Bouc-Wen model (MBW), Clough model (CL), and two conditions of the Trilinear model (TR1 and TR2), were used in this study.

Prior to ANN training, data from the nonlinear model library, which consists of seven (7) nonlinear models, will be generated through numerical simulation using each nonlinear model's formula and randomly chosen parameters based on the parameter ranges discussed in the previous section. The shear stress will be produced during the numerical simulation using the HDR-S shear strain data.

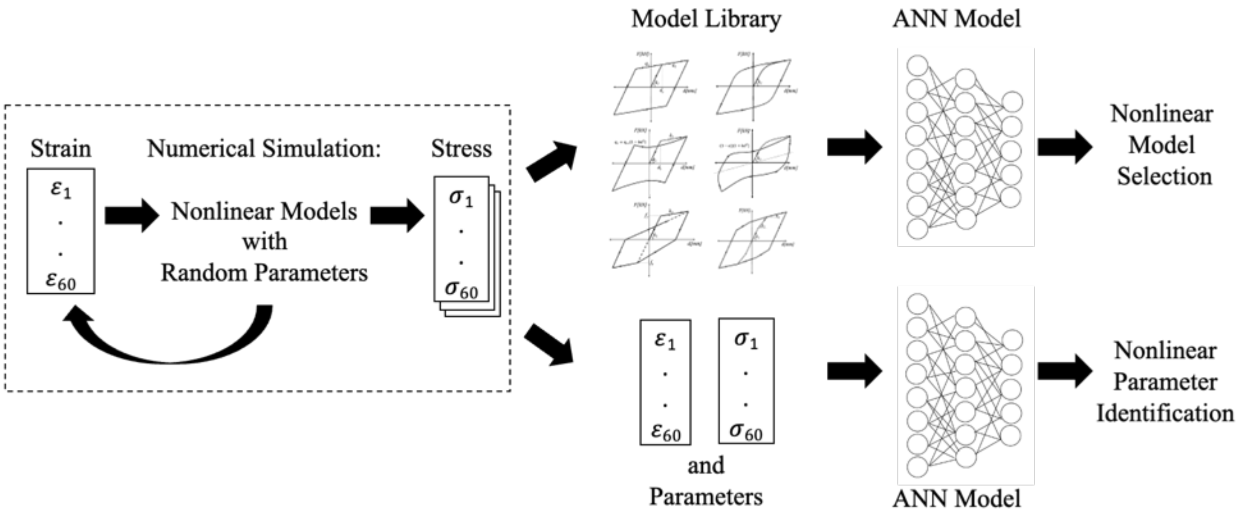


Fig. 21 Data Generation for Nonlinear Model Selection and Parameter Identification

The training data for choosing a nonlinear model will be the shear strain and stress on Figure 21 and repeated in each nonlinear model. On the other hand, the nonlinear parameter identification will make use of the identical shear strain and stress data together with the related nonlinear parameters.

3.1 Data Generation and Nonlinear Model Parameter Range

It was necessary to transform the HDR-S data from force and displacement to shear stress and strain in order to make it universal and remove the influence of the bearing's cross-section area and rubber thickness. Five different amplitudes—50%, 100%, 150%, 200%, and 250%—were represented in the data, and each amplitude had five loops. In order to prepare the loops for ANN training, they were split and normalized using the nearest neighbor normalization approach. The optimal size was determined by trial-and-error training of the neural network model, and it was found to be 60 data points each of shear stress and shear strain.

Due to the combination of 60 shear strain and 60 shear stress that were numerically simulated with the classified nonlinear model, the input data size for the ANN nonlinear model classification was 120, and the total training data for each nonlinear model is 150. The training process will be greatly influenced by the data size and values; therefore, normalization is typically necessary. Python was the data generating platform, which was used at the Google Collaboratory. Figure 22 visualized the data generation process. Using the shear strain from the experimental data, 60 points of shear stress will be generated during the numerical simulation.

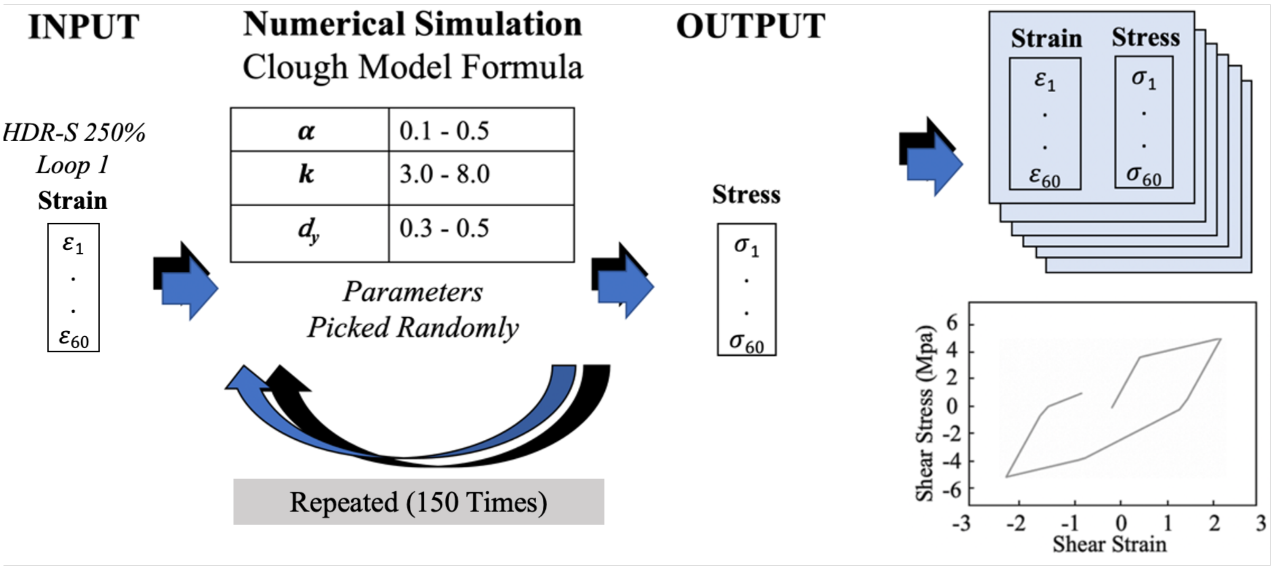


Fig. 22 Data Generation with Random Nonlinear Parameter Selection

3.2 ANN Model Architecture for Nonlinear Model Classification

The objective of this ANN training is to categorize the nonlinear model using the pair set of numerically simulated shear strain and shear force data. Bilinear (BL), Modified Bilinear (MBL), Bouc-Wen (BW), Modified Bouc-Wen (MBW), Clough (CL), and Trilinear (TR1, TR2) are the seven nonlinear models that needed to be classified.

Each nonlinear model's input size is 120, while the overall dataset is 1050. Figure 23 shows the input and output data of the ANN model with two hidden layers. The optimizer is Adam [82] which is suggested for classification problems and has a learning rate of 0.001. The loss function is categorical cross entropy [83], it categorizes the classes into 0 or 1, with a value close to 1 representing the nonlinear model with the highest probability.

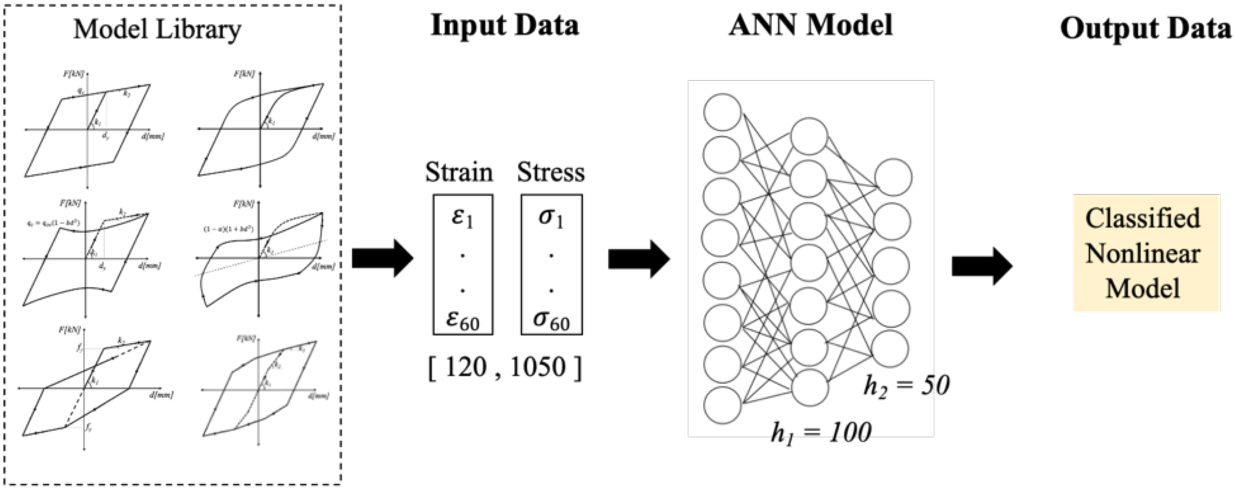


Fig. 23 Nonlinear Model Classification Input and Output Data Visualization

3.2.1 Nonlinear Model Classification Result and Evaluation

As shown in Figure 24, after 50 epochs, the validation accuracy was 0.99 and the validation loss was 0.03. Figure 25 displayed the confusion matrix, which represented the classification outcome of the true label and predicted label from the seven nonlinear models.

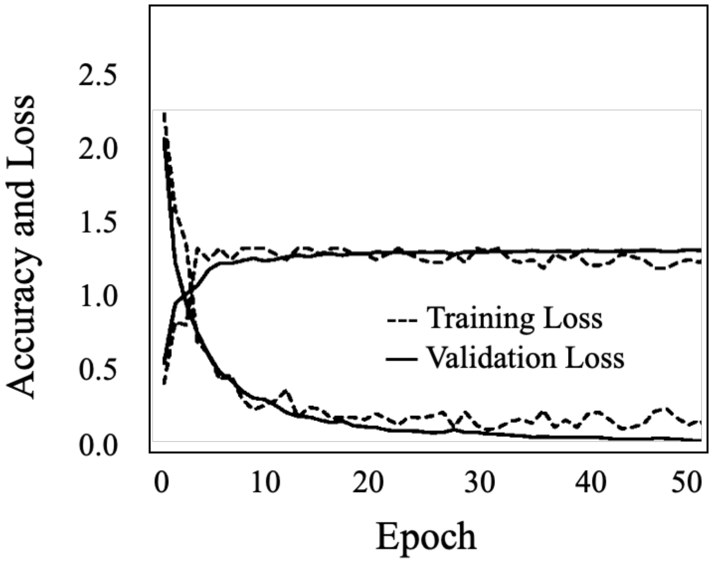
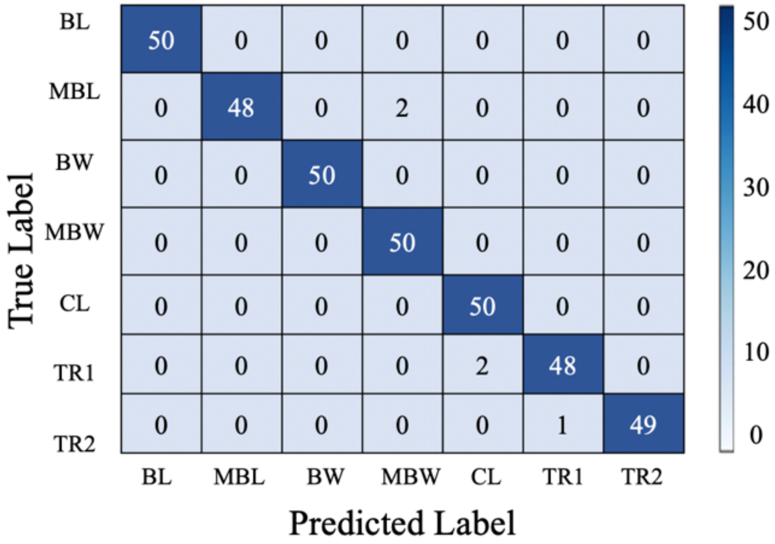


Fig. 24 Trained ANN for Nonlinear Model Classification Accuracy and Loss



True Label \ Predicted Label	BL	MBL	BW	MBW	CL	TR1	TR2
BL	50	0	0	0	0	0	0
MBL	0	48	0	2	0	0	0
BW	0	0	50	0	0	0	0
MBW	0	0	0	50	0	0	0
CL	0	0	0	0	50	0	0
TR1	0	0	0	0	2	48	0
TR2	0	0	0	0	0	1	49

Fig. 25 Confusion Matrix for Nonlinear Model Classification

The classification outcome was displayed in Figure 26 using the first loop at each amplitude of the HDR-S quasi-static loading data as an input to the trained neural network. The classified nonlinear model was bilinear at lower amplitude levels from 50% up to 150%, however at higher amplitude levels, the classified model was modified Bouc-Wen model. The Mullins effect and temperature impacts, however, are still not considered by the nonlinear models in use and will be in future research.

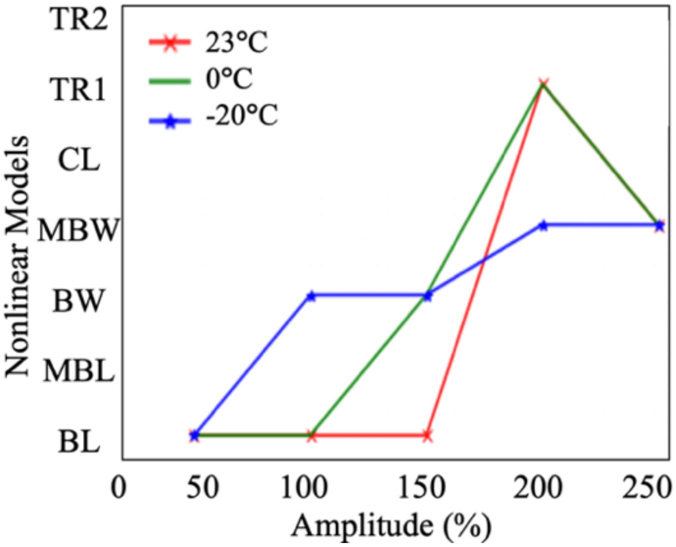


Fig. 26 Trained ANN Classified Nonlinear Model for 23 °C, 0 °C, and -20 °C

3.3 Machine Learning Based Nonlinear Parameter Identification

Artificial neural network training and results for identification of parameters with respect to HDR-S Quasi-Static Load Data Using Different Nonlinear Models are included in this chapter. The models taken into consideration were Modified Bouc-Wen model, Bilinear Model, Clough Model and Trilinear Model. Then, as compared to the experimental data, equivalent energy absorption and stiffness have been calculated.

3.3.1 Modified Bouc-Wen Model Nonlinear Parameters ANN Prediction Result

To represent the HDR-S nonlinear behavior numerically, Modified Bouc-Wen model was used. The shear stress σ is influenced by the strain ε, pinching effect by parameter b, hardening and softening due to stiffness degradation which are represented by parameters γ and β, and z which is the plastic component with A generally equal to 1. During the data generation for ANN training, the parameters were expressed to shear strain and shear stress, and the parameter ranges was shown in the previous section. Combination of 60 shear strain data and 60 shear stress with 1000 data size were used as an input data for the ANN training.

The ANN architecture as shown in Figure 27 consists of 120 by 1000 input features, 2 hidden layers, and five nonlinear parameters of modified Bouc-Wen Model as an output. The activation function used was ReLU [61] because it's a regression problem and specific values was required as an output. RMSProp was the optimizer used with a learning rate of 0.0001. The loss function was mean squared error and should be near to zero.

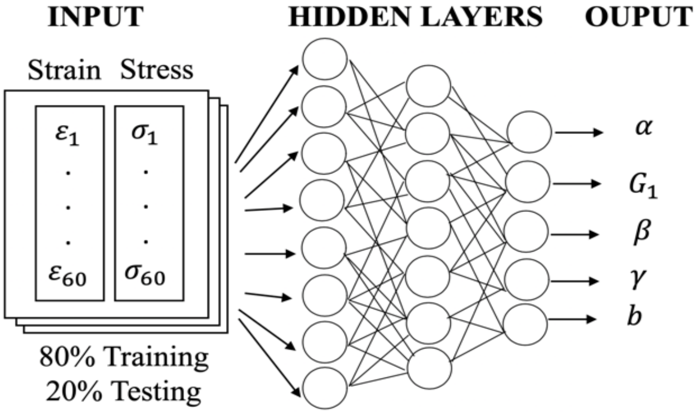


Fig. 27 ANN Model for Modified Bouc-Wen Nonlinear Parameter Prediction

After 200 epochs, the training loss was 0.047 and the validation loss was 0.134 as shown in Figure 28. To visually examine the accuracy of the trained model, the ANN predicted nonlinear parameters were plotted and the hysteresis was compared to the normalized HDR-S quasi-static loading data at 250% amplitude loop 1 as shown in Figure 29.

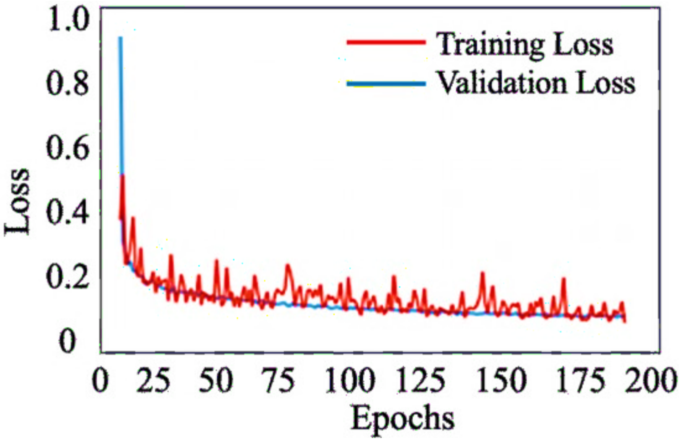
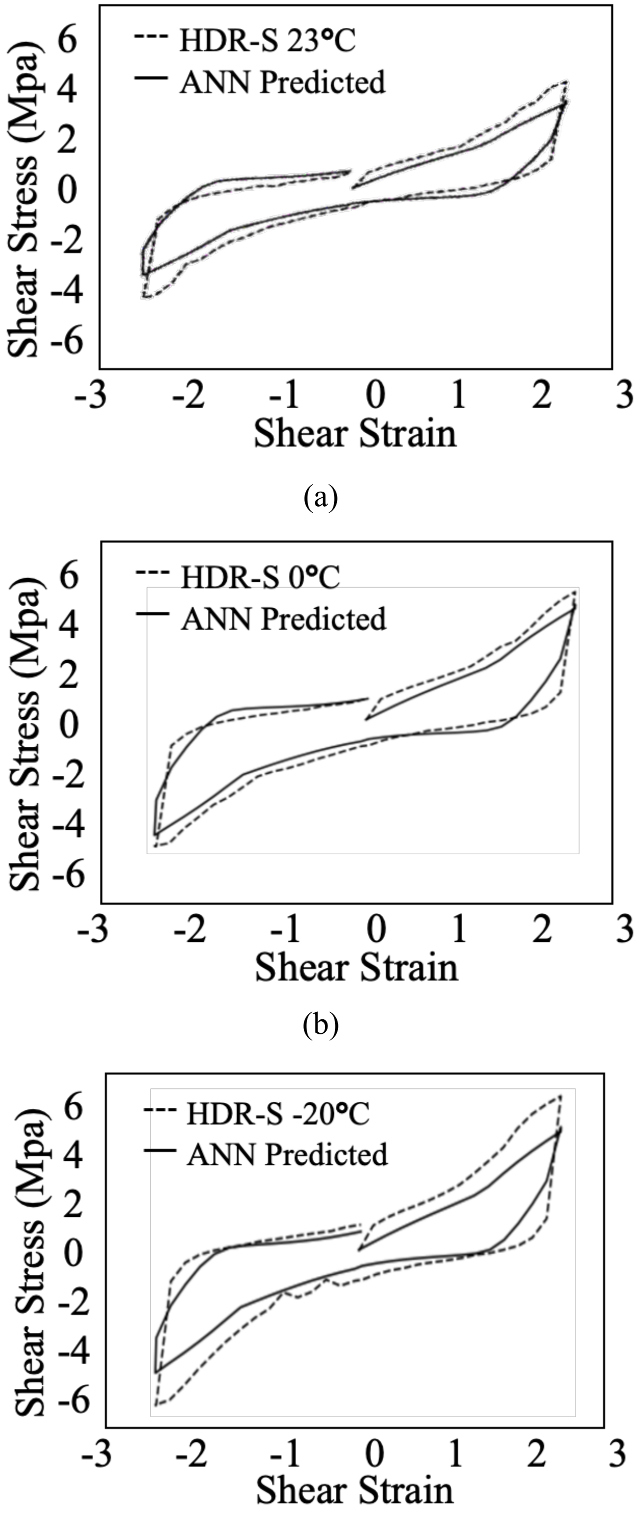


Fig. 28 Modified Bouc-Wen ANN Loss Visualization

The HDR-S quasi-static data at 23°C, 0°C, and -20°C, 250% amplitude, loop 1, were reduced to 60 data points of shear strain and shear stress to fit for the ANN input data for nonlinear parameter identification. The trained ANN model and KH Method predicted nonlinear parameters were plotted and the hysteresis was compared to the HDR-S quasi static loading full data as shown in Figure 30. The contribution rates for HDR-S data at 23°C were 0.97 for ANN and 0.98 for KH Method as shown in Table 9. On the other hand, at -20°C, the contribution rates were 0.91 for ANN and 0.94 for KH Method. There was a 1% up to 3% difference in the contribution rate between ANN and KH Method.

Table 9. Modified Bouc-Wen ANN and KH Method Predicted Parameters Comparison

Temp.	*Method*	α	$G_1(\mathrm{N}/mm^2)$	β	γ	b	R	*Steps*
23°C	ANN	0.33	1.308	4.778	-2.575	0.415	0.97	1
	ANN + KH Method	0.37	0.757	4.352	-3.872	0.421	0.98	4
0°C	ANN	0.33	1.898	5.221	-2.91	0.461	0.95	1
	ANN + KH Method	0.258	1.802	5.783	-3.645	0.331	0.97	3
-20°C	ANN	0.408	1.983	5.672	-3.071	0.532	0.91	1
	ANN + KH Method	0.235	2.613	7.197	-4.412	0.267	0.94	3



(a)

(b)

(c)

Fig. 29 Modified Bouc-Wen Model ANN Predicted Parameter's Visualization at (a) 23 °C (b) 0 °C (c) -20 °C

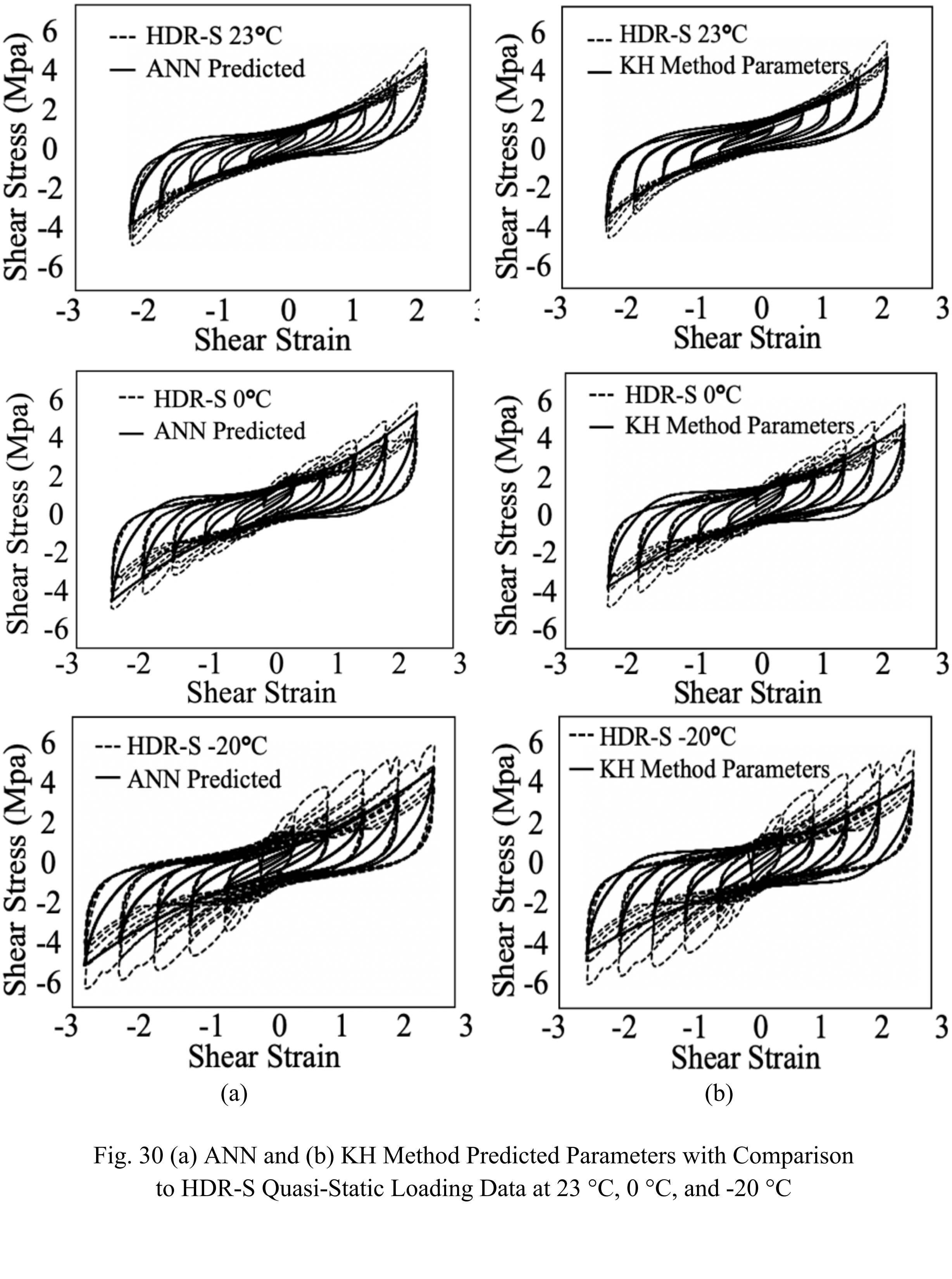


Fig. 30 (a) ANN and (b) KH Method Predicted Parameters with Comparison to HDR-S Quasi-Static Loading Data at 23 °C, 0 °C, and -20 °C

3.3.2 Bilinear Model Nonlinear Parameters ANN Prediction Result

The neural network model training was supervised learning therefore the hidden layers and hyperparameters needs to be carefully specified and will highly affect the training result. This trained ANN model consist of a two-layer neural network with input size of 120, the hidden layers are 100 and 50, and the output size is 3 as shown in Figure 31. The output was 3, which represents the parameters of Bilinear model: initial stiffness, stiffness ratio, and yielding force. Since the AI model focused on regression problem, the activation function used was rectified linear activation function (ReLU)[84], which output's a value from zero to any positive number. Therefore, the output values does not have any limitations on the positive values compare to other activation function like SoftMax which is commonly used for classification problem because it will limit the output values from zero to one.

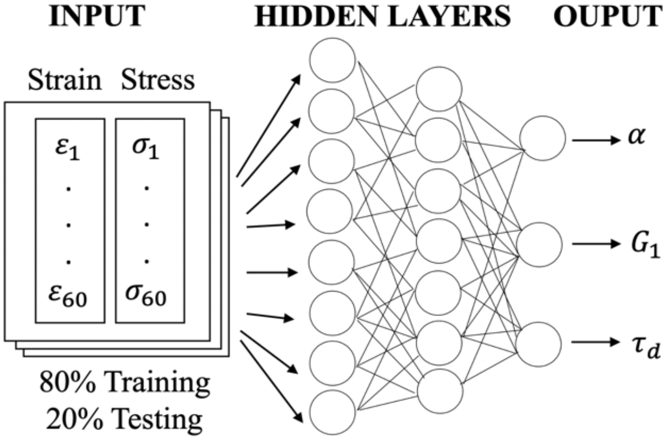


Fig. 31 ANN Model for Bilinear Model Nonlinear Parameter Prediction

The optimizer used was RMSProp[85], which is an optimization technique used to eliminate the vanishing gradient problem. The initial learning rate was set to 0.001. It uses an adaptive learning rate which normalize and balance the momentum step size depending on the value of loss to avoid skipping the optimum gradient. This optimizer was highly recommended for development of neural network which involves regression problem. The neural network continuously updates each parameter weights all throughout the training process until the loss approached to zero.

The optimum loss depends on the human's tolerance and can be visualized during the prediction process. The loss function for training was mean square error (MSE) and mean average error (MAE) for the validation training. The validation was split to 80% training and 20% validation so that even if the model is on the training stage, it was already validated. The total training input was 120 by 1000, so the validation consist of 120 by 200 dataset. The testing data was 120 by 800, and was tested separately after the training.

After 3000 epochs, the testing data mean average error (MAE) was 0.08 and the mean absolute error (MSE) was 0.26. This indicates a good AI model because both the losses was close to zero. The visualization of the loss and epochs was shown in Figure 32. To determine the correlation of the HDR-S quasi-static loading data to the predicted nonlinear parameters by ANN model and KH Method, the contribution rate R was calculated as shown in Table 10.

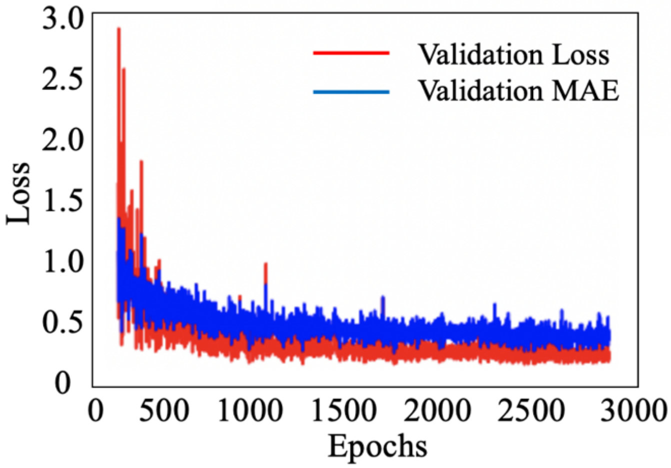
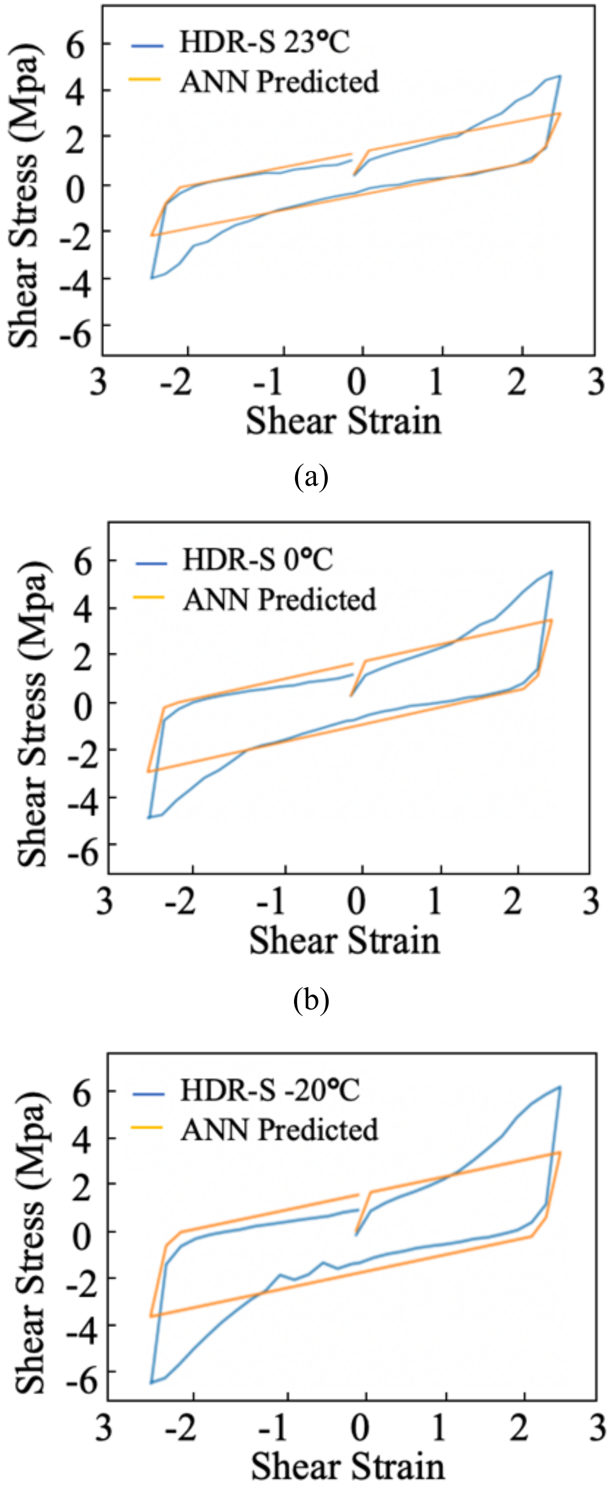


Fig. 32 Bilinear Model ANN Loss Visualization

Using the HDR-S data, at first loop of 250% amplitude as an input to the trained ANN model. Then using the predicted parameters parameters, the hysteresis was plotted and compared to the HDR-S experimental data as shown in Figure 33 and Figure 34. It can be observed that the hardening effect at three temperatures cannot be captured by Bilinear model since it doesn't have any parameters for that.

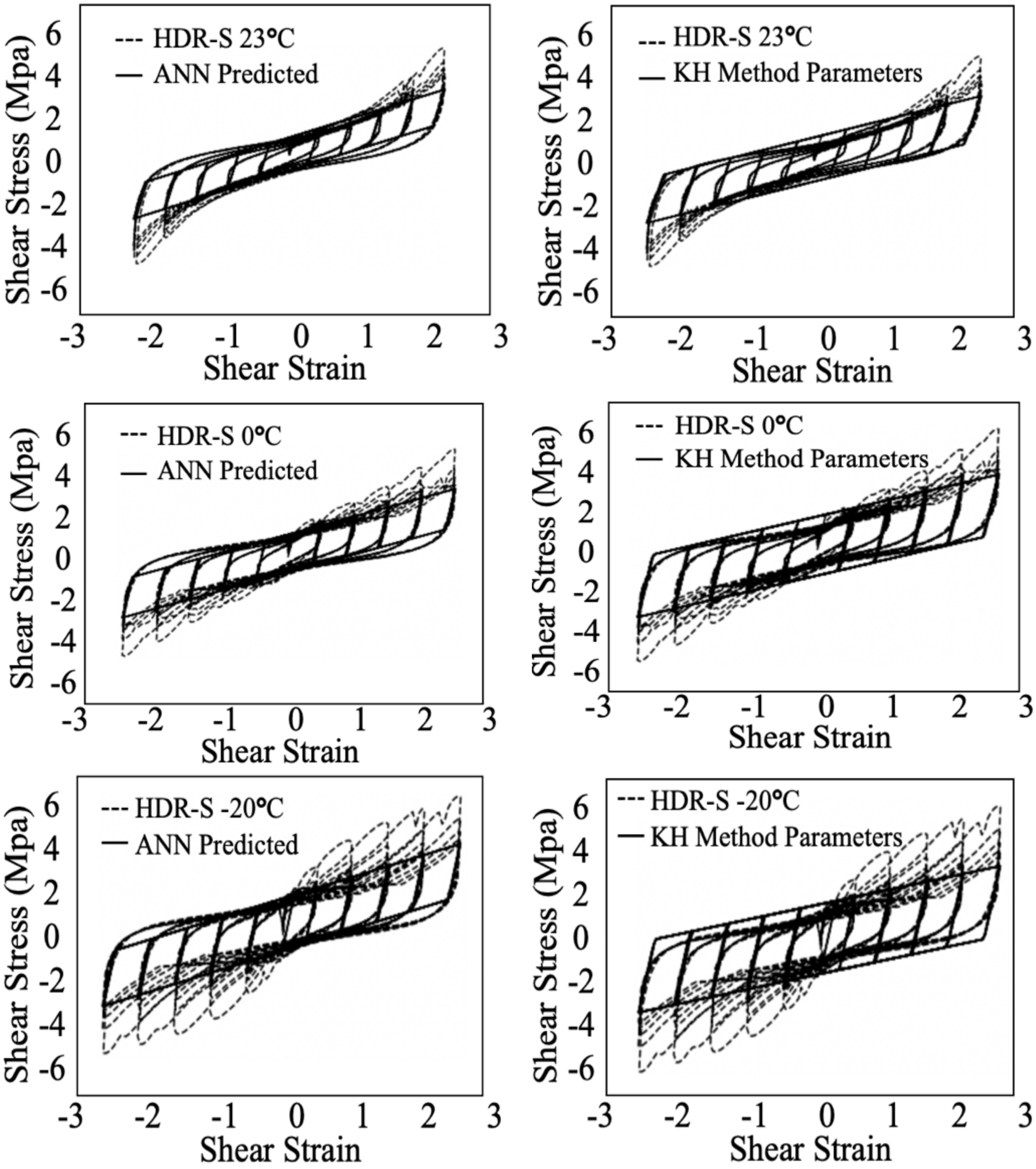


(a)

(b)

(c)

Fig. 33 Bilinear Model ANN Predicted Parameter's Visualization at (a) 23 °C (b) 0 °C (c) -20 °C



(a)

(b)

Fig. 34 (a) ANN and (b) KH Method Predicted Parameters with Comparison to HDR-S Quasi-Static Loading Data at 23 °C, 0 °C, and -20 °C

Table 10. Bilinear Model ANN and KH Method Predicted Parameters Comparison

Temperature	Method	α	$G_1(\frac{N}{mm^2})$	$\tau_d(\frac{N}{mm^2})$	R	Steps
23°C	ANN	0.087	7.097	0.814	0.944	1
	ANN + KH Method	0.097	7.159	0.594	0.966	13
0°C	ANN	0.056	13.423	1.313	0.938	1
	ANN + KH Method	0.062	14.745	0.948	0.97	20
-20°C	ANN	0.046	15.874	1.665	0.921	1
	ANN + KH Method	0.065	16.849	1.286	0.956	20

3.3.3 Clough Model Nonlinear Parameters ANN Prediction Result

The same machine learning approach process was repeated for clough model for nonlinear parameter identification. The ANN model architecture was shown in Figure 35, which has three output parameters. After 1000 epochs, the validation loss was 0.002 as shown in Figure 36.

The predicted nonlinear parameters was then compared to the HDR-S quasi static loading data at 250% amplitude loop 1 and all amplitudes as shown in Figure 37 and Figure 38. The details of the nonlinear parameters was shown in Table 11, including the contribution rate comparison between ANN model and KH Method suggested parameters.

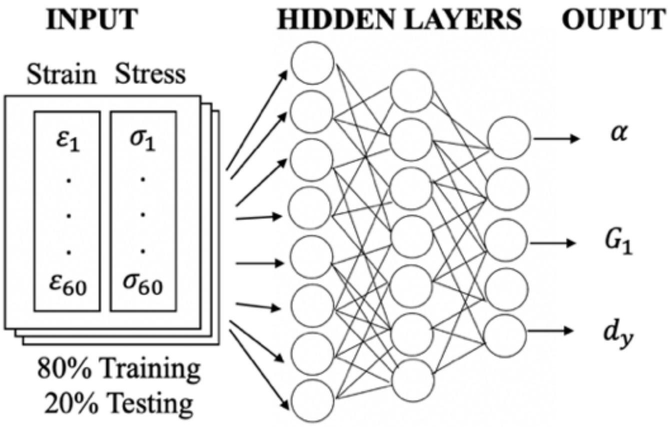


Fig. 35 ANN Model for Clough Model Nonlinear Parameter Prediction

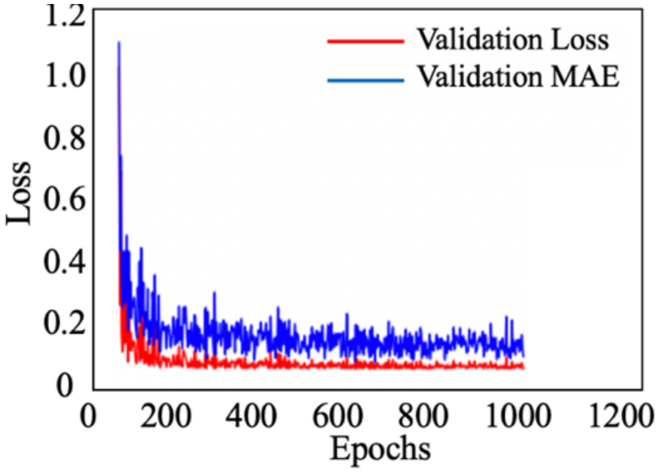
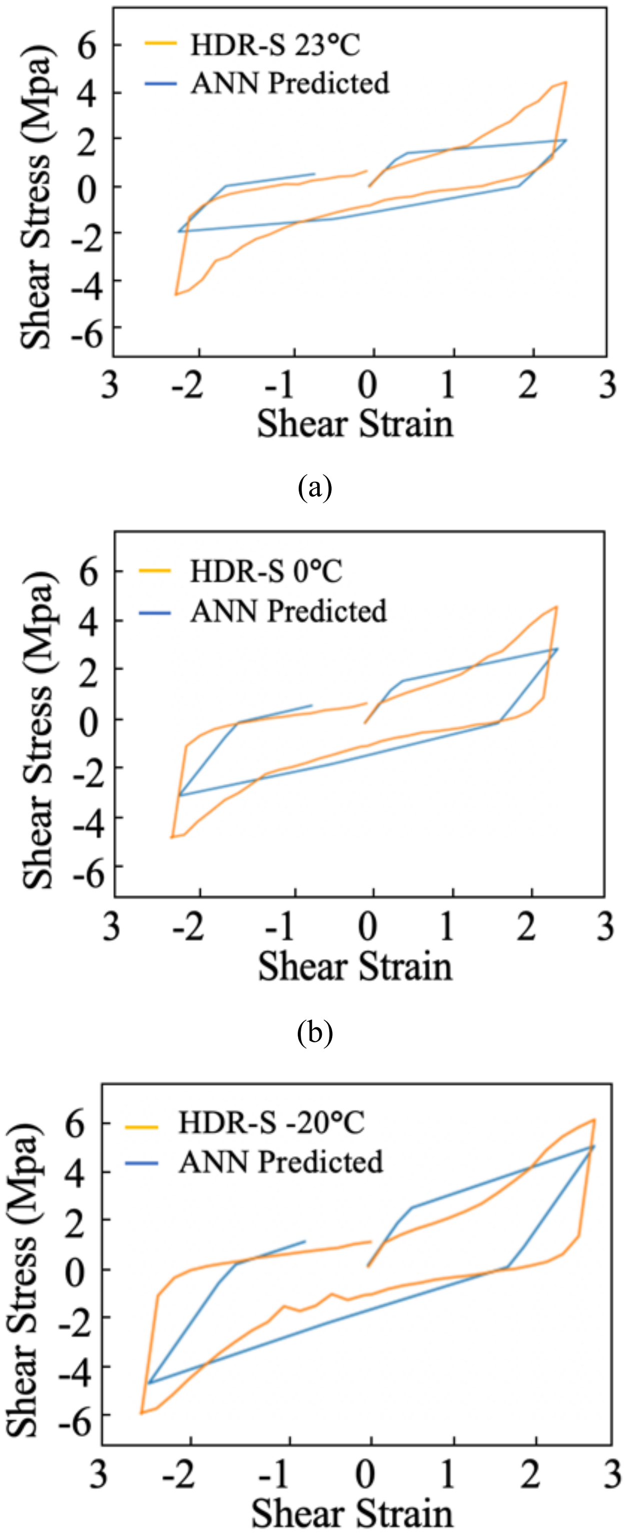


Fig. 36 Clough Model ANN Loss Visualization

Table 11. HDR-S Predicted Clough Model Parameters by ANN and KH Method Comparison

Temperature	Method	α	k	d_y	R
23°C	ANN	0.10	2.85	0.41	0.93
	ANN + KH Method	0.24	3.16	0.41	0.96
0°C	ANN	0.16	4.38	0.42	0.92
	ANN + KH Method	0.14	4.04	0.38	0.94
-20°C	ANN	0.24	5.62	0.43	0.78
	ANN + KH Method	0.21	4.93	0.38	0.86

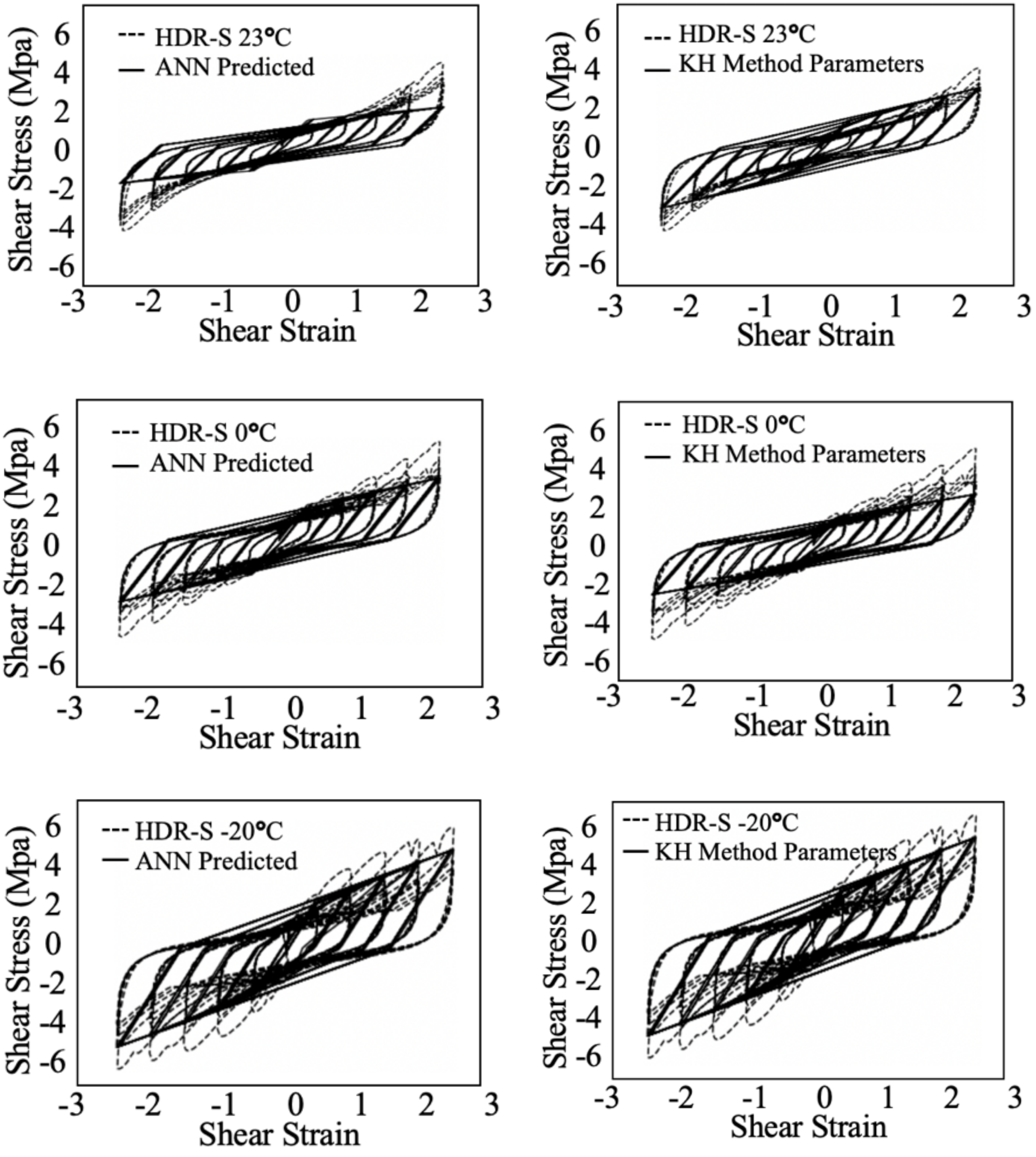


(a)

(b)

(c)

Fig. 37 Clough Model ANN Predicted Parameter’s Visualization at (a) 23 °C (b) 0 °C (c) -20 °C



(a)

(b)

Fig. 38 (a) ANN and (b) KH Method Predicted Parameters with Comparison to HDR-S Quasi-Static Loading Data at 23 °C, 0 °C, and -20 °C

3.3.4 Trilinear Model Nonlinear Parameters ANN Prediction Result

The ANN model for Trilinear model parameter identification has five nonlinear parameters as an output as shown in Figure 39. After 1000 epochs, the validation loss was 0.027 as plotted in Figure 40.

The predicted nonlinear parameters was then compared to the HDR-S quasi static loading data at 250% amplitude loop 1 and all amplitudes as shown in Figure 41 and Figure 42. The details of the nonlinear parameters was shown in Table 12, including the contribution rate comparison between ANN model and KH Method suggested parameters.

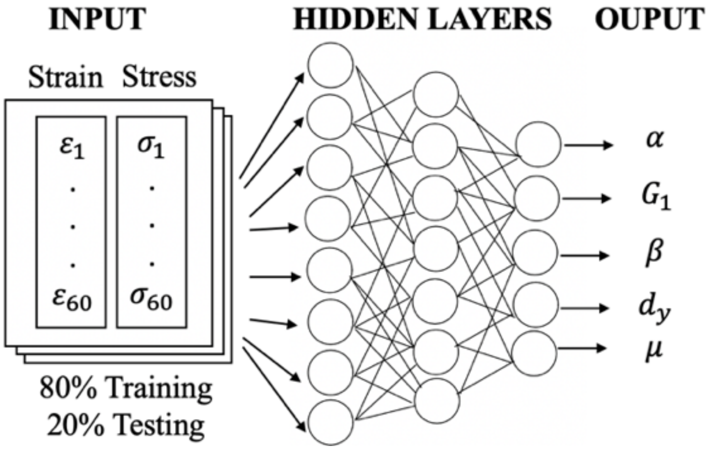


Fig. 39 ANN Model for Trilinear Model Nonlinear Parameter Prediction

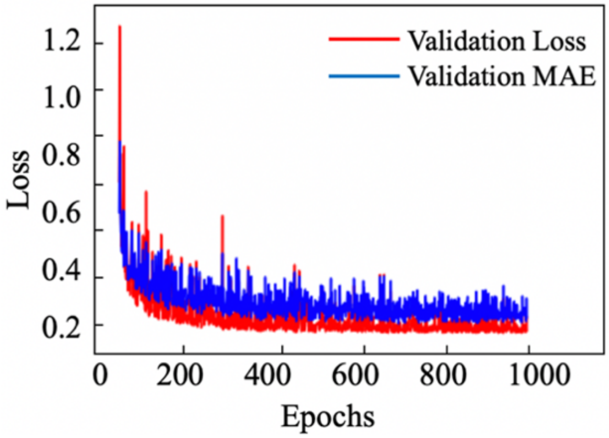
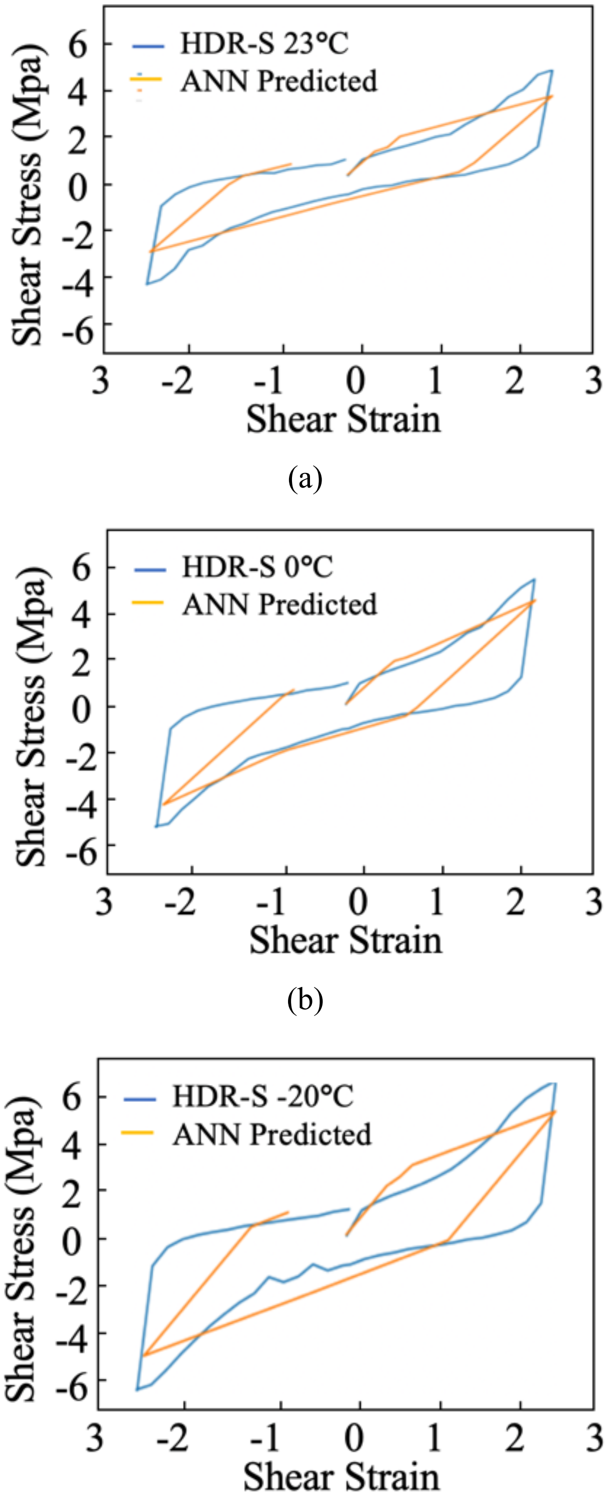


Fig. 40 Trilinear Model ANN Loss Visualization

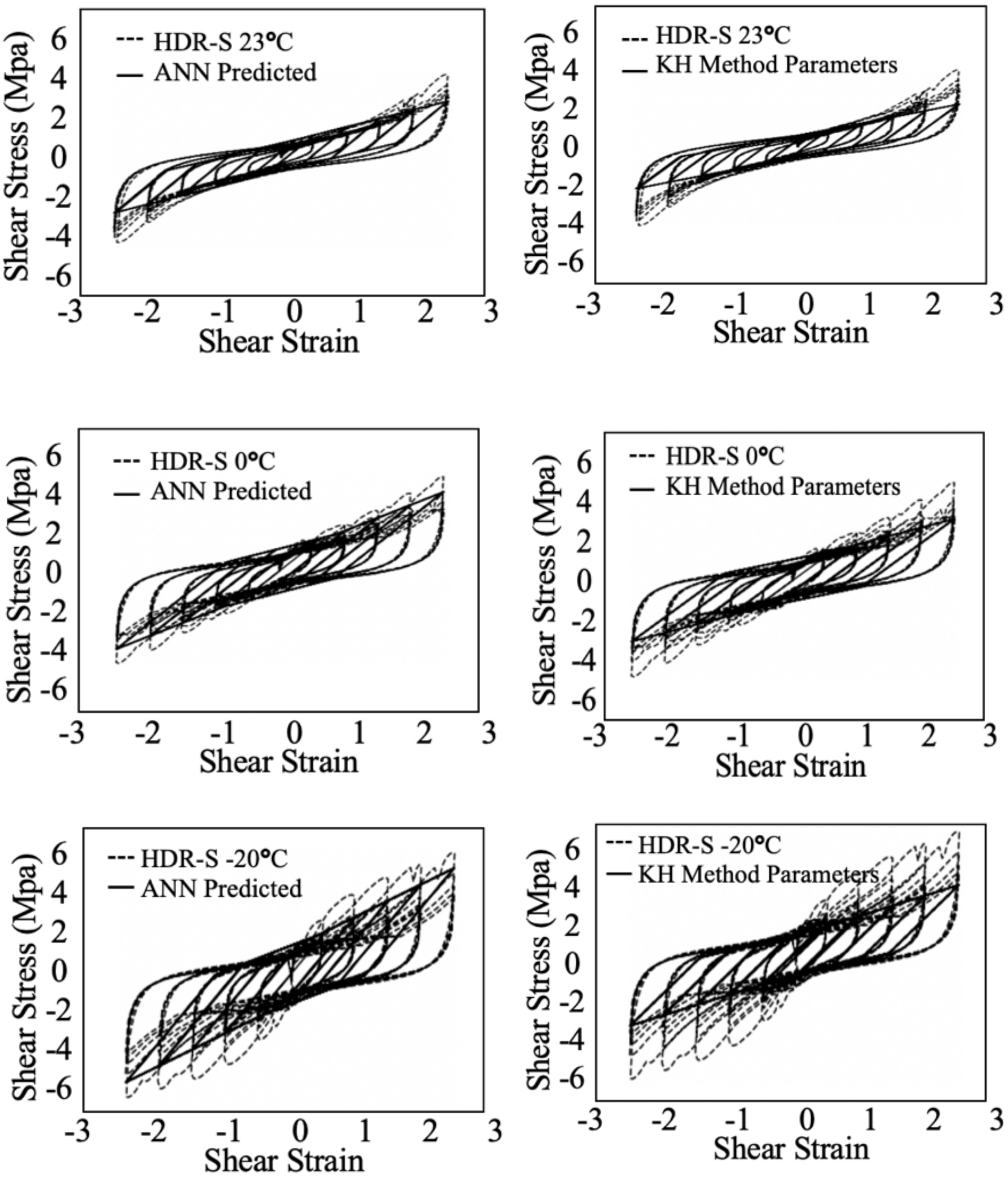


(a)

(b)

(c)

Fig. 41 Trilinear Model ANN Predicted Parameter's Visualization at (a) 23 °C (b) 0 °C (c) -20 °C



(a)

(b)

Fig. 42 (a) ANN and (b) KH Method Predicted Parameters with Comparison to HDR-S Quasi-Static Loading Data at 23 °C, 0 °C, and -20 °C

3.3.5 Comparison of Energy Dissipation and Equivalent Stiffness of HDR-S Data

The trained ANN models and KH Method were used to determine the nonlinear parameters of the HDR-S quasi-static loading data before calculating contribution rates. However, evaluating which model is the best only based on contribution rate is challenging. The area enclosed by the hysteresis curve was plotted using the specified nonlinear parameters for the four different nonlinear models, and the result was used to calculate the energy dissipation, which was then compared to the hysteresis area of the HDR-S quasi-static loading data at 250% amplitude, first loop. Table 13 shows the result of the computed hysteresis area. Additionally, as indicated in Table 14, the equivalent stiffness was estimated and compared with the results of the HDR-S experiment. It can be observed that Bilinear Model (BL) and Modified-Bouc-Wen Model (MBW) have a higher accuracy on the hysteresis area and equivalent stiffness. Therefore, for the hybrid simulation, only these two models were used and compared to the experiment data.

Table 13 . Hysteresis Area using the ANN Predicted Nonlinear Parameters
And HDR-S Quasi-Static Loading Data at 250% Amplitude, 1st Loop Comparison

HDR-S Data	Area	BL	MBW	CL	TR
23 °C	8.17	7.47	7.01	7.05	5.23
0 °C	12.54	12.14	10.50	11.32	7.40
-20 °C	15.61	7.01	12.92	14.86	10.80

Table 14. Equivalent Stiffness using the ANN Predicted Nonlinear Parameters
and the HDR-S Quasi-Static Loading Data at 250% Amplitude, 1st Loop Comparison

HDR-S Data	k_{eq}	BL	MBW	CL	TR
23 °C	1.51	0.95	1.27	0.72	1.04
0 °C	2.10	1.28	1.91	1.34	1.75
-20 °C	2.52	1.41	2.05	2.11	2.24

3.3.6 HDR-S Hybrid Simulation Comparison

Bilinear model (BL) and Modified Bouc-Wen model (MBW) have a higher accuracy in the curve fitting to the HDR-S quasi-static loading data, according to the comparison of contribution rate, energy dissipation, and equivalent stiffness using the plotted hysteresis curve of the ANN suggested nonlinear parameters from each of the four different nonlinear models. This serves as design input parameters for the HDR-S prior to time-history analysis using the ANN proposed parameters of these two nonlinear models. The outcome was next compared with the experimental HDR-S hybrid simulation result.

Given the hypothetical bridge for the hybrid simulation, time-history analysis was carried out using the specified ANN nonlinear parameters under Bilinear Model and Modified Bouc-Wen model (MBW). The hybrid experimental result is compared together with a plot of the hysteretic behavior of the HDR-S bearing installed on the bridge. Figure 43 and Figure 44, compares the HDR-S hysteresis behavior predicted by the hybrid simulation results using the ANN and KH methods under the Bilinear model and Modified Bouc-Wen Model to the real HDR-S hybrid experiment data. Both Nonlinear model parameters were found to suit the HDR-S hybrid simulation experiment well at 23 °C, but poorly at lower temperatures.

The displacement reaction of the pier was also plotted and the experimental outcome is shown in Figure 45. As can be shown, the HDR hybrid simulation hysteresis result and the displacement of the pier at 23 °C are well fit by the suggested nonlinear parameters using the Modified Bouc-Wen model.

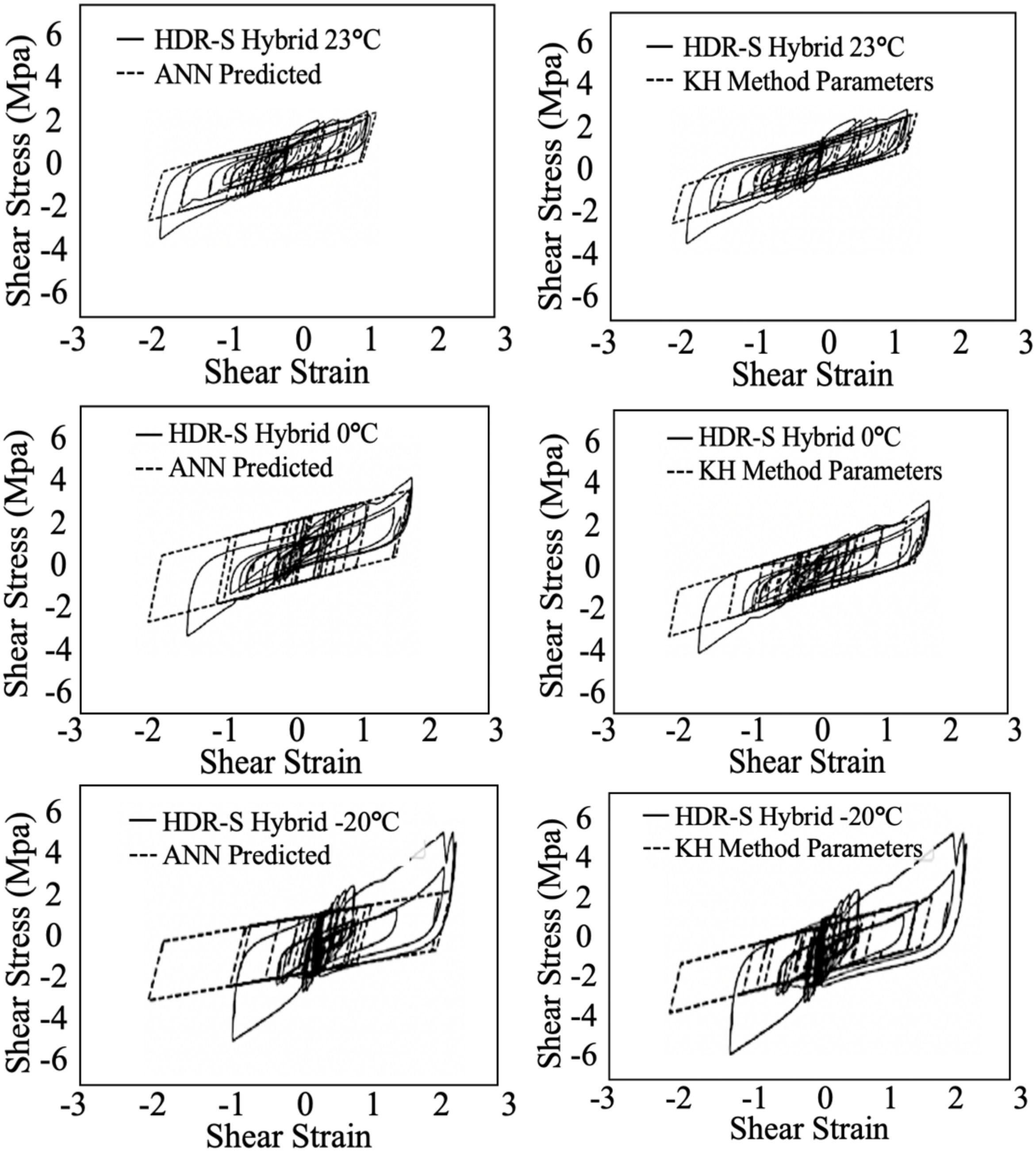
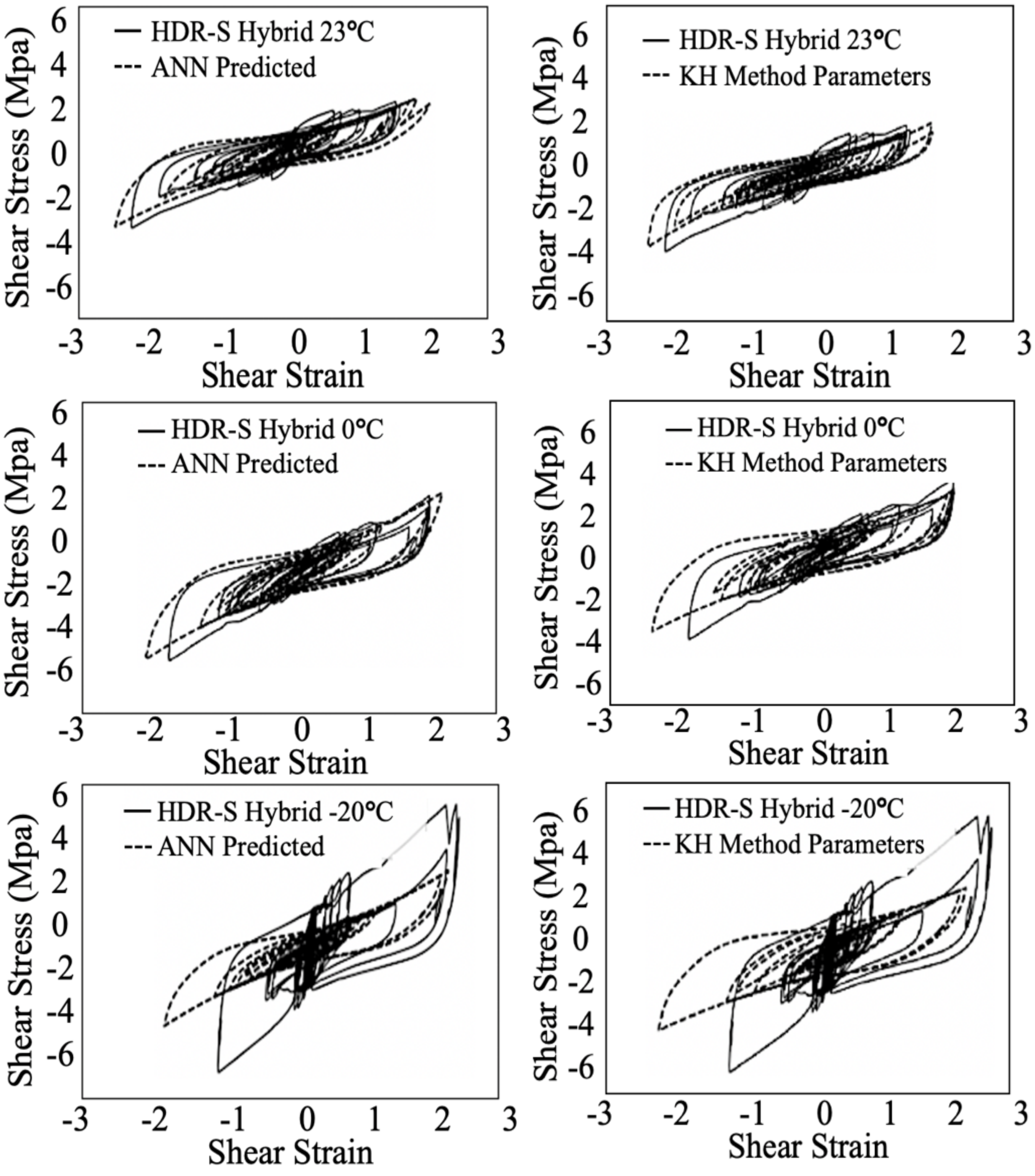


Fig. 43 Hybrid Simulation Comparison Results using Bilinear Model's (a) ANN Predicted Parameters and (b) KH Method Predicted Parameters



(a)

(b)

Fig. 44 Hybrid Simulation Comparison Results using Modified Bouc-Wen's (a) ANN Predicted Parameters and (b) KH Method Predicted Parameters

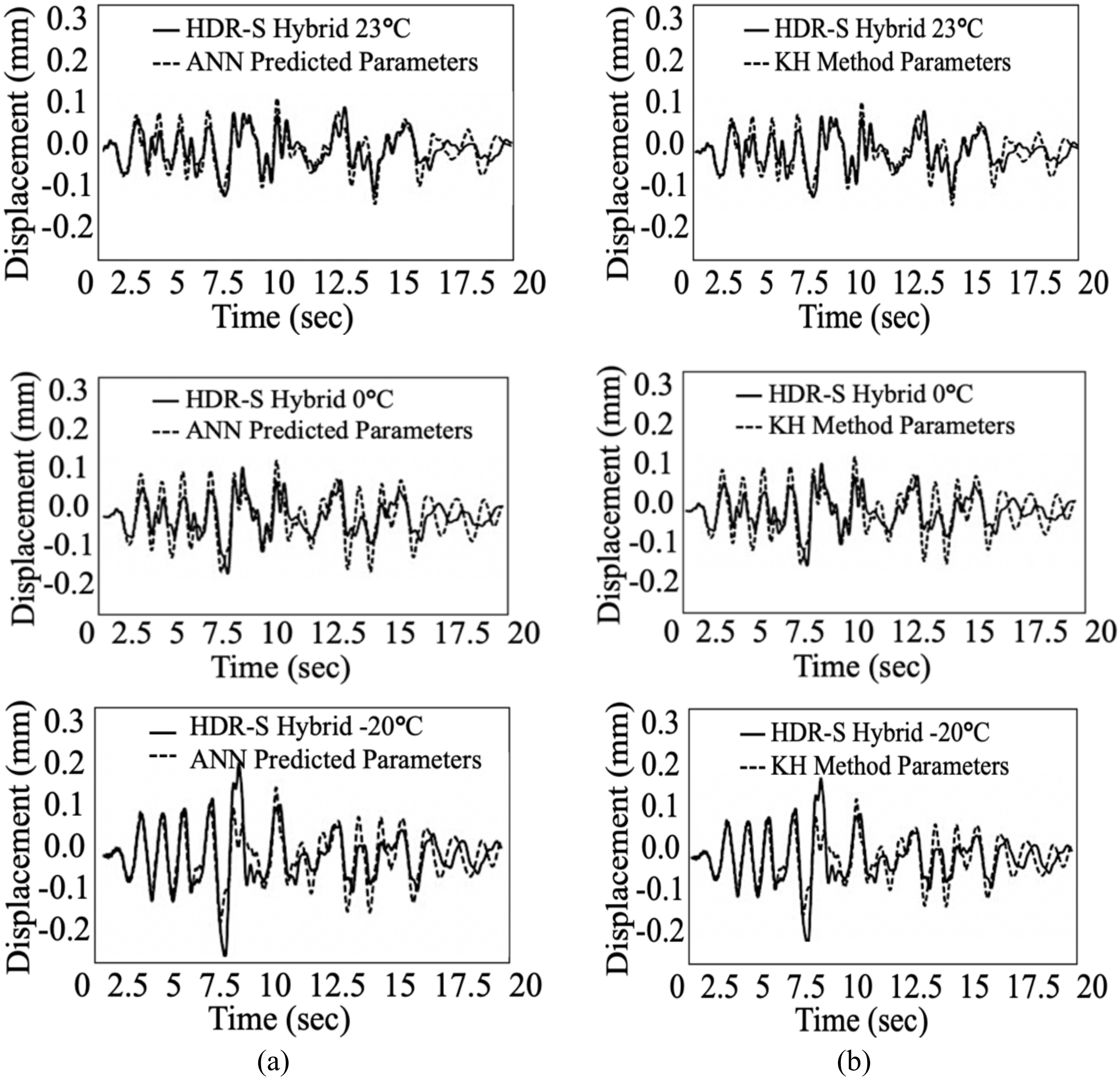


Fig. 45 Hybrid Simulation Pier's Response Comparison Results using (a) ANN Predicted Parameters and (b) KH Method Predicted Parameters under Modified Bouc-Wen Model

Chapter 4: Bridge Seismic Performance Evaluation and Design Suggestion using ANN

This chapter will discuss the data generation and ANN training for seismic performance evaluation and design suggestion. As shown in Figure 46, the bridge isolation design process starts from the initial assumptions of the isolation device parameters, bridge parameters, and other factors. After the consideration of three design earthquakes, the time history analysis will be conducted, and the structure's response will be checked. If the design standards were not satisfied, the initial design parameters will be revised, and the process of time history analysis will be repeated. Time history analysis is the most accurate approach to estimate the bearing's displacement/strain/isolated structure's response however the complexity of numerical models and nonlinearity of bearing's can result to significant computational time [83]. To reduce the trial-and-error processing time, this study trained two ANN model which aims to suggest design parameters and conduct seismic performance evaluation.

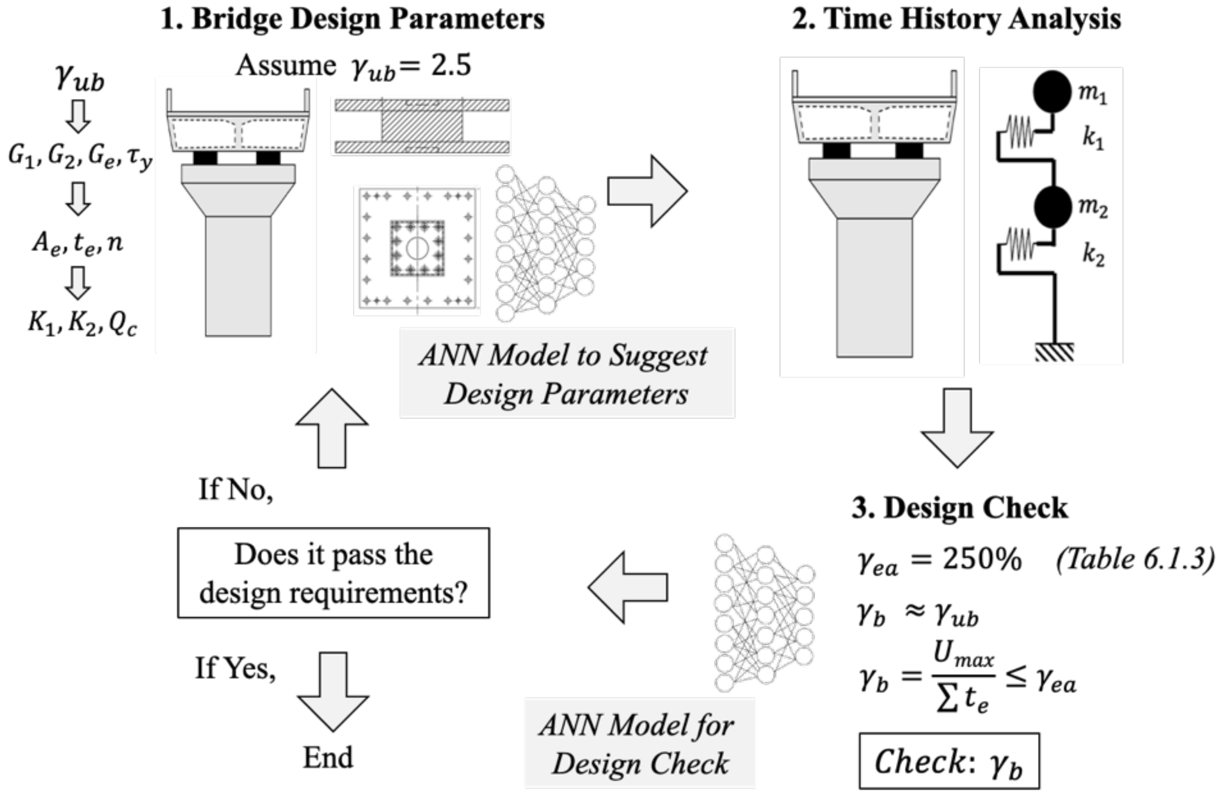

1. Bridge Design Parameters
Assume γ_{ub}= 2.5
γ_{ub}
G_1, G_2, G_e, τ_y
A_e, t_e, n
K_1, K_2, Q_c
ANN Model to Suggest Design Parameters
2. Time History Analysis
m_1
k_1
m_2
k_2
3. Design Check
γ_{ea} = 250% (Table 6.1.3)
$\gamma_b \approx \gamma_{ub}$
$\gamma_b = \frac{U_{max}}{\sum t_e} \leq \gamma_{ea}$
Check: γ_b
ANN Model for Design Check
If No,
Does it pass the design requirements?
If Yes,
End

Fig. 46 Bridge Seismic Design Performance Check Overview

4.1 Bridge Setting

The hypothetical bridge as shown in Figure 47 is a steel bridge with 2 installed HDR-S bearing in each pier. The vertical pressure of each HDR-S ranges from 6 Mpa up to 9 Mpa, given the cross-sectional area of the HDR-S, the mass m_1 of the superstructure can be calculate. Then, the mass of the pier m_2 is 1/3 of the mass of the superstructure. k_1 is the stiffness of the rubber bearing and k_2 is the stiffness of the pier. The stiffness of the pier is assumed to be 4 times than the stiffness of the rubber and it should always be larger for the bearing to experience shear strain.

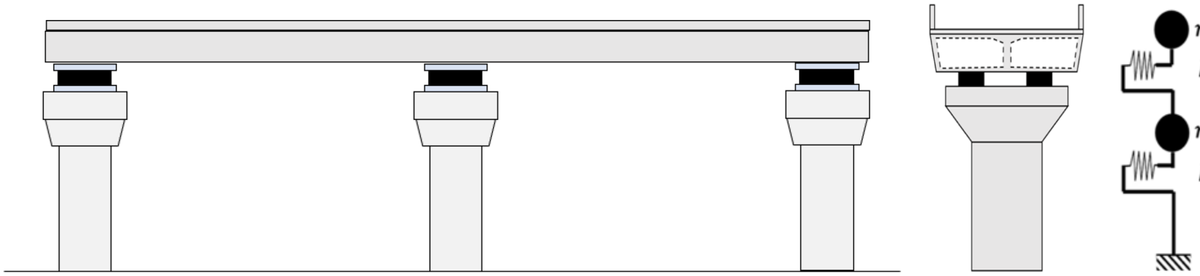


Fig. 47 Bridge 2DOF Model

Considering the 2-Degree-of-Freedom system in the analysis, the equation of motions is shown in equation 38. $\{\ddot{x}\}$ is the acceleration of the structure along the x-axis, $\{\dot{x}\}$ is the velocity of the structure along the x-axis, $\{x\}$ is the displacement of the structure along the x-axis, and $\{\ddot{y}\}$ is the acceleration with the influence of the design input earthquake. The mass matrix is shown in equation 39 which consists of the mass of the superstructure and the mass of the pier. Then the stiffness matrix is shown in equation 40 which consists of the stiffness of the bearing and the pier's stiffness.

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [K]\{x\} = [M]\{\ddot{y}\} \tag{38}$$

$$[M] = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \tag{39}$$

$$[K] = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \tag{40}$$

For the damping $[C]$, equation 41 was used, the parameters a_0 and a_1 are needed in which the calculation can be seen in equation 42 and 43. The w_1 and w_2 are the structures angular velocity, and h_1 and h_2 are equal to 0.05.

$$[C] = a_0[M] + a_1[K] \tag{41}$$

$$a_0 = \frac{2w_1w_2(w_2h_1 - w_1h_2)}{{w_2}^2 - {w_1}^2} \tag{42}$$

$$a_1 = \frac{2(w_2h_2 - w_1h_1)}{{w_2}^2 - {w_1}^2} \tag{43}$$

4.2 HDR-S Nonlinear Parameters

Bilinear model was considered for HDR-S and the nonlinear parameters can be calculated using the following equations based on bridge seismic control design draft. The rubber specification is G12 and the constant values of a_n , b_n, and c_n can be seen in table 15. γ_{uB} is the design shear strain of the HDR-S and the initial assumption is 250%. The shear modulus $G_1(\gamma_{uB})$ and $G_2(\gamma_{uB})$ can be calculated using equation 44 and equation 45. The equivalent shear modulus for design yield strain $G_e(\gamma_{uB})$ can be calculated using equation 46, this is needed to calculate the initial stiffness of the HDR-S as shown in equation 49 which will be use as the nonlinear parameter for bilinear model during time history analysis. $\tau_d(\gamma_{uB})$ is the design shear strain when the design shear strain is zero and $\tau_y(\gamma_{uB})$ is the design shear strain at yielding.

$$G_1(\gamma_{uB}) = a_o + a_1\gamma_{uB} + a_2{\gamma_{uB}}^2 + \ \dots + a_i{\gamma_{uB}}^i \tag{44}$$

$$G_2(\gamma_{uB}) = b_o + b_1\gamma_{uB} + b_2{\gamma_{uB}}^2 + \ \dots + b_i{\gamma_{uB}}^i \tag{45}$$

$$G_e(\gamma_{uB}) = c_o + c_1\gamma_{uB} + c_2{\gamma_{uB}}^2 + \ \dots + c_i{\gamma_{uB}}^i \tag{46}$$

$$\tau_d(\gamma_{uB}) = \gamma_{uB}(G_e(\gamma_{uB}) - G_2(\gamma_{uB})) \tag{47}$$

$$\tau_y(\gamma_{uB}) = \tau_d(\gamma_{uB}) \frac{G_1(\gamma_{uB})}{G_1(\gamma_{uB}) - G_2(\gamma_{uB})} \tag{48}$$

Table 15. Constant Values of HDR-S G12 for Bilinear Nonlinear Parameter Computation

HDR-S G12	a_o	a_1	a_2	a_3	a_4	a_5
	35.13	-54.83	43.51	-16.44	3.019	-0.2114
	b_o	b_1	b_2	b_3	b_4	b_5
	3.345	-5.221	4.144	-1.565	0.2872	-0.0201
	c_o	c_1	c_2	c_3	c_4	c_5
	5.128	-7.971	6.227	-2.331	0.4162	-0.0276

$$k_1 = \frac{G_1(\gamma_{uB})A_e}{\Sigma t_e} \tag{49}$$

$$k_2 = \frac{G_2(\gamma_{uB})A_e}{\Sigma t_e} \tag{50}$$

$$k_B = \frac{G_e(\gamma_{uB})A_e}{\Sigma t_e} \tag{51}$$

$$Q_d = \tau_d(A_e) \tag{52}$$

$$Q_y = \tau_y(A_e) \tag{53}$$

Aside from the nonlinear parameter of the HDR-S bearing, the primary and secondary shape coefficient should be checked. Figure 48 shows the cross section and rubber thickness of the bearings, The first shape factor S_1 should be larger than 6, and the second shape factor S_2 should be larger than 4 as shown in equation 54 and equation 55. Based on the assumed cross-sectional area and rubber thickness, the dataset distribution is shown in Figure 49 and Figure 50.

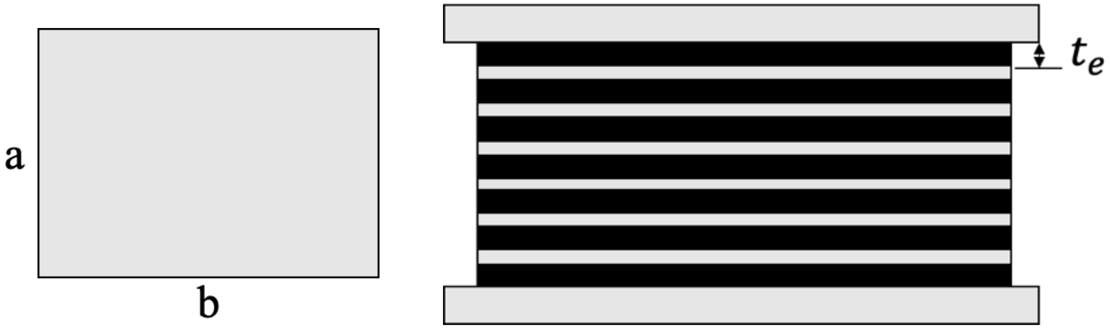
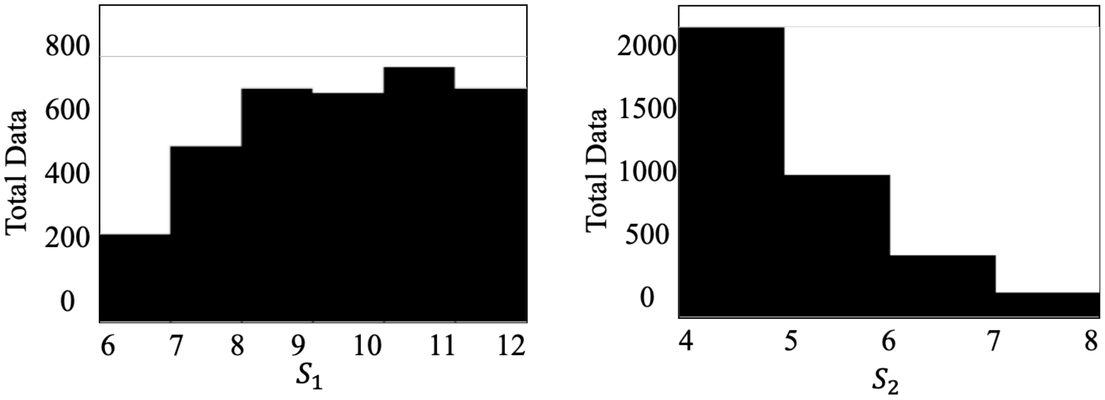


Fig. 48 HDR-S Cross Section and Rubber Thickness

$$S_1 = \frac{A_e}{2(a+b)t_e},\ S_1 > 6.0 \tag{54}$$

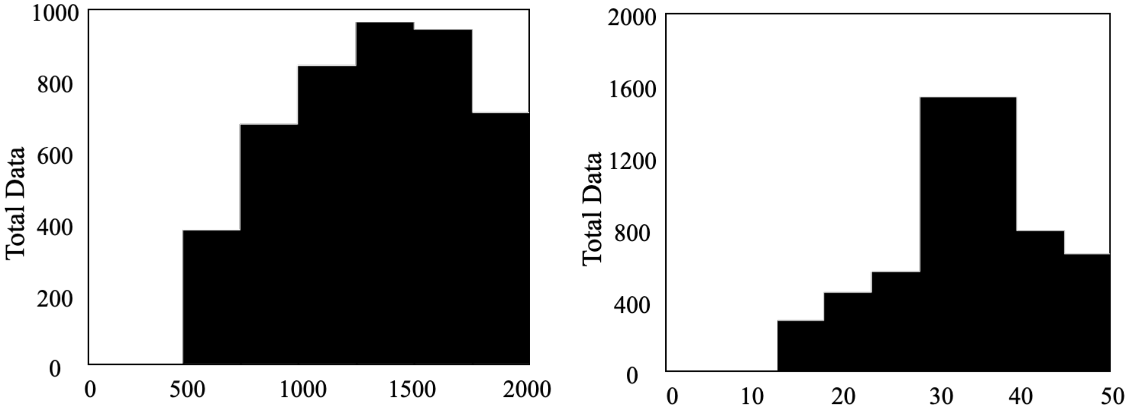
$$S_2 = \frac{a\ or\ b}{\Sigma t_e}\ \ ,\ S_2 > 4.0 \tag{55}$$



(a) (b)

Fig. 49 Shape Factor Data Distribution

The range of the HDR-S bearings cross section width is 400mm to 1500 mm, the thickness of the rubber ranges from 15 mm to 40 mm, and the number of rubber layers are 6 pcs to 11 pcs.

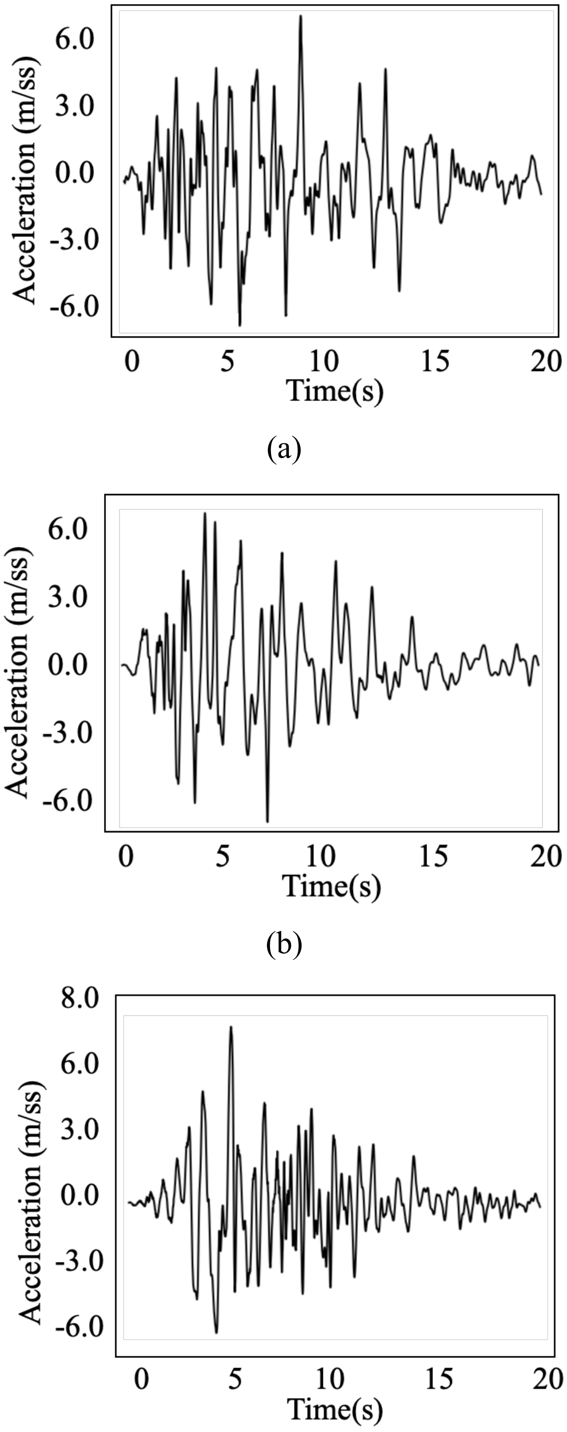


(a) (b)

Fig. 50 HDR-S Cross Section in mm (a) and Rubber Thickness Parameter's Data Distribution

4.3 Input Earthquake

The three design earthquakes under level 2 ground type II are shown in figure 51 with the standard acceleration response spectrum [87] as shown in table 16. The three-earthquake data were gathered from the 1995 Kobe earthquake [30] and the specific locations are North-south of JR Takatori station, JR West Takatori station inland board East-West, and North 27 degrees West of Osaka gas supply station.



(a)

(b)

(c)

Fig. 51 Design Earthquakes under Level 2 G2 Category (a) (b) (c)

Table.16 Standard Acceleration Response Spectrum $S_{II\,0}$ for Type II Ground Motion

Ground Type	Structure's Period T(s) for Response Spectrum $S_{II\,0}$		
I	T< 0.3 $S_{II\,0} = 4463\,T^{2/3}$	$0.3 \leq T \leq 0.7$ $S_{II\,0} = 2000$	0.7 < T $S_{II\,0} = 1104\,T^{5/3}$
II	T< 0.4 $S_{II\,0} = 3224\,T^{2/3}$	$0.4 \leq T \leq 1.2$ $S_{II\,0} = 1750$	1.2 < T $S_{II\,0} = 2371\,T^{5/3}$
III	T< 0.5 $S_{II\,0} = 2381\,T^{2/3}$	$0.5 \leq T \leq 1.5$ $S_{II\,0} = 1500$	1.5 < T $S_{II\,0} = 2948\,T^{5/3}$

4.4 HDR-S Maximum Shear Strain Classification

After creating the large dataset with different combinations of design parameter's, initial structure conditions, and response checked data, the ANN model was trained to classify the maximum shear strain if it passes the performance requirements or not. As shown in figure 52, the input parameters are mass of the superstructure, mass of the pier, stiffness of the bearing, stiffness of the pier, HDR-S cross section width, rubber thickness, total number of rubbers, and the vertical pressure. The main objective is to classify whether the output γ_b is greater than 250% or less. This is important to not repeat the time history analysis given the initial design combinations. There are 600 total training datasets when γ_b is greater than 250% and 700 datasets when γ_b is less than 250%.

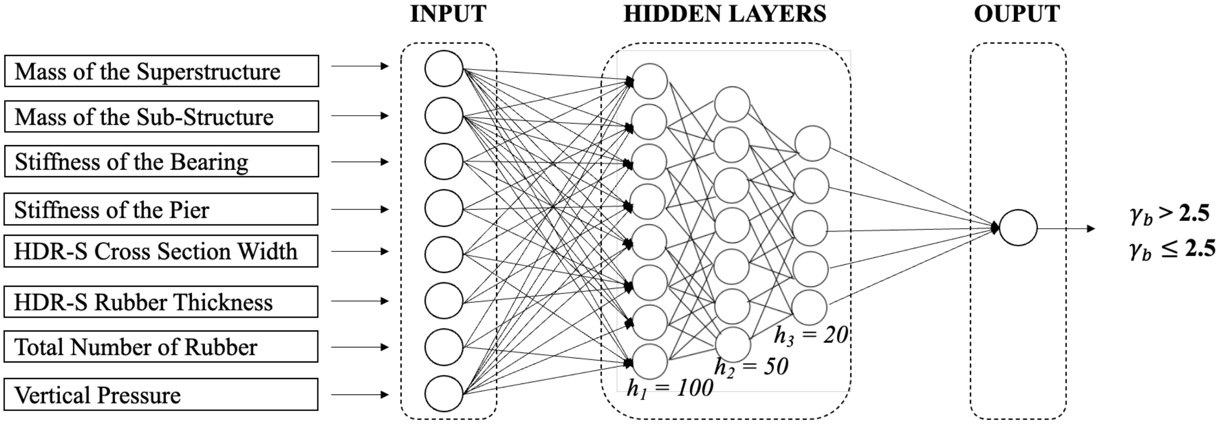


Fig. 52 Input-Output Relationship of ANN Model for Maximum Shear Strain Classification

As shown in Table 17, the ANN model consists of 3 hidden layers, the optimizer is RMSProp, the initial learning rate is 0.001, the activation functions are ReLU in the hidden layers and Softmax at the output layer because the aim is classification.

Table 17. ANN Model Parameters for Maximum Shear Strain Classification

3 Hidden Layers	100,50,20
Optimizer	Root Mean Squared Propagation (RMSProp)
Initial Learning Rate	0.001
Activation Function	ReLU and Softmax
Epochs	100

After 100 epochs, the validation loss is 0.02 and the validation accuracy is 0.99 as visualized in Figure 53. The confusion matrix in Figure 54 shows the number of true positives, true negatives, false positives, and false negatives. 100% were classified as γ_b is greater than 250%, and 497 out of 500 were classified as γ_b is less than 250%.

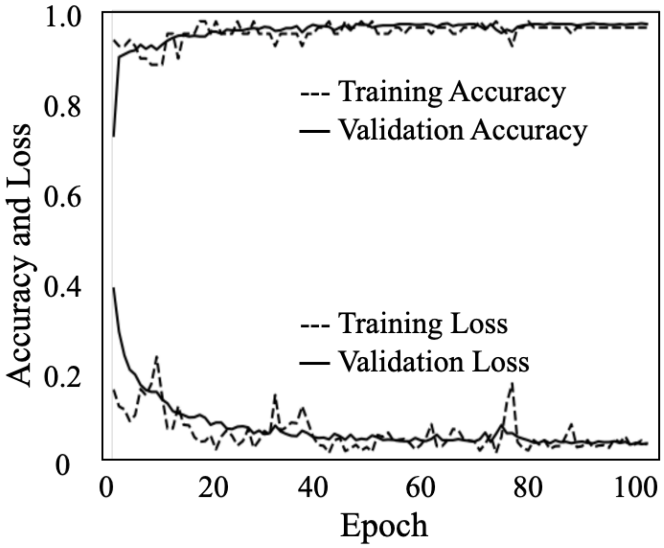


Fig. 53 Accuracy and Loss of ANN Model for Maximum Shear Strain Classification

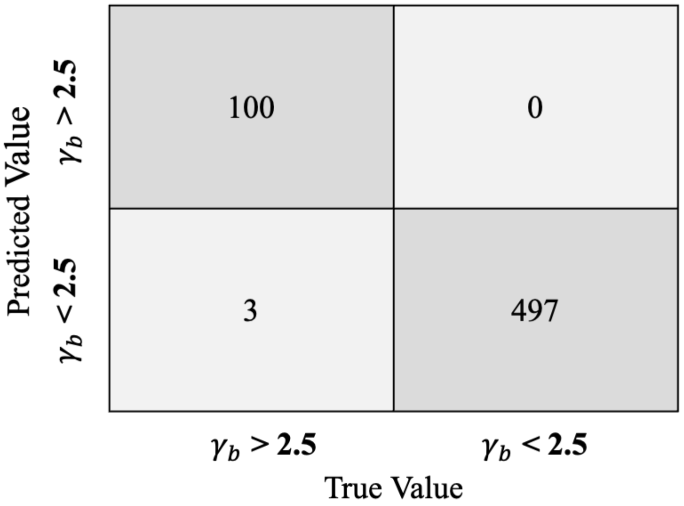


Fig. 54 Confusion Matrix of ANN Maximum Shear Strain Classification

4.5 ANN for HDR-S Maximum Shear Strain Prediction

Aside from classification of the HDR-S shear strain, the next ANN model was trained to predict the response HDR-S shear strain γ_b. The same input parameters were used but the output will be the specific value of γ_b as shown in Figure 55.

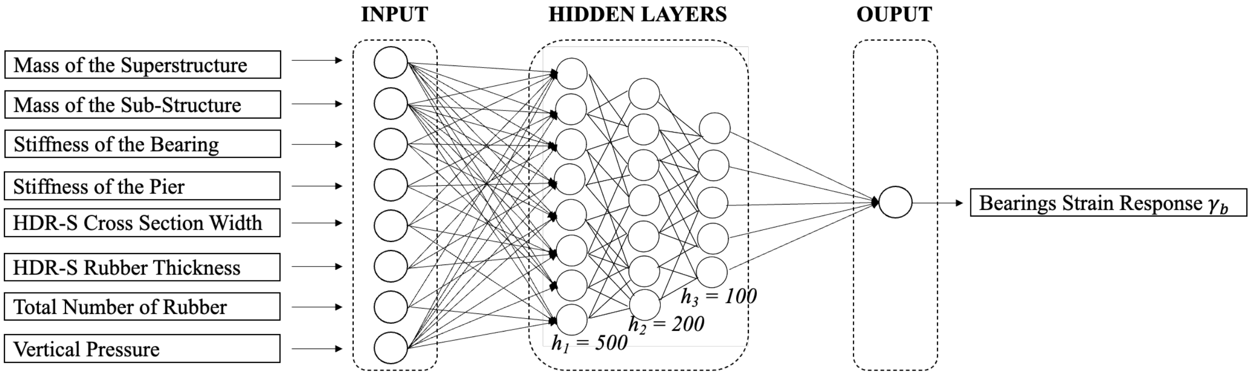


Fig. 55 Input-Output Relationship of ANN Model for Maximum Shear Strain Prediction

As shown in Table 18, the ANN model consists of 3 hidden layers, the optimizer is RMSProp, the initial learning rate is 0.001, the activation function is ReLU because the output is an integer. The total dataset for training and testing were also shown.

Table 18. ANN Model Parameters for Maximum Shear Strain Prediction

3 Hidden Layers	500, 200, 100
Optimizer	Root Mean Squared Propagation (RMSProp)
Activation Function	Rectified Linear Unit (ReLU)
Epochs	100
Input Data	Training: [8, 3500], Testing: [8, 1000]
Output Data	Training: [1, 3500], Testing: [1, 1000]

After 100 epochs, the validation loss was 0.04 as visualized in Figure 56. Then the training and ANN predicted bearing shear were plotted in Figure 57. The coefficient of determination is 0.996 which means that the two value are in good fit.

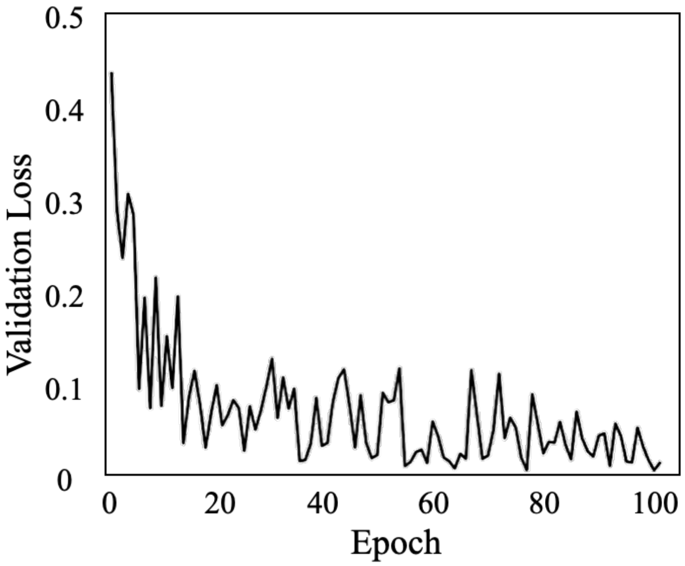


Fig. 56 Validation Loss of ANN Model for Maximum Shear Strain Prediction

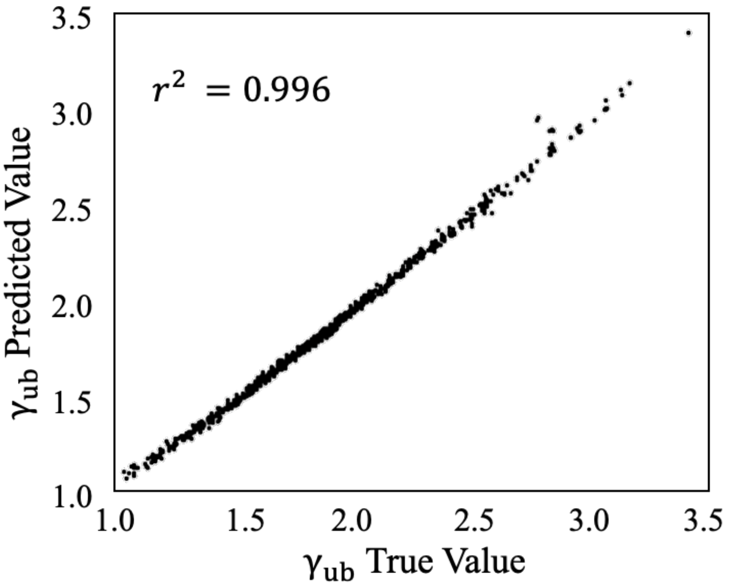


Fig. 57 HDR-S Maximum Shear Strain Prediction and True Value Visualization

4.7 ANN HDR-S Design Parameter's Suggestion

This ANN model aims to suggest design parameter's specifically the HDR-S cross section width and the total rubber thickness. The input parameters are mass of the superstructure, mass of the sub-structure, stiffness of the bearing, stiffness of the pier, and the vertical pressure as shown in Figure 58. Out of the 4500 datasets, this study only considered the dataset wherein the bearings response strain is between 200% to 250%. These data are considered as economical and passed the design requirements. The total dataset was 1000 and was separated into training and testing as shown in table 19.

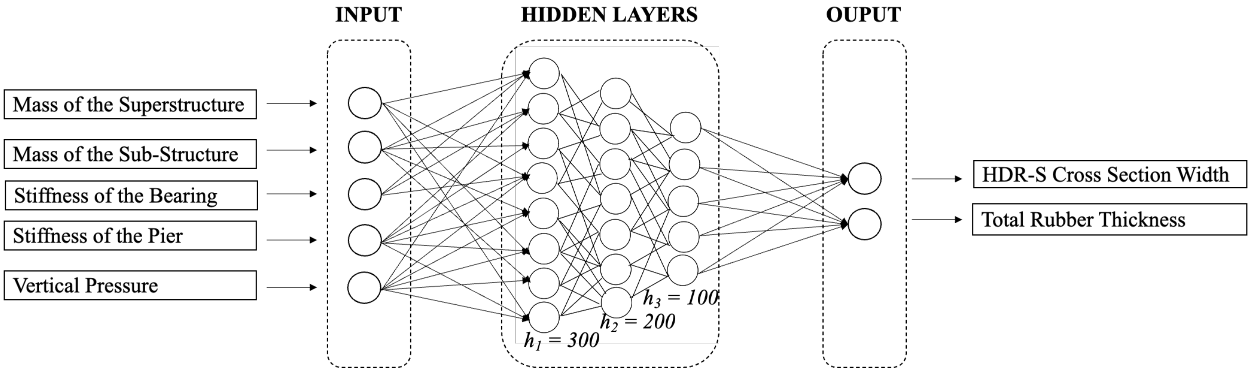


Fig. 58 Input-Output Relationship of ANN Model for Design Parameter Suggestion

Table 19. Training Dataset for Design Parameter's Suggestion

Input Data	Training: [5, 700], Testing: [5, 300]
Output Data	Training: [2, 700], Testing: [2, 300]
Total Dataset	1000
γ_{ub}**>2.5**	300 Dataset
γ_{ub} **<2.5**	700 Dataset

As shown in Table 20, the ANN model consists of 3 hidden layers, the optimizer is RMSProp, the initial learning rate is 0.001, and the activation function is ReLU. The total dataset for training and testing were also shown.

Table 20. ANN Parameter Setting for Design Parameter's Suggestion

4 Hidden Layers	300, 200, 100
Optimizer	Root Mean Squared Propagation (RMSProp)
Initial Learning Rate	0.001
Activation Function	Rectified Linear Unit (ReLU)
Epochs	100

After 100 epochs, the validation loss 0.01 as visualized in Figure 59, the ANN model can suggest HDR-S cross section width and total rubber thickness with high accuracy based on the five input parameters. However, cost analysis and most economical section were not considered in this study.

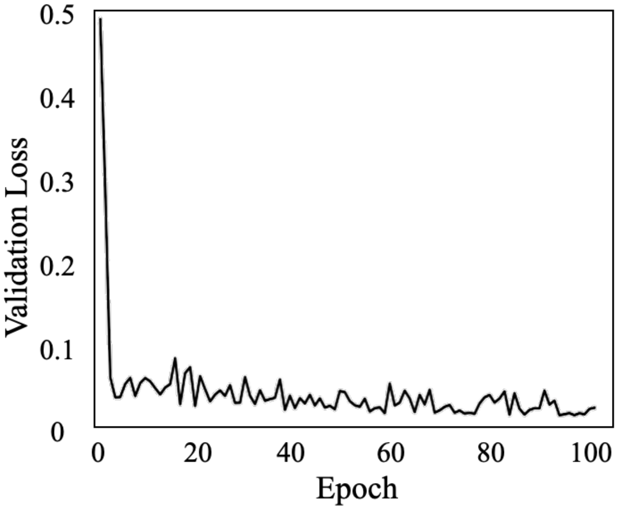


Fig. 59 Validation Loss of ANN Model for Design Parameter Suggestion

The ANN suggested HDR-S cross section width and total rubber thickness were compared and plotted with the simulated data as shown in Figure 60 and Figure 61. The coefficient of determination are both 0.99 which means that the two data are in good fit.

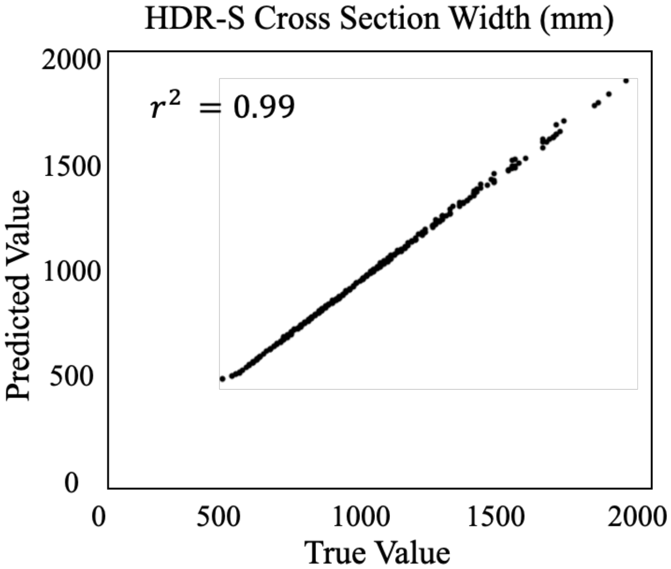


Fig. 60 HDR-S Cross Section Width ANN Predicted and True Value Visualization

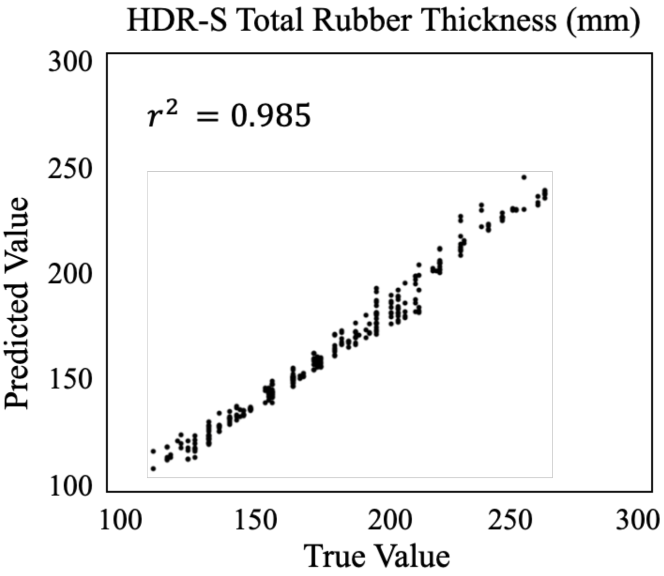


Fig. 61 HDR-S Total Rubber Thickness ANN Predicted and True Value Visualization

Chapter 5: Enhanced Bridge Visual Inspection and 3D Model Reconstruction

Bridge vision-based structural health monitoring is a crucial component of proactive early damage diagnosis. Due to the volume of data, it is challenging for the periodic bridge inspection to monitor the progress of maintenance and deterioration. Additionally, the conventional inspection process demands a significant investment in labor, equipment tools, and time. In order to facilitate maintenance monitoring progress and track deterioration over time, a digitalized and advanced approach to managing bridge data was proposed in this research. This approach uses deep learning, SfM, and MR platforms to store the generated bridge 3D model data with the segmented damages. The bridge time capsule idea as shown on Figure 62, describes the comparison of the BIM bridge model, the 3D model without segmented damages, and the 3D bridge model with segmented damages during periodic maintenance. The proposed approach can help the engineers to further visualize the actual damage prognosis in the whole bridge.

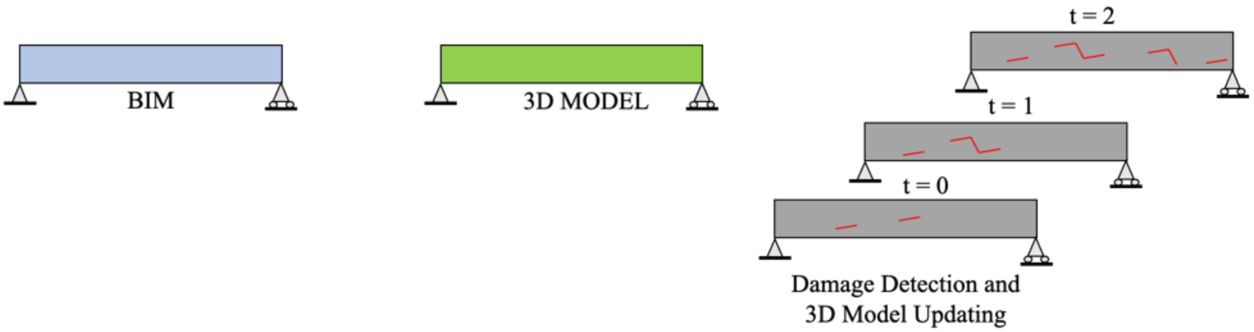
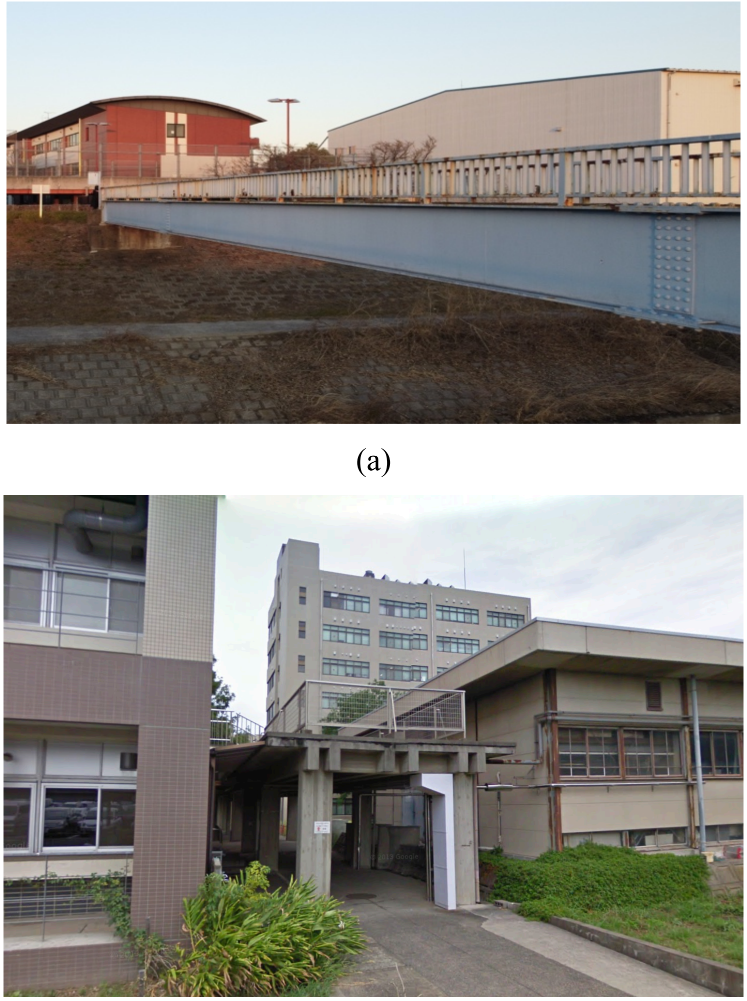


Fig. 62 Bridge 3D Model Damage Segmentation Proposed Concept

By fusing many technologies—specifically the use of drones, deep learning, structure from motion, and mixed reality—this study presented an enhanced bridge visual inspection procedure. The study's overall framework, which aims to significantly improve and revolutionize the conventional method of bridge inspection, included the integration of the four different methods. This study deployed UAVs to quickly and frequently patrol bridges while optimizing the flight path. This helped to speed up the process, cut costs, address the issue of a shortage of skilled labor, and make it paperless and digital. This may reduce the number of people required for the inspection and encourages a safer visual inspection technique, particularly for difficult-to-access areas of the bridge. The suggested method will allow engineers to observe and work together in a virtual environment where danger of site exposure was eliminated.

5.1 Target Bridges

The first bridge, as shown in Fig. 63-a, is a two-span steel pedestrian bridge located in Japan. This bridge will be used for the 3D model generation with and without the segmented corrosion damages. The second bridge as shown in Figure 63-b, is a concrete pedestrian bridge in Japan, which was used to gather RGBD data for 3-dimensional structure element point cloud data generation that will be discuss in Chapter 6.



(a)

(b)

Fig. 63 Target Bridge (a) Steel Bridge (b) Concrete Pedestrian Bridge

5.2 Data Gathering using Unmanned Aerial Vehicle (UAV)

The photographs were taken with a DJI Mavic mini, as seen in Figure 64 and Figure 65, and the ideal flight route was planned as shown on Figure 66 and Figure 67. Careful consideration was given to variations in the vertical distance, horizontal distance, and angle of view from the bridge. A more realistic 3D model can be created with SfM if there is a 70% image overlap. The sample photographs of the acquired bridge, which were among the 950 total images. Lightning conditions, weather, particularly the wind velocity, and UAV flying duration restrictions are few of the challenges experienced.

Fig. 64 Unmanned Aerial Vehicle (UAV)

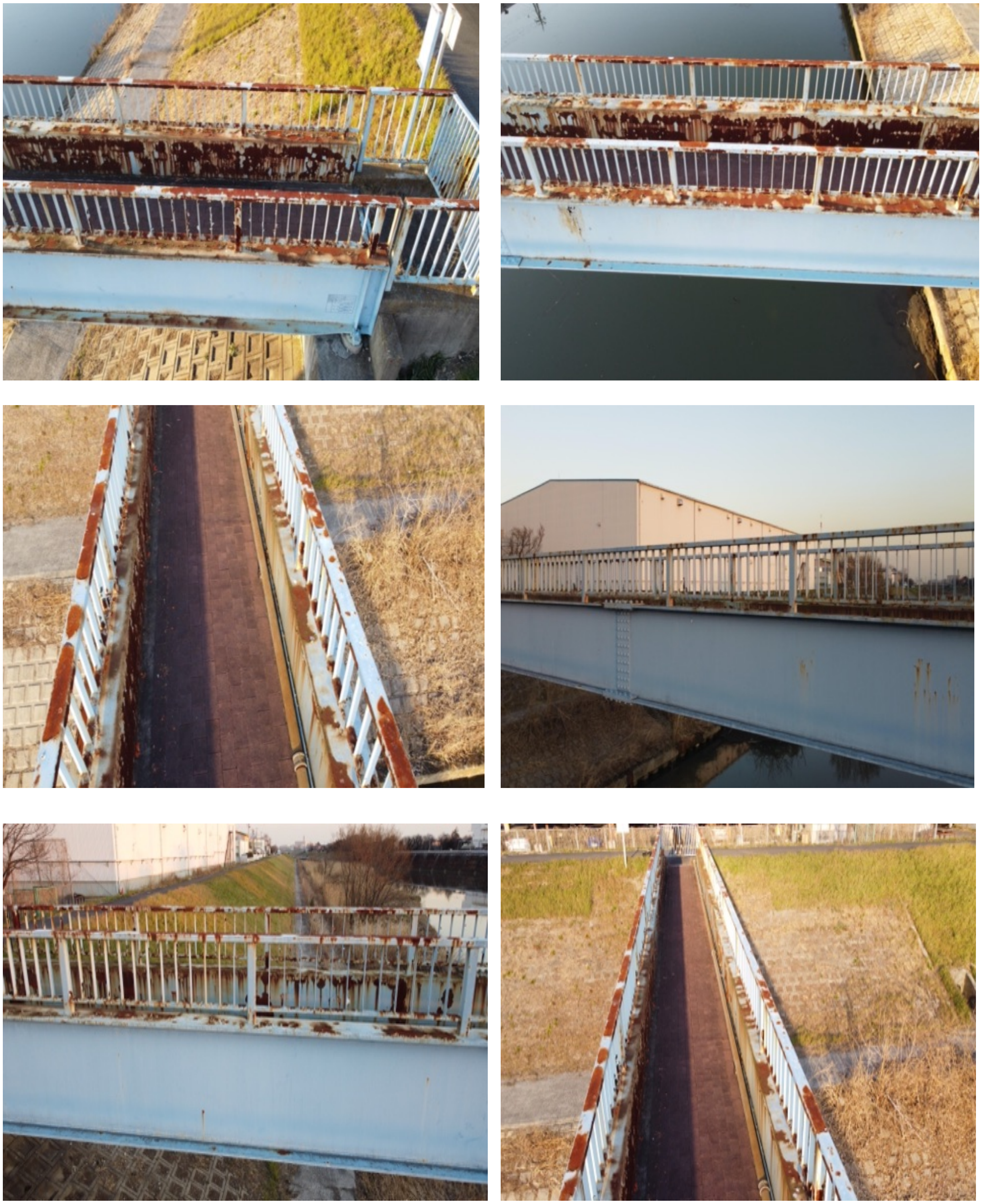


Fig. 65 Sample Bridge Images Captured by the UAV

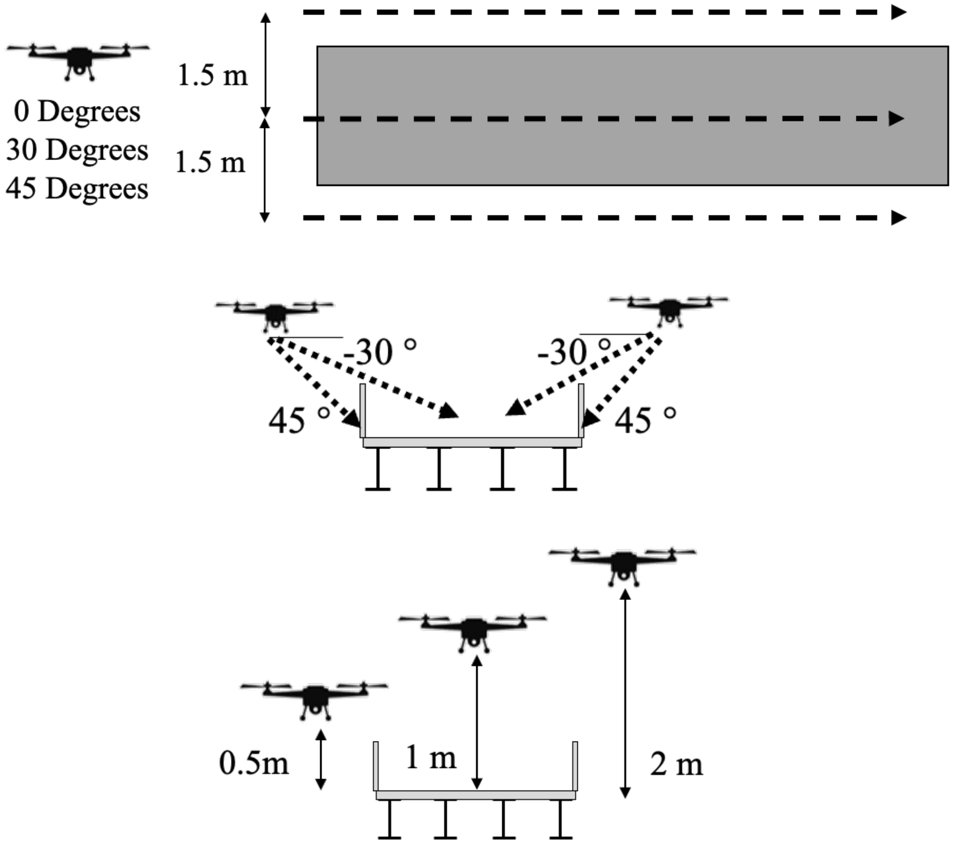


Fig. 66 Bridge Top View UAV Flight Path for Capturing Images

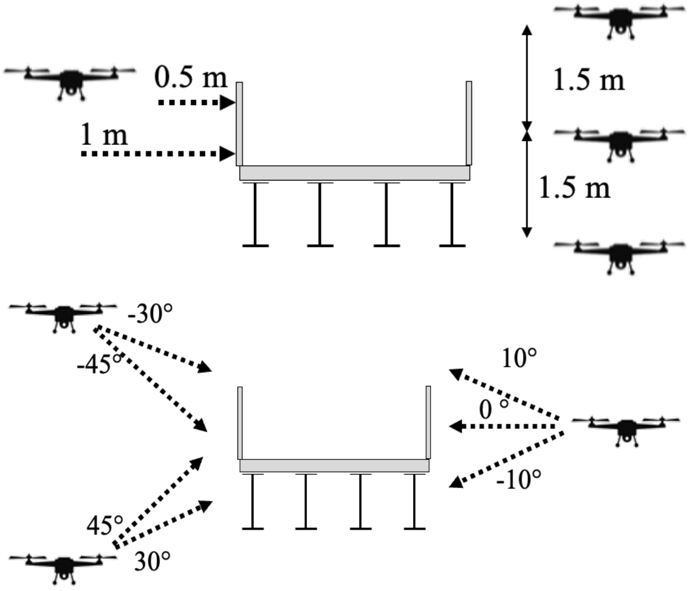


Fig. 67 Bridge Side View UAV Flight Path for Capturing Images

5.3 Bridge 3D Model using Structure-from-Motion

After the gathering of images, Structure from Motion (SfM) method was used to reconstruct the bridge 3D model without segmented damages. The process starts from feature points alignments, cloud points generation, creating the dense model, and adding texture as shown from Figure 68 up to Figure 71. This will be further compared with the 3D Bridge model with segmented corrosion in the next chapter.

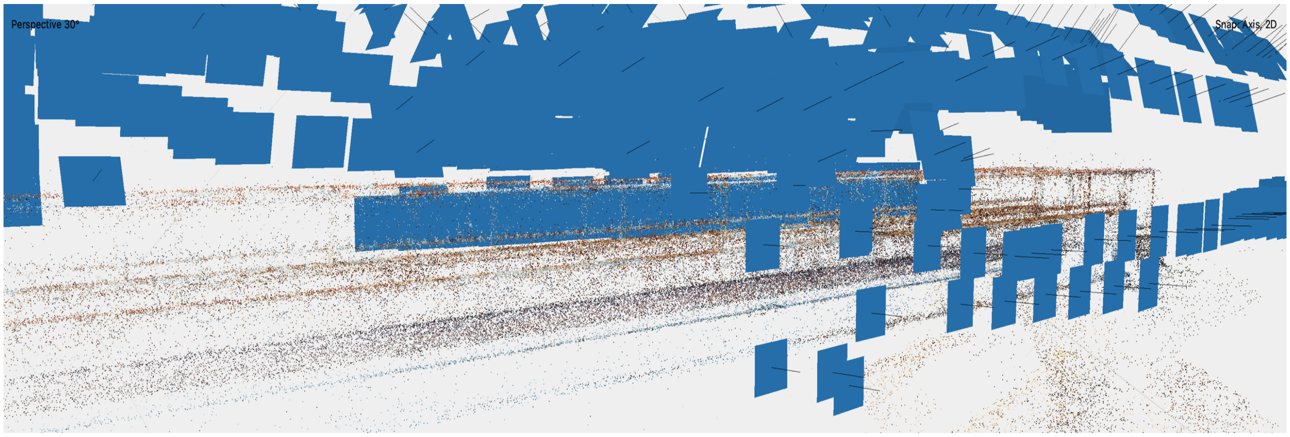


Fig. 68 Feature Points Alignments

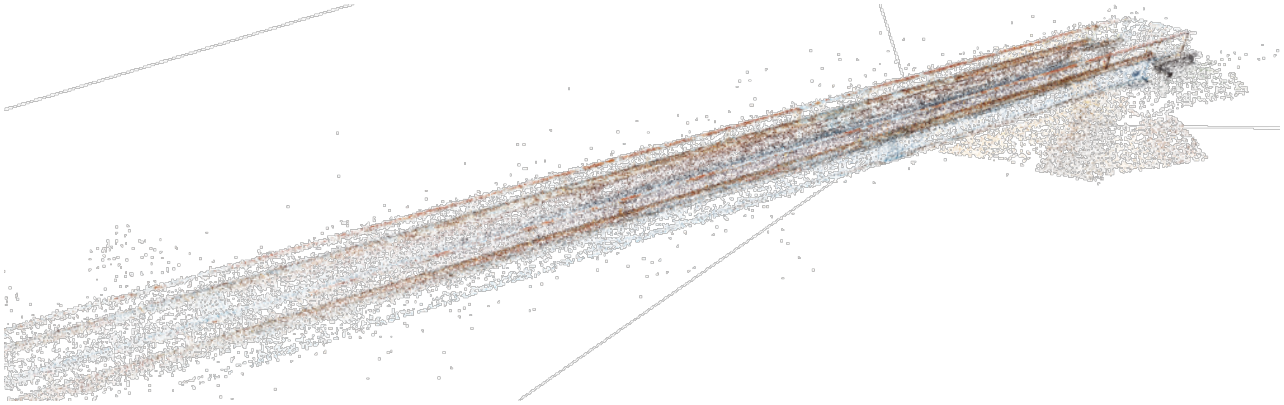


Fig. 69 Cloud Points Generation

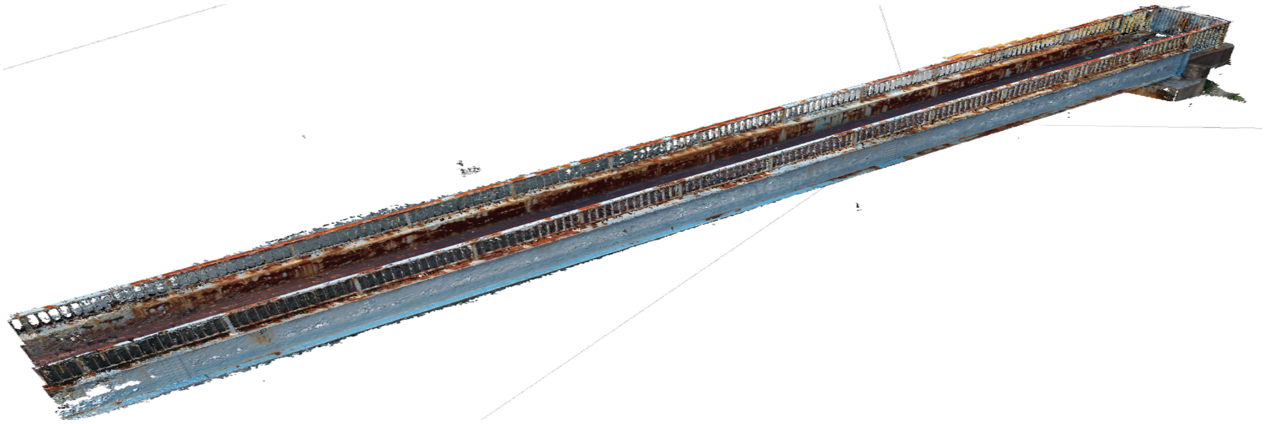


Fig. 70 Bridge Dense Model

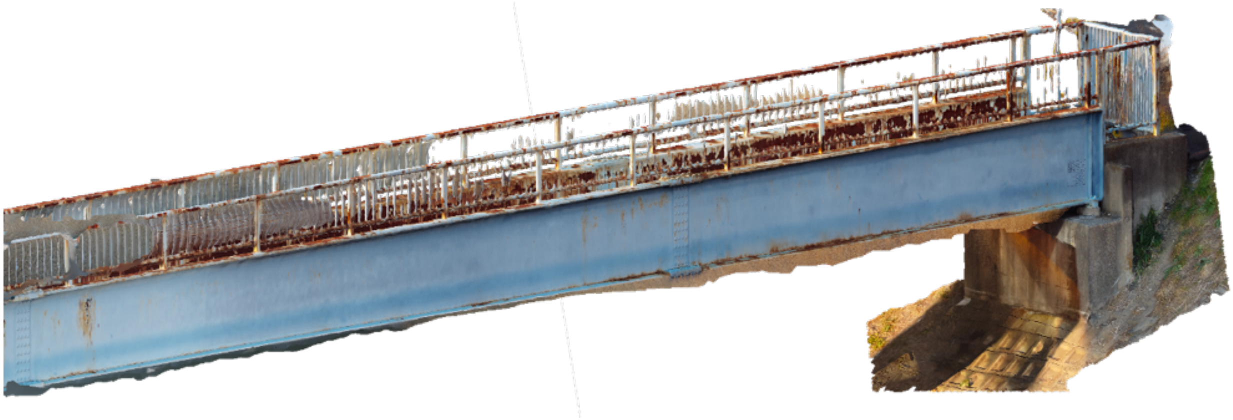


Fig. 71 Bridge 3D Model

Chapter 6: AI Enhanced Vision-Based Bridge Damage Inspection

To improve the AI's ability to further comprehend the reason for damage, identify the source, and suggest a maintenance method, it is crucial task using for 2D or 3D damage detection and diagnosis that AI understand the relationship between damage and its location in the bridge. Existing damages and the bridge's structural component are related in a way that can aid in determining the cause of the damage and predicting how it will progress. This study firstly segmented corrosion damages using Deeplabv3+ based on the UAV images gathered in chapter 5. Then the 3D bridge model was reconstructed with the segmented corrosion damages using SfM.

Recently, there are various of structural damage assessment efforts using RGB images, however, it has not yet reached the same degree of relevant applicability as those from conventional bridge inspection. For instance, a typical damage description in engineering practice can be both quantitatively and semantically rich with 3D insights (for example, "on the bottom and side surface of [Structural Element A] there exists a crack with a length [X] m and a width of [Y] mm"); however, in an image-based process, most studies only reveal detected damage in a localized structural surface with no or few geometric references to the related structural elements that may be 3D. It is necessary to identify structural elements and damage patterns in a 3D subspace (including detection, localization, and quantification) in order to enable such practical applicability.

Since there are currently no 3D databases available for understanding structural components and damage patterns, this study proposed a 3D dataset for bridge structural component segmentation and damage detection. In order to collect data for this investigation, a low-cost LiDAR-enabled imaging device (Intel RealSense) was used to create a special 3D dataset based on an actual bridge. Additionally, the use of 3-dimensional data contains additional, extremely significant information like component surfaces and damage continuation patterns.

Figure 72 shows the proposed enhanced bridge inspection using 3D bridge component and damage segmentation with 3D model generation and damage evaluation concept. However, this studies progress limitation is on the RGBD data gathering and 3D model reconstruction of bridge without and with segmented damages which will be discuss in this chapter.

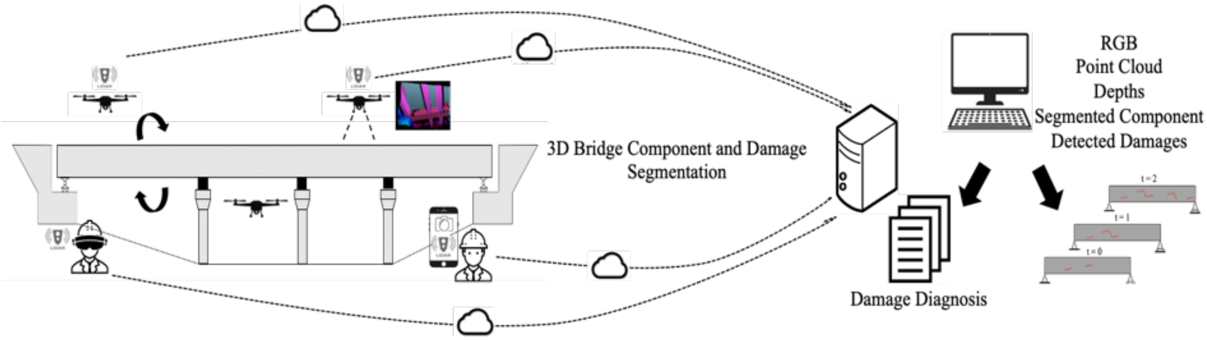


Fig. 72 Enhanced 3D Bridge Component and Damage Segmentation
with Damage Evaluation Concept

6.1 2D Deep Learning Based Damage Segmentation

The suggested methodology encourages an advanced strategy and a paperless way for maintaining bridges that makes use of recent technological development. This study successfully used an unmanned aerial vehicle (UAV) to inspect a two-span steel bridge. It then used a trained DeepLabv3+ model [89] to segment the corrosion in the collected images, reconstruct a 3D bridge model with segmented corrosion, and displayed it in a Mixed Reality (MR) [90] platform. This is still a preliminary development, and each step still must be further improved as a recommendation.

6.1.1 Training Dataset

The training dataset is made up of the UAV and onsite bridge inspection report photos as visualized in Figure 73. Table 21 shows the statistics of the 2290 photos that were used [91]. Each image was manually annotated using an open-sourced annotation tool [92].

Table 21. Training Dataset for DeepLabv3+

Type	Train	Validation	Test	Total
Damage Image from Inspection Report	1684	292	129	2105
Damage Image from UAV	61	13	3	77
Background Image from UAV	78	20	10	108
Total	1823	325	142	2290

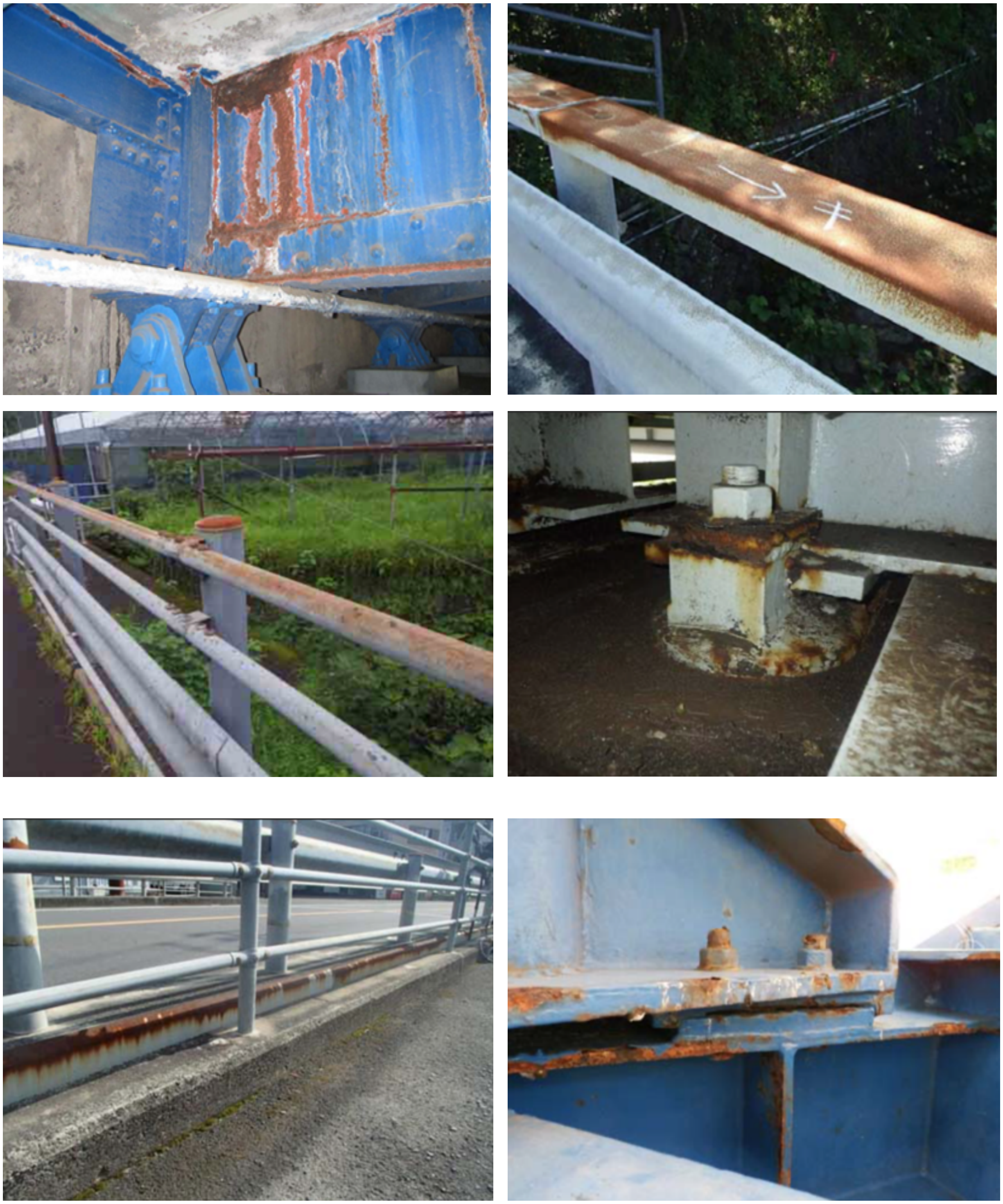


Fig. 73 Sample Training Image for Deeplabv3+

6.1.1 Deeplabv3+

The trained model used DeepLabv3+ architecture [89], which consist of an encoder-decoder architecture as shown in Figure 74, which has an excellent performance for object segmentation application. The encoder learns the features of the target class and create a feature map. Then, the decoder regenerates the input image based on the feature map processed by the encoder. The environment used was Pytorch version 1.9.0, the backbone used was Resnet-101cuda 10.2, GPU of Testa P1000, and the platform was in googlecolaboratory pro using python as programming language. Before training, there were a few minor adjustments made using a batch size of 6, a crop size of 78, a learning rate of 0.001 with poly decay, and a loss function of cross entropy.

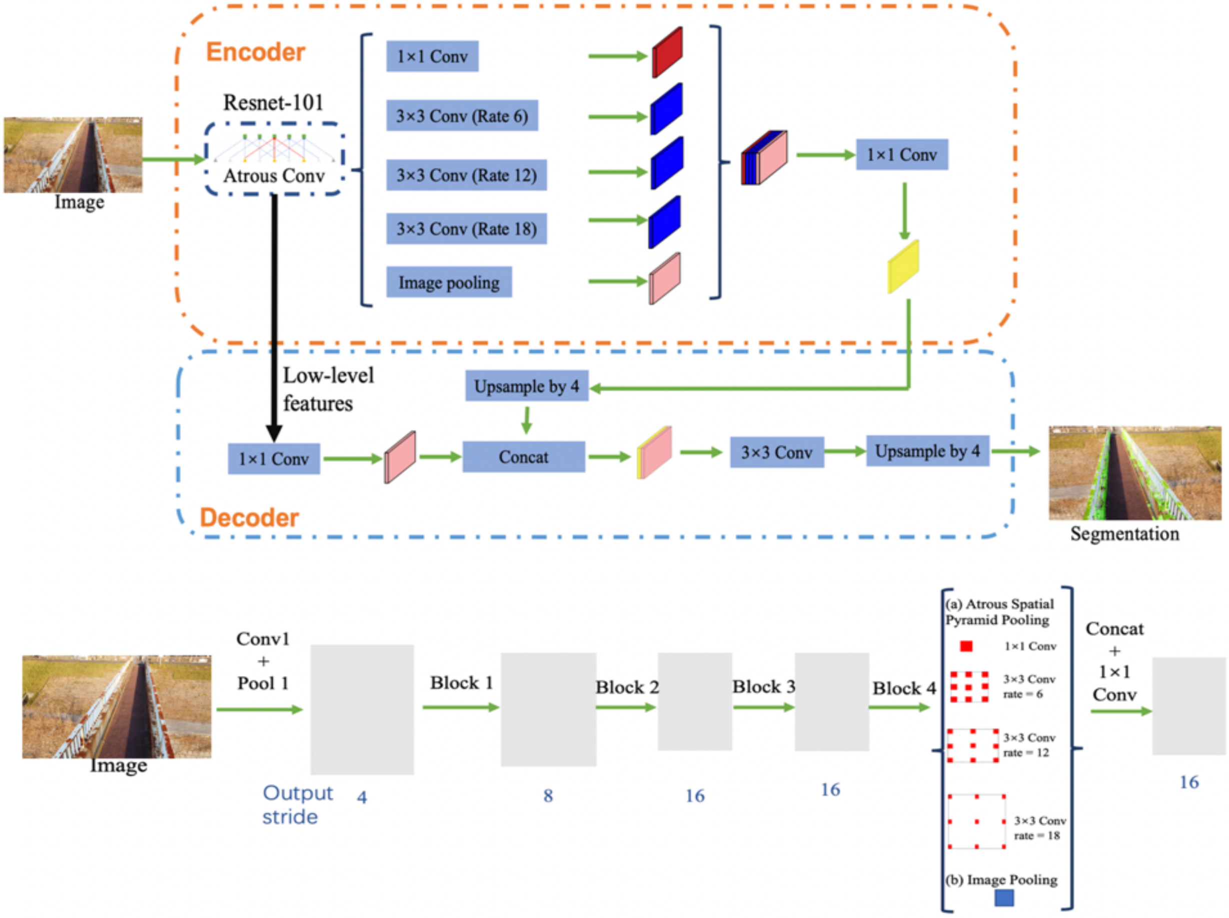


Fig. 74 Deeplabv3+ Architecture

After 50 epochs, the frequency weighted Intersection of Union (fwIoU) and validation mean Intersection of Union (mIoU) are 80.81% and 78.43%, respectively as shown in Table 22 [90]. The average accuracy rate of the damage segmentation is described by mIoU, which is crucial for training the model.

Table 22. Evaluation Index Results

Parameters	Validation Dataset	Test Dataset
Pixel Accuracy (%)	94.77%	94.94%
Mean Pixel Accuracy (%)	87.31%	85.32%
mIoU (%)	80.81%	78.43%
fwIoU (%)	90.37%	90.77%

6.2 Bridge Segmented Corrosion Damage

Figure 75 displays the Deeplabv3+ segmented corrosion damages from the UAV bridge images that are marked in green color. The segment features were combined to the original image and was used to reconstruct the 3D bridge model. In this way, the location corrosion damages can be seen easily in the whole bridge, however the accuracy highly depends on the trained Deep learning segmentation model.

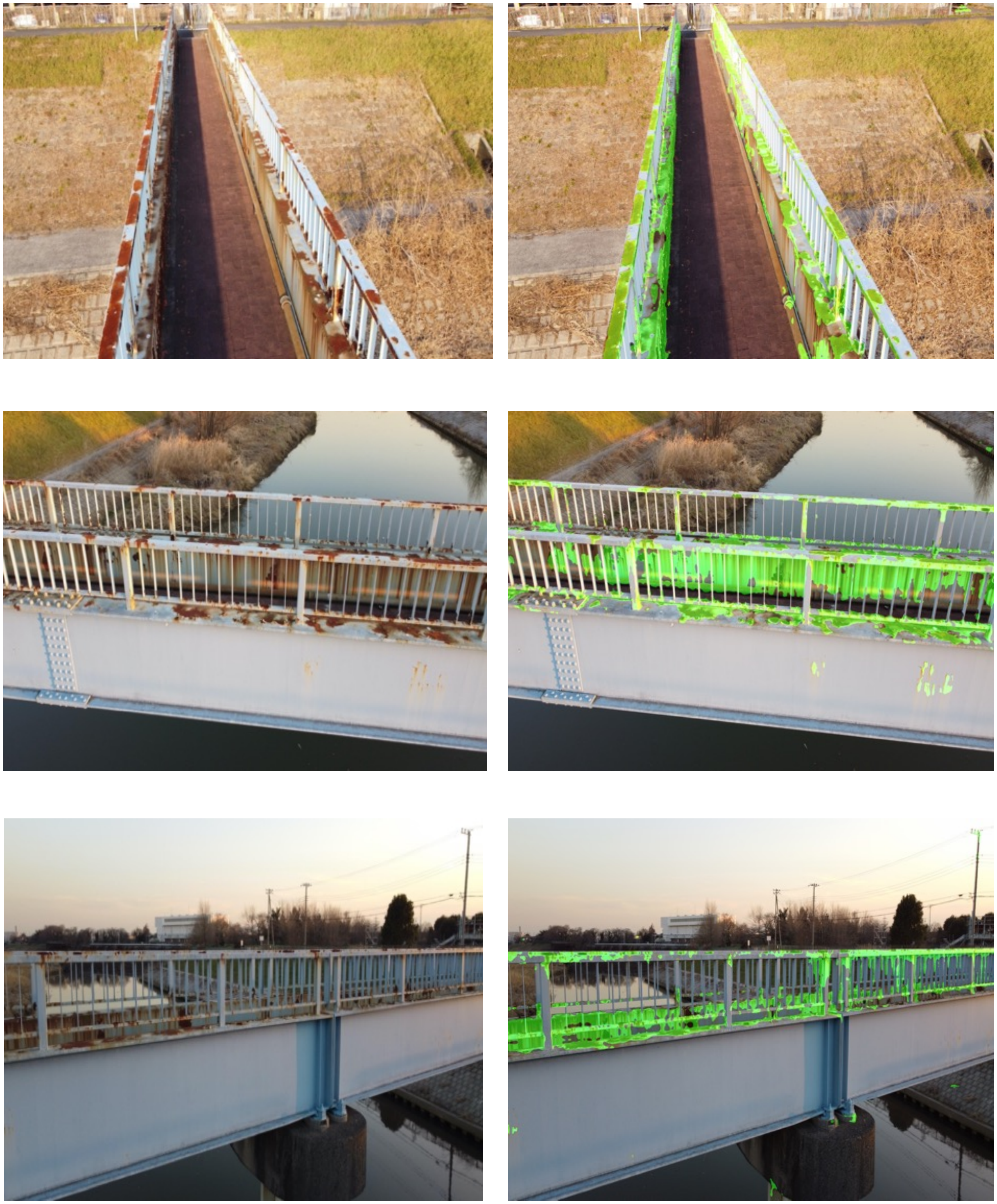


Fig. 75 Segmented Corrosion Damages using the Trained DeepLabv3+ Network

6.3 Bridge 3D Model with Corrosion Damages

Bridge 3D models were reconstructed to observe the global location of corrosion in a comprehensive manner after the corrosion damages were segmented. Starting with the aligned feature points as seen in Figure 76, using the photos with segmented corrosion. Since each pixel was projected, a 70% picture overlap was advised. Then, as shown in Figure 77, the point cloud was created. Figure 78 illustrates the dense model that was produced by giving the point cloud a depth. Finally, texture was included to the bridge model, as seen in Figure 79. The final output was then compared with the 3D bridge model without segmented corrosion. It can be observed that the corrosion damages in green color was projected to the 3D bridge model, and the location of the corrosion damage can be easily seen. However, further studies such as evaluation of the relationship of the corrosion to the bridge component, quantification of the corrosion damage per specific area, and other damage diagnosis should be examined.

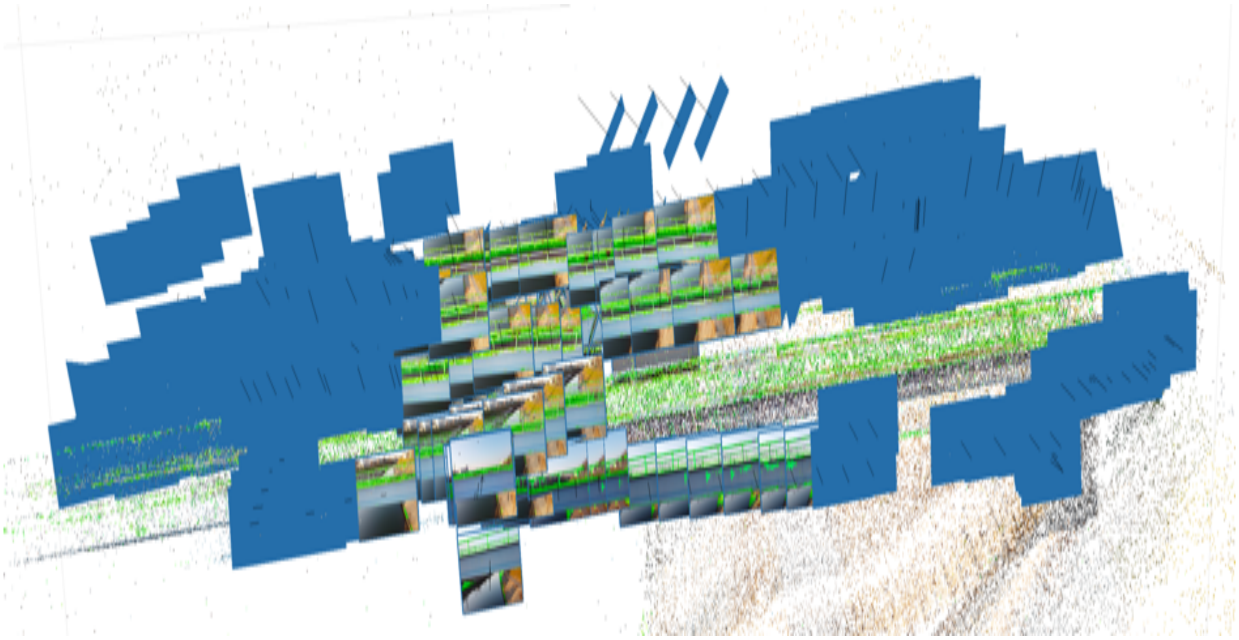


Fig. 76 Feature Points Alignments of Bridge Images with Segmented Corrosion

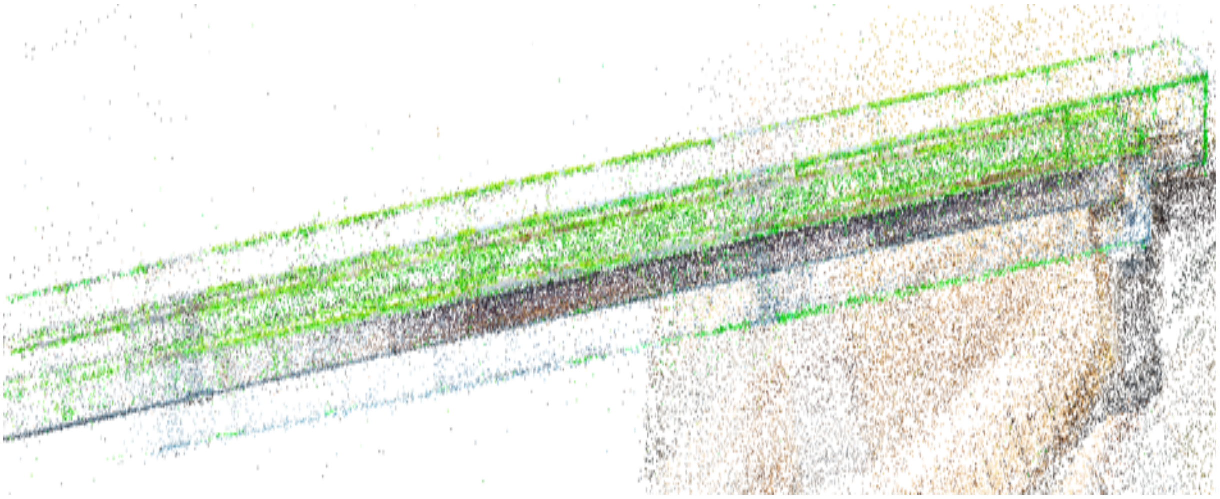


Fig. 77 Cloud Points Generation of Bridge Images with Segmented Corrosion

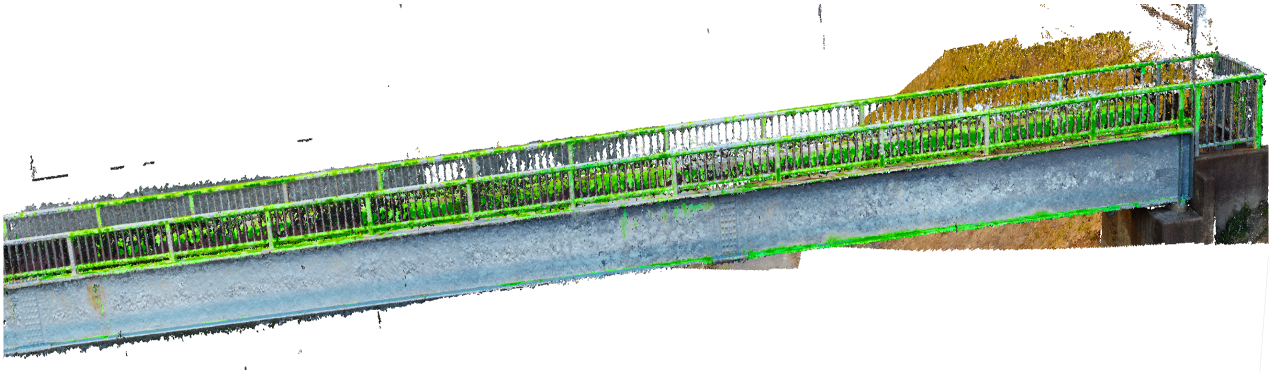


Fig. 78 Bridge Dense Model of Bridge Images with Segmented Corrosion

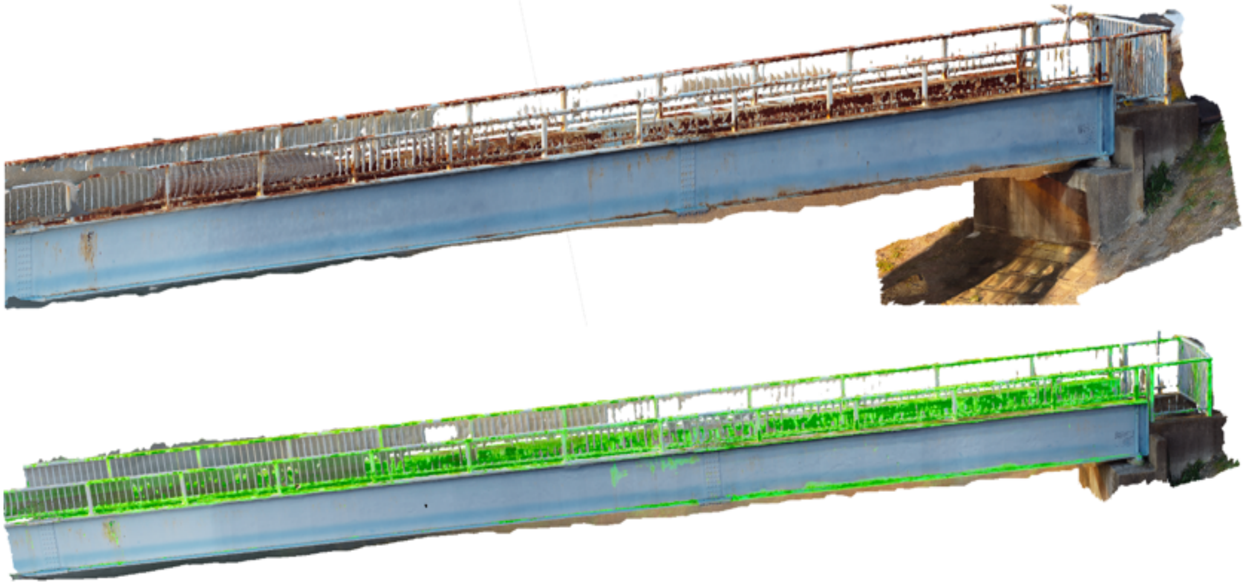


Fig. 79 Final 3D Bridge Model with and without Segmented Corrosion

6.4 Mixed-Reality Platform Inspection Visualization

Hololens2 [90] was used to visualize the bridge 3D model image into Mixed Reality as shown in Figure 80. The operator wearing this device can resize, rotate, and move the bridge to see it clearly. This platform enables the engineers to observe the bridge health status and inspection progress through time. The generated bridge 3D models can be stored which can serves as a time capsule of its physical state after several periodic inspection. The deterioration of the bridge can be easily monitored using the proposed platform compared to sorting out a large amount of bridge inspection paper reports. This method promotes a paper-less and can make the inspection data storage more organized. However, development of bridge inspection application can be a further study to incorporate more inspection details.

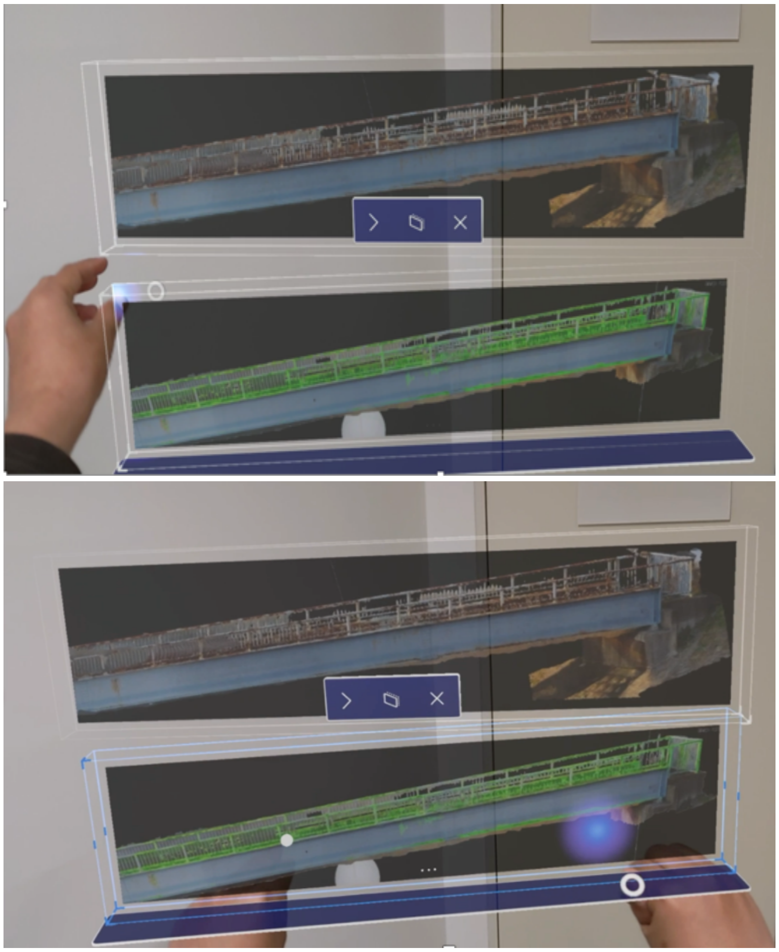


Fig. 80 Mixed-Reality Platform Visualization

6.5 3-Dimensional Bridge Component and Damage Segmentation

Using an economical LiDAR-enabled imaging equipment (Intel RealSense) [93], this study produced a special 3-dimensional dataset based on a real bridge construction for future CNN model segmentation training as shown in Figure 81. In order to better determine the damage source and make a diagnosis, this study intends to include the segmented bridge damages to the bridge component.

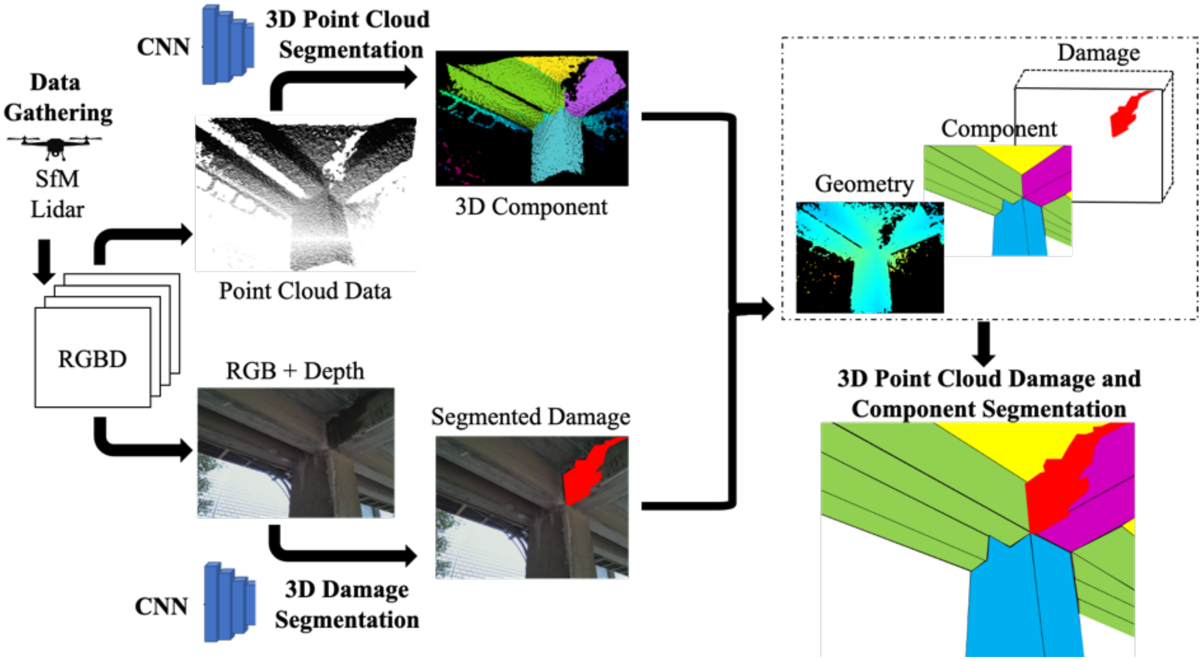


Fig. 81 Proposed 3D Point Cloud Damage and Component Segmentation Concept

During the on-site bridge inspection as shown in Figure 82, RGBD data was gathered using a low-cost lidar as shown in Figure 82 and Figure 83, and annotated using two open software programs. The beam, column, transverse girder, and main girder are the major emphasis of the dataset for segmenting bridge components, while corrosion, spalling, cracks, and water leaks are among the damages.

Fig. 82. Intel® RealSense™ Lidar Camera L515.

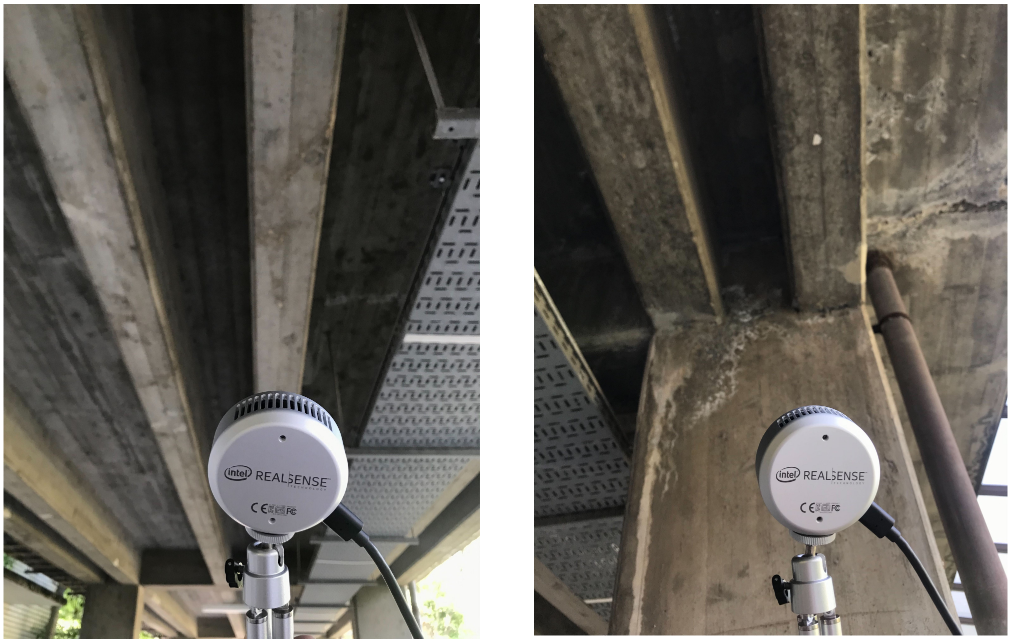


Fig. 83 Data Gathering using RealSense Lidar Camera L515

Both an RGB image and a depth image can be recorded simultaneously using the Intel® RealSenseTM Lidar camera as shown in Figure 84. While the depth image is 320 by 240 pixels in size, the RGB image resolution is 640 by 480 pixels. The specifications of the lidar device can be seen on table 23. The quality of the point cloud data, the lighting, and the distance of the Lidar from the surface are some of the difficulties in data collection. Both the RGB and point cloud annotations are saved in json file.

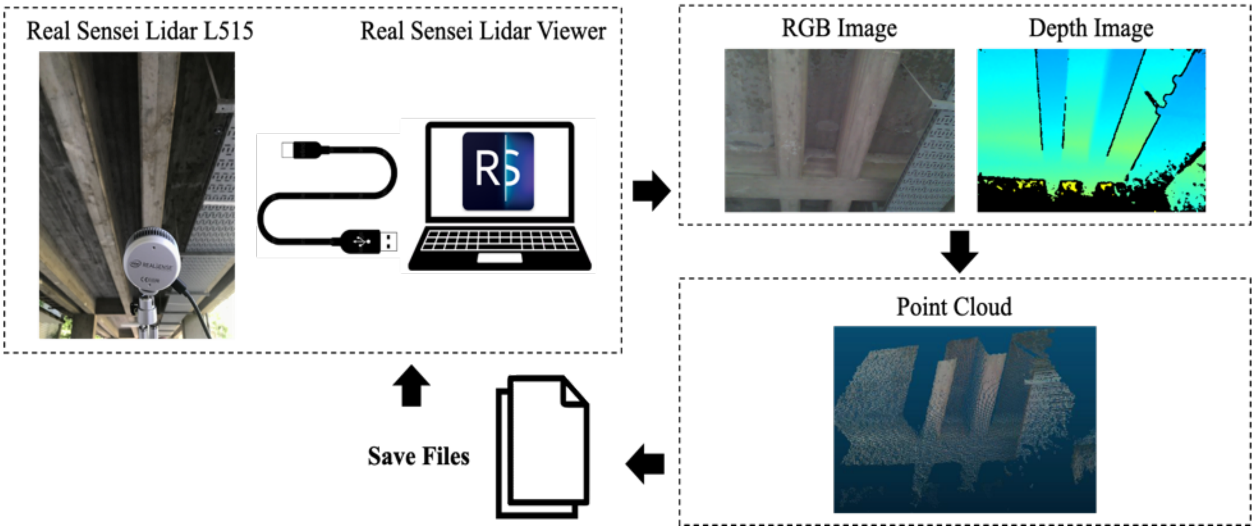
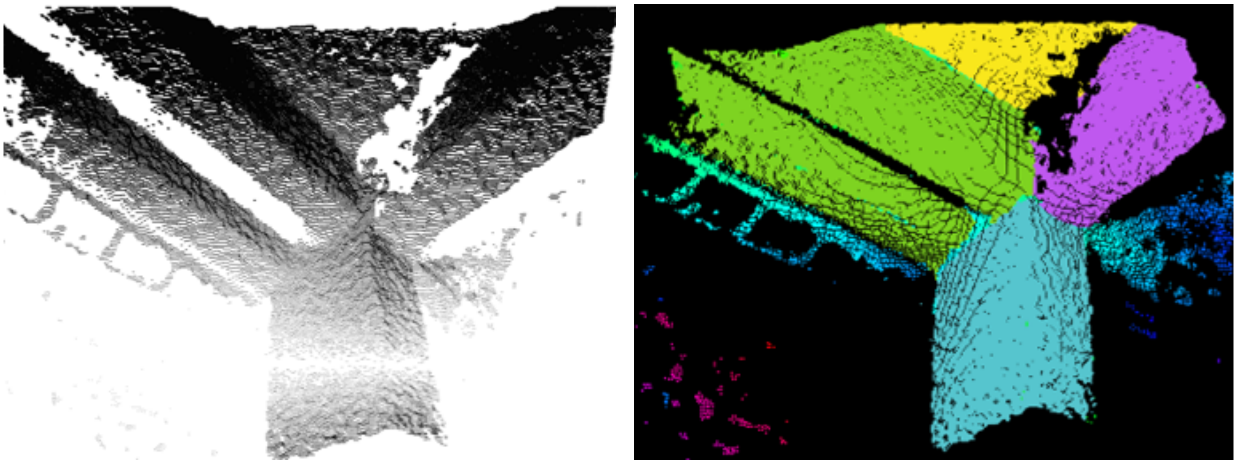


Fig. 84 Point Cloud Data Gathering Process

Table 23. RealSense Lidar Camera L515 Specifications

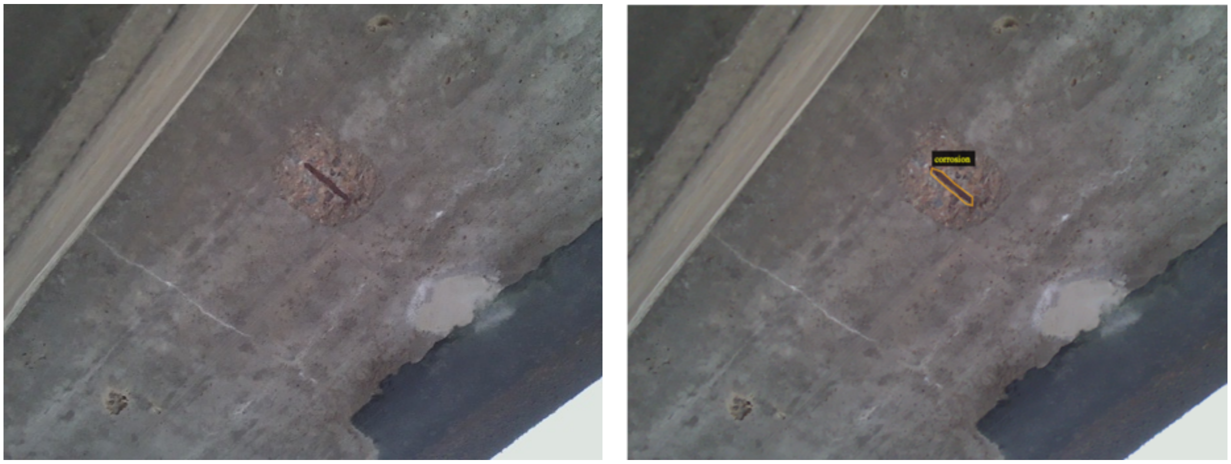
Specifications	Real Sense Lidar L515
Weight	100 grams
Depth Field of View (FoV)	70° × 55° (±3°)
Depth Output Resolution	1024 x 768, 30fps
Depth Accuracy	5 mm ~ 14 mm thru $9m^2$
RGB Frame Resolution	1920 x 1080, 30 fps
Ideal Range	0.25 m to 9 m
RGB Sensor FoV (H x V)	70° × 43° (±3°)

An open-sourced program called "supervisely" was used to annotate the point cloud of the bridge component [91]. The classes of structural members are "main girder," "transverse girder," "deck," and "column." The results of the point cloud annotation can be seen in Figure 85, where the main girder, transverse girder, column, and deck are each represented by a different color: violet for the main girder, green for the transversal girder, yellow of the deck, and blue for the column. Figure 86 shows the annotation of damages using the RGB images.



(a) (b)

Fig. 85 Point Cloud Bridge Component Annotation (a) Point-Cloud Data (b) Annotation



(a) (b)

Fig. 86 2DDamage annotation: (a) RGB image and (b) annotated image with damage

Chapter 7: Conclusion

Based on the results and evaluation gathered for the proposed machine-learning based bridge seismic isolation design process, this study concludes the following:

1. The created nonlinear model library consisted of Bilinear, Modified Bilinear, Clough, Trilinear, Bouc-Wen, and Modified Bouc-Wen model.

2. The HDR-S quasi-static loading experiment hysteresis data was used observe the feasibility of the train two-layer neural network model that were intended to classify the nonlinear model. The validation accuracy was 0.99 and the validation loss was 0.03.

3. For the nonlinear parameter identification, the six trained neural network models using the different nonlinear models from the nonlinear model library was able to suggest the nonlinear parameters of the hysteresis experimental data.

4. The ANN suggested nonlinear parameters was compared to other optimization method [KH Method], and the result has a 3% to 5% difference in the contribution rate.

5. For the design and performance evaluation, this study trained three ANN models for bearings shear strain classification, bearings shear strain prediction, and design suggestions. However, the proposed method is highly dependent on the training data. This study considered a 2DoF bridge with fixed foundation, more complicated structure model and different types of bearings is suggested for further improvement.

On the other hand, for the proposed enhanced bridge damage detection and maintenance, the following conclusions are as follows:

1. The use of UAV for data gathering can reduce the manpower and can inspect some parts of the bridges that are difficult to access without the need of expensive equipment's. The suggested flight path was proposed for a steel bridge, however more complicated bridges require a different strategy of flight path inspection. Some challenges encountered includes the UAVs limited battery life which depends on the type of UAV used, the weather condition specially the wind and light, the environment factors like trees or other obstacles, and the skill of the drone pilot.

2. For the damage segmentation and 3D model reconstruction, this study focuses on corrosion damages which are visible and have a large area on the bridge that's why it can be easily seen in the projection of the segmented corrosions feature points in the 3D model. However, damages such as cracks which are thin and difficult to be seen might be difficult to project into the 3D model using the proposed method and is suggested to be further studies.

3. The mixed-reality platform could export and display the images of the 3D model of bridge with and without corrosion and can be access in the office for engineers to evaluate. The storage of the 3D model of the bridge during repetitive inspection can help to visualize the damage progression and the maintenance progress. However, the researcher suggests exploring the creation of application inside the mixed-reality platform to include more inspection information for future works.

4. Lastly, for the point cloud bridge component data and annotated damages, the dataset can be accessed and will be available to public for further segmentation training. The concept of 3D bridge component segmentation and incorporation of damages in a 3D environment was proposed to further evaluate the damages and to visualize the relationship of damages in different component surfaces.

REFERENCES

1. Japan Road Association (1964). "Design specifications for steel bridges," Maruzen, Tokyo, Japan
2. Road Association (1971). "Guide specifications of seismic design of highway bridges," Maruzen, Tokyo, Japan.
3. Japan Road Association (1980, 1990, 1996, 2002). "Part V Seismic design, Design specifications of highway bridges," Maruzen, Tokyo, Japan.
4. Japan Road Association (1995). "Guide specifications for repair of bridges which suffered damage due to the 1995 Kobe earthquake," Tokyo, Japan.
5. Kawashima, K. (2012). Damage of Bridges Due to the 2011 Great East Earthquake. *Journal of Japan Association for Earthquake Engineering*, 12(4).
6. Kawashima, K., Matsuzaki, H. (2012). Damage of Road Bridges by 2011 Great East Japan (Tohoku) Earthquake*, 15th World Conference on Earthquake Engineering (WCEE)*
7. Ghasemi, K., Otsuka, H., Cooper, J., Nakajima, H. (1996). Aftermath of the Kobe Earthquake, Public Roads*, U.S. Department of Transportation, Federal Highway Administration*, 60(2)
8. Kawashima, K., (2004) Seismic Isolation of Highway Bridges, Journal of Japan Association of Earthquake Engineering, 4(3).
9. Fujita, T.; Sasaki, Y.; Fujimoto, S. (1990). Tsuruya, C. Seismic isolation of industrial facilities using lead-rubber bearing. *JSME Int. J. Ser. 3 Vib. Control Eng. Eng. Ind. 33*, 427–434.
10. Tsopelas, P., Constantinou, M.C., Kim, Y.S., Okamoto, S. (1996) Experimental Study on FPS System in Bridge Seismic Isolation, *Journal of Earthquake Engineering and Structural Dynamics*, Vol. 1 pp. 65-78, 1996
11. Roy, S., S. Dash, S. (2016). Dynamic Behavior of Multi-span Continuous Girder bridge with Isolation bearings, *International Journal of Bridge Engineering (IJBE),* 06(2), pp. 01-23
12. Okui, Y., Onoue, Y., Sato, T., Imai, T. (2017). Temperature Dependence of Mechanical Behavior of High Damping Rubber Bearings Considering Self Heating, *Journal of Japan Society pf Civil Engineers*, Ser. A1 (Structural Engineering and Earthquake Engineering (SE.EE). Vol. 73(1), pp. 165-173). JSCE. Volume 73
13. Mullins, L. (1947). Effect of Stretching on the Properties of Rubber, Journal of Rubber Research, Vol. 16, pp. 275-289

14. Nguyen, D.A., Dang, J., Okui, Y. (2015). A.F.M.S. Amin, S. Okada, and T. Imai. An Improved Rheology Model for the Description of the Rate-Dependent Cyclic Behavior of High Damping Rubber Bearings. *Soil Dynamics and Earthquake Engineering,* 77: 416-431

15. Itoh, Y., Gu, H., Satoh, K., Yamamoto, Y. (2006). Long-Term Deterioration of High Damping Rubber Bridge Bearing, Dobokku Gakkai Ronbunshuu A. Vol. 62, No. 3., pp. 595-607

16. Itoh, Y., Gu, H., Satoh, K., Kutsuna, Y. (2006) Experimental Investigation on aging behaviors of rubbers used for bridge bearings, *Journal of Structural Mechanics and Earthquake Engineering, JSCE,* No. 808/I-74, pp. 17-32

17. Yar, M., Hammond, JK, (1976). Modelling and Response of Bilinear Hysteresis Systems. *Journal of Engineering Mechanics Division.* 102: pp. 249-263

18. Ramberg, W., Osgood, W.R. (1943). Description of Stress Strain Curves by Three Parameters. Technical Note 902, *National Advisory Committee For Aeronautics*

19. Ismail, M., Ikhouane, F., Rodellar, J. (2009) The Hysteresis Bouc-Wen Model, a survey. *Archives of Computational Methods in Engineering*, Vol. 16, pp. 161-188

20. Dang, J., Igarashi, A., Murakoshi, Y. (2013). Nonlinear Numerical Hysteresis Model for Bi-Directionally Loaded Elastomeric Isolation Bearings, *JAEE*, Vol. 2

21. Kikuchi, M., Aiken, I. (1997). An Analytical Hysteresis Model for Elastomeric Seismic Isolation Bearing, *Earthquake Engineering and Structural Dynamics*, Vol. 26, pp. 215-231

22. Dang, J., Tan, Y., Igarashi, A., Himeno, T., Hamada, Y. (2021) Hybrid Simulation Test for High Damping Rubber Bearing in Low Temperature. *Proceedings of 41st JSCE Earthquake Engineering Symposium 2021*. 41.C15-2330

23. Salic, R.B.; Garevski, M.A. Milutinovic, Z.V. (2008). Response of lead-rubber bearing isolated structure. *In Proceedings of the 14th World Conference on Earthquake Engineering (14WCEE)*, Beijing, China, 12–17

24. Tan, Y., Dang, J., Igarashi, A., Himeno, T., Hamada, Y. (2021) S. Uno, and K. Terachi: Hysteretic Restoring Force Model of HDRB including Thermo-Mechanical Coupling Effect and Verifications on Hybrid Simulation, *76th Annual Meeting JSCE. 2021*. Vol 76. CS2-29

25. Tan, Y., Dang, J., Igarashi, A., Himeno, T., Hamada, Y. (2021) Development of Restoring Force Model Considering Mullins Effect and Temperature Dependence of High Damping Rubber Bearing. *Summary of the 16th Annual Meeting of JSCE for Earthquake Engineering*. Vol. 16, T2021-059

26. Terachi, K., Dang, J., Hamada, Y., Himeno, T., Igarashi, A., Tan, Y. (2021). Low-Temperature Hybrid Loading Test of Spring-Confined Pb High Damping Laminated Rubber Bearing. Vol. 76. *Annual Meeting of JSCE*.
27. Bridge Seismic Control Design Method Draft: Seismic Isolation Structure Research Committee for Road and Bridges, 2011
28. I. Sarker, Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, doi. 10.1007/s42979-021-00592-x.2021
29. John, G.H., Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers, *In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence,* Morgan Kaufmann Publishers Inc. pp. 338-345
30. Roth, V., Steinhage, V. (1999) Nonlinear Discriminant Analysis using Kernel Functions, *Advances in Neural Information Processing Systems*, MIT Press. pp 41-48,
31. LeCessie, S., Van Houwelingen, J.C. (1992). Ridge Estimators in Logistic Regression. *Applied Statistics*, pp. 191-201
32. Aha, D.W., Kibler, D., Albert, M. (1991). Instance-Based Learning Algorithms. *Machine Learning*. pp. 37-66
33. Schlkopf, B., Burges, C.J.C., Smola, A.J. (1998). Advances in Kernel Methods Support Vector Learning. *Cambridge*, Mass
34. Quinlan, J.R. (1994) C4.5, Programs for Machine Learning. *Machine Learning*. Vol. 16. pp. 235-240.
35. Breiman, L. (2001). Random Forests. Machine Learning. Vol. 45, pp. 5-32
36. McCulloch, W., Walter, P. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *The Bulletin of Mathematics Biophysics*. Vol. 5, pp. 115-133
37. Wu, Y., He, J., Huang, G., Yao, H., Zhang, P., Xu, W., Guo, M., Li, Y. (2019). Enhanced Classification Models for Iris Dataset, *7th International Conference on Information Technology and Quantitative Management (ITQM 2019).* Vol. 162, pp. 946- 954
38. Vyas, S., Upadhyay, D. (2014) Classification of Iris Plant using Feedforward Neural Network. *International Refereed Journal of Engineering and Science(IRJES).* 3(12), pp. 65-69
39. Suzuki, T., Matsuoka, Y. (2017) Basic Research on the Construction of Material Constitutive Law by Recurrent Neural Network. *Journal of Architectural Institute of Japan*, Vol. 82, No. 734

40. Fisher, R. A. (1920) A Mathematical Examination of The Methods of Determining the Accuracy of Observation by the Mean Error, And by The Mean Square Error, Mon. Not. R. Astron. Soc., 80, 758–770, https://doi.org/10.1093/mnras/80.8.758

41. Fisher, R. A. (1922) On the Mathematical Foundations of Theoretical Statistics, Philos. T. R. Soc. Lond., 222, 309–368, https://doi.org/10.1098/rsta.1922.0009,

42. Sun, H., Burton, H., Huang, H. (2021) Machine Learning Applications for Building Structural Design and Performance Assessment: State-of-the-art Review. *Journal of Building Engineering,* Vol. 33

43. Matsuda, Y., Otsuka, H., Kabuyama, Y.,Yahagi, W. (1998) Recognition of Nonlinear Historical Behavior of High- Damping Laminated Rubber by Artificial Neural Network and Application to Dynamic Analysis. *Journal of Japanese Society of Civil Engineers,* Review no. 605. Vol. 10, pp 29-36

44. Anderson, D., Hines, E.L., Arthur, S.J., Eiap, E.L. 1997, Application of Artificial Neural Networks to the Predictions of Minor Axis Steel Connections. *Computer and Structures*, 63(3), pp. 685-692

45. Fahmy, A., Madawy, M., Gobran, Y. (2016). Using Artificial Neural Networks in the Design of Orthotrophic Bridge Decks. *Alexandria Engineering Journal*. 55(4), pp. 3195-3203, doi. https://doi.org/10.1016/j.aej.2016.06.034

46. Kim, T., Kwon, O., Song, J. (2019) Response Prediction of Nonlinear Hysteretic Systems by Deep Neural Networks, *Neural Networks*, Vol. 111, pp. 1-10

47. Oh, B., Lisix, B. G., Park, S., Park, H. (2020) Neural Network-Based Seismic Response Prediction Model for Building Structures using Artificial Earthquakes. *Journal of Sound and Vibration.* Vol 468

48. Nguyen, H., Dao, N., Shin, M. (20220 Machine Learning-Based Prediction for Maximum Displacement of Seismic Isolation Systems, Journal of Bridge Engineering, Vol. 51

49. Hu, S., Zhu, S., Wang, W. (2022). Machine Learning-driven Probabilistic Residual Displacement-based Design Method for Improving Post-Earthquake Repairability of Steel-Moment-Resisting Frames using Self-Centering Braces. *Journal of Building Engineering*, 61(1), doi. https://doi.org/10.1016/j.jobe.2022.105225

50. MLIT (Ministry of Land, Infrastructure, Transport and Tourism in Japan) 2014, White Paper, Section 2, Paragraph 2.

51. 2021 Report Card for America's Infrastructure, ASCE, https://infrastructurereportcard.org/ at-item/bridges-infrastructure.
52. Shirato, M., Tamakoshi, T., Bridge Inspection Standards in Japan and US, https://www.pwri.go.jp/eng/ujnr/tc/g/pdf/28/28-3-1_Matsumoto
53. Zhang, Y., Yuen, K. (2022) Review of Artificial Intelligence-Based Bridge Damage Detection, *Journal of Advances in Mechanical Engineering*, https://doi.org/10.1177/16878132221122770
54. Goodfellow, I, Bengio, Y., Courville, A. (2016) Deep Learning, *MIT Press*, pp. 321-359
55. Taher, S., Dang, J. (2020) Autonomous Multiple Damage Detection and Segmentation in Structures Using Mask R-CNN, *Experimental Vibration Analysis for Civil Engineering Structures*, doi. 10.1007/978-3-030-93236-7_45
56. Dang, J., Mizumoto, T., Chun, P., Liu, J., Fujishima, T. (2021) Multi-Type Bridge Damage Detection Method Based on YOLO, *Japan Society of Civil Engineering*, doi. 10.11532/jsceiii.2.J2_447,
57. Shi, J., Dang, J., Cui, M., Zuo, R., Shimizu, K. (2021) Improvement of Damage Segmentation Based on Pixel-Level Data Balance using VGG-Unet, *Applied Sciences*, Vol. 11, pp. 518.1-17, doi: 10.3390/app11020518
58. Ni, F., Zhang, J., Chen, Z. (2018) Zernike-moment Measurement of Thin-Crack Width in Images Enabled by Dual-Scale Deep Learning. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 34, doi. 10.1111/mice.12421
59. Narazaki, Y., Hoskere, V., Hoang, T.A. and Spencer Jr., B.F. (2017) Vision-Based Automated Bridge Component Recognition Integrated with High-level Scene Understanding. *The 13th International Workshop on Advanced Smart Materials and Smart Structures Technology*
60. Narazaki, Y., Hoskere, V., and Spencer Jr., B.F. (2021) Synthetic Environments for Facilitating Field Implementation of Vision-Based Structural Management Techniques. *Journal of Artificial Intelligence and Data Science*, Vol. 2, Issue J2, pp. 20-28
61. Kim, H., Narazaki, Y., and Spencer Jr., B.F. (2023) Automated Bridge Component Recognition using Close-Range Images from Unmanned Aerial Vehicles. *Engineering Structures*, Vol. 274,
62. Kim, H., Yoon, J. and Sim, S. (2022) Automated Bridge Component Recognition from Point Clouds using Deep Learning, Structural Control health Monitoring, 27(9)

63. Inadomi, S. and Chun, P. (2021) Segmentation of Bridge Point Clouds using Point Cloud Imaging and Deep Learning, Journal of Artificial Intelligence and Data Science, Vol. 2, Issue J2, pp. 418-427

64. Yamane, T., Chun, P. and Honda, R. (2022) Detecting and Localizing Damage Based on Image Recognition and Structure from Motion, and Reflecting it in a 3D Bridge Model, *Structure and Infrastructure Engineering*, doi. 10.1080/15732479.2022.2131845, 2022

65. Kawashima, K. (2004) Seismic Isolation of Highway Bridges. *Journal of Japan Association of Earthquake Engineering*, Vol. 4, No. 3 (Special Issue), 2004

66. Kuroda, H. (2001) Visual Basic Engineering Calculation Program, *CQ Press*, Tokyo, 69-70, (In Japanese)

67. Argyros, I.K. (1988) On Newton's Method and Non-discrete Mathematical Induction. *Bull. Austral. Mathematics Society*, volume 38, pp. 131-140,

68. D.G.T. Denison, B.K. Mallik, and A.F.M. Smith: Automatic Bayesian Curve Fitting. *Journal of the Royal Statistical Society*: Series B (Statistical Methodology) 60(2): 333-50

69. Yang, L., Fu, Z., Wang, D. (2022) Ground Motion Time History Simulation for Seismic Response History Analysis, Front. Earth Science, https://doi.org/10.3389/feart.2022.908498

70. Nguyen, H., Dao, N., Shin, M. (2022) Machine Learning Based Prediction for Maximum Displacement of Isolated System, *Journal of Building Engineering*, Vol. 51

71. Clough, R., Johnston, S. (1966) Effect of Stiffness Degradation on Earthquake Ductility Requirements. *Japan Earthquake Engineering Symposium*. Tokyo, pp. 195- 198

72. Sozen, M.A., Nielsen, N. (1973) Effect of Degrading Stiffness on the Response of Multistory Frames Subjected to Earthquakes, *Proceedings on the 5th World Conference on Earthquake Engineering*, pp. 1756 – 1765

73. Takeda, T., Sozen. M.A. (1970) R/C Response to Simulated Earthquakes. *Journal of Structure Division, ASCE*, Vol. 96, pp. 2557 – 2573

74. Takeda, T., Sozen, M.A., Nielsen, N. (1973) Effect of Degrading Stiffness on the Response of Multistory Frames Subjected to Earthquakes, *Proceedings on the 5th World Conference on Earthquake Engineering*, pp. 1756 – 1765.

75. Reinhorn, A., Mander, J.B., Kunnath, S. (1992) Seismic Design of Structures for Damage Control, Nonlinear Seismic Analysis, and Design of Reinforced Concrete Buildings, *Elsevier Applied Science*, ISBN 0-203-27153-X

76. Esquivel, J.C. (1992). Influence of Some Parameters on the Inelastic Earthquake Response Using Different Hysteretic Models for Reinforced Concrete, *10th World Conference in Earthquake Engineering*, ISBN 9054100605.

77. Wei, C., Zong, J. (2021). Hysteretic model of Reinforcement Based on Clough Model Constitutive Model and Considering Strength, *3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST)*, doi.10.1109/IAECST54258.2021.9695842

78. Shafei, B., Zareaian, F. (2008) Development of a Quasi-Static Loading Protocol for Displacement-Sensitive Nonstructural Building Components*, in proceedings of the 14th World Conference on Earthquake Engineering WCEE*, Beijing, China

79. Tan, Y., Dang, J., Igarashi, A., Himeno, T., Hamada, Y. (2023) Hybrid Simulation for Seismic Isolation Effectiveness Assessment of HDR Bearings at Low Temperature, *In book of Experimental Vibration Analysis for Civil Engineering Structure*, doi.10.1007/978-3-030-93236-7_26

80. Takatori Earthquake Data, (2023, May 20), In *Wikipedia*, https://www.road.or.jp/dl/tech.html

81. Goodfellow, I., Bengio, Y. (2016) A. Courville, Deep Learning, MIT Press, pp. 301

82. https://www.tensorflow.org/api_docs/python/tf/keras/losses/CategoricalCrossentropy

83. Goodfellow, I., Bengio, Y., & Cournille, A. (2016). Deep Learning (pp.117). MIT Press.

84. Goodfellow, I., Bengio, Y., & Cournille, A. (2016). Deep Learning (pp.298). MIT Press.

85. Bridge Seismic Control Design Method Draft: Seismic Isolation Structure Research Committee for Road and Bridges, pp. 5-16, 2011

86. Type 2 G2 Earthquake Data Website. https://www.road.or.jp/dl/tech.html

87. A. Chopra, Dynamics of Structures, Pearson Education, IBSN. 8131713296, 2007

88. Chen L. C., Zhu Y, Papandreou G, et al. (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation, Proceedings of the European conference on computer vision (ECCV). pp. 801-808

89. Cuong, N., Nguyen, Q., Jin, R., Jeon, C., Shim, C. (2021) BIM-based Mixed-Reality Application for Bridge Inspection and Maintenance, Journal of Construction Innovation, doi. 10.1108/CI-04-2021-0069

90. Liu, J., Dang, J. (2022) Semantic Segmentation for Bridge Corrosion Damage using Deeplabv3+ model, Annual Meeting of Japan Society of Civil Engineers

91. VGG Image Annotator, (2023, May 5), In *Wikipedia*, https://www.robots.ox.ac.uk/~vgg/software/via/
92. Realsense Lidar Camera L515 (2023, May 5), In *Wikipedia*, https://www.intelrealsense.com/lidar-camera-l515
93. Supervisely Annotation Tool (2023, May 5), In *Wikipedia*, https://supervisely.com

Appendix-A: Nonlinear Models Program

A.1 Bilinear Model

```
def Bilinear(d,dn,fn,para):
    dd=d-dn
    alf=para[0]
    k1=para[1]
    qc=para[2]
    k2=alf*k1
    fe=d*k2
    fen=k2*dn
    fpn=fn-fen

    fp=fpn+(1-alf)*k1*dd

    if fp>qc:
        fp=qc
    if fp<-qc:
        fp=-qc

    f=fp+fe
    return f
```

A.2 Modified Bilinear Model

```
def ModifiedBilinear(d,dn,fn,para):
    dd=d-dn
    alf=para[0]
    k1=para[1]
    qc=para[2]
    b=para[3]
    qc1=qc+b*d*d # b is introduced for pinching effect

    k2=alf*k1
    fe=d*k2
    fen=k2*dn
    fpn=fn-fen

    fp=fpn+(1-alf)*k1*dd

    if fp>qc1:
        fp=qc1
    if fp<-qc1:
        fp=-qc1

    f=fp+fe
    return f
```

A.3 Bouc-Wen Model

```
def BouWen(d,dn,Z,para):
    f=0
    A=1
    N=1
    alpha=para[0]
    k=para[1]
    B=para[2]
    gama=para[3]
    dy=np.sqrt(A/(B+gama))
    dd=d-dn
    eps=dy*0.01
    ddr=dd

    while np.abs(ddr)>0:
      if np.abs(ddr) < eps:
          ddz=ddr
          ddr=0
      else:
          ddz=eps*(ddr/np.abs(ddr))
          ddr=ddr-ddz
      dZ=A*ddz-Z*B*np.abs(ddz)*np.abs(Z)**(N-1)-gama*ddz*Z**N
      Z=Z+dZ

    f=alpha*k*d+(1-alpha)*k*Z
    return f,Z
```

A.4 Modified Bouc-Wen Model

```
def ModifiedBoucWen(d,dn,Z,para):
    f=0
    A=1
    alpha=para[0]
    k=para[1]
    B=para[2]
    gama=para[3]
    b=para[4]
    dy=np.sqrt(A/(B+gama)) # yield displacement
    eps=dy/100
    dd=d-dn
    ddr=dd
    while np.abs(ddr)>0:
      if np.abs(ddr) < eps:
          ddz=ddr
          ddr=0
      else:
          ddz=eps*(ddr/np.abs(ddr))
          ddr=ddr-ddz

      dZ=A*ddz-Z*B*np.abs(ddz*Z)-gama*ddz*Z*Z
      Z=Z+dZ
    f=alpha*k*d+(1-alpha)*k*(1+b*d*d)*Z
    return f,Z
```

A.5 Clough Model

```
def Clough_model(u,un,dun,fn,alf,k,qc,s,dy1,dy2,dt1,dt2,ft1,ft2):
  # s=0 :elastic, s=1 plastic
  #dy1*positiveside yiled, dy2, negative siede yield
  f=0
  du=u-un
  #print(dy2)

  if s==0:
    f=fn+du*k
    if u>dy1:
      s=1
      f1=fn+(dy1-un)*k
      f=f1+(u-dy1)*alf*k
    elif u<dy2:
      s=1
      f1=fn+(dy2-un)*k
      f=f1+(u-dy2)*alf*k
  elif s==1:
    f=fn+du*alf*k
    if dun*du<0:

      f=fn+du*k
      if du>0:
        s=4
        dy2=un
        dt2=un
        ft2=fn
        dy1=un-fn/k
      else:
        s=2
        dy1=un
        dt1=un
        ft1=fn
        #print(ft1)
        dy2=un-fn/k
```

```
elif s==2:
  f=fn+du*k
  if u>dy1:
    s=1
    f1=fn+(dy1-un)*k
    f=f1+(u-dy1)*alf*k
  elif u<dy2:
    s=3
    k3=-ft2/(dy2-dt2)
    f=k3*(u-dy2)
elif s==3:
  k3=-ft2/(dy2-dt2)
  f=k3*(u-dy2)
  if u<dt2:
    s=1
    f=ft2+(u-dt2)*alf*k
  if du*dun<0:
    s=4
    dy2=un
    dy1=un-fn/k
elif s==4:
  f=fn+du*k
  if u>dy1:
    s=5
    k3=ft1/(dt1-dy1)
    f=(u-dy1)*k3
  elif u<dt2:
    s=3
    f=ft2+(u-dt2)*alf*k
elif s==5:
  k3=ft1/(dt1-dy1)
  f=(u-dy1)*k3
  if u>dt1:
    s=1
    f=ft1+(u-dt1)*alf*k
  if du*dun<0:
    s=2
    dy1=un
    dy2=un-fn/k
return f,du,s,dy1,dy2,dt1,dt2,ft1,ft2
```

A.6 Trilinear Model

```
def Trilinear(u,un,dun,fn,alf,k,qc,s,dy1,dy2,myu,beta):
  # s=0 :elastic, s=1 plastic
  #dy1*positiveside yiled, dy2, negative siede yield
  f=0
  du=u-un
  #print(dy2)

  fy=qc/(1-alf)
  dy=fy/k
  fu=(1+myu*alf-alf)*fy
  k3=beta*k


  if s==0:
    f=fn+du*k
    if u>dy1:
      s=1
      f1=fn+(dy1-un)*k
      f=f1+(u-dy1)*alf*k
    elif u<dy2:
      s=1
      f1=fn+(dy2-un)*k
      f=f1+(u-dy2)*alf*k
  elif s==1:
    f=fn+du*alf*k
    if dun*du<0:
      s=0
      f=fn+du*k
      if du>0:
        dy2=un
        dy1=un+2*dy
      else:
        dy1=un
        dy2=un-2*dy
    elif u*du/np.abs(du)>dy*myu:
      s=2
      f=abs(du)/du*(fu+k3*(abs(u)-myu*dy))
  elif s==2:
    f=fn+k3*du
    if dun*du<0:
      s=0
      f=fn+du*k
      de=(k*un-fn)/(1-alf)/k
      #print(de)
      if du>0:
        dy2=un
        dy1=de+dy
      else:
        dy1=un
        dy2=de-dy
    if np.abs(f)/f*(f-alf*k*u)<0: # broken
      f=0
      s=3
  else: # damage status s==3
    f=0

  return f,du,s,dy1,dy2
```

Appendix-B: Nonlinear Time History Analysis Program

B.1 Structure Settings (Build Matrix, Eigen Value Analysis, and Damping Matrix)

```
def BuildMatrix(N,m,k,h):
    M=np.zeros((N,N))
    K=np.zeros((N,N))
    C=np.zeros((N,N))
    w=np.zeros(N)
    T=np.zeros(N)
    f=np.zeros(N)
    for i in range(N):
        M[i,i]=m[i] ##Mass Matrix
        K[i,i]=k[i] ##Stiffness Matrix
        if i>0:
            K[i,i]=k[i-1]+k[i]
            K[i,i-1]=-k[i-1]
        if i<N-1:
            K[i,i+1]=-k[i]
            K[i,i]= k[i]

    # Eigen Value Analysis

    Minv=np.linalg.inv(M)

    X=Minv.dot(K)
    w,v=eig(X)
    w=np.sqrt(w)
    f=w/np.pi/2
    T=1/f

    # Damping Matrix

    w1=w[-1]
    w2=w[-2]
    h1=h[0]
    h2=h[1]
    a0=2*w1*w2*(w2*h1-w1*h2)/(w2*w2-w1*w1)
    a1=2*(w2*h2-w1*h1)/(w2*w2-w1*w1)
    C=a0*M+a1*K
    return M,C,K,w,f,T
```

B.2 MDOF Bilinear Model

```
def Mdofbilinear(Ndof,fn,u,un,k, Tr, Nr, Ae, G,alf,qc):

    for i in range(Ndof):
      if i==1: #add this part
        f[i]=k[i]*u[i] #Pier Elastic (linear)
      if i==0: #Bearing, Nonlinear
        eps=u2eps(u[i],Tr) #Convert displacement to strain
        epsn=u2eps(un[i],Tr)
        sign=f2sig(fn[i],Nr,Ae) #Convert the force to shear stress (N ->
N/m2)
        sig=bilinear(sign,eps, epsn, G, alf,qc)
        f[i]=sig2f(sig,Nr,Ae)
    r=f2r(f)
    return f,r
```

B.3 Conversion of Displacement to Strain

```
def u2eps (u,Tr):
    ub = u
    eps = ub/Tr #No unit(strain)
    return eps

def f2sig (f,Nr,Ae):
    fb = f/Nr
    sig = fb/Ae #N/mm2
    return sig
```

B.4 Conversion of Shear Stress to Force

```
def sig2f (sig,Nr,Ae):
    fb = sig*Ae
    f = fb*Nr #N

    return f
```

B.5 Calculation of HDR-S Displacement

```
def d2u(d):
    Ndof=len(d)
    u=np.zeros(Ndof)
    for i in range(Ndof):
        if i==Ndof-1:
            u[i]=d[i]
        else:
            u[i]=d[i]-d[i+1]

    return u
```

B.6 Solve the Restoring Force

```
def f2r(f):
    Ndof=len(f)
    r=np.zeros(Ndof)
    for i in range(Ndof):
        if i==0:
            r[i]=f[i]
        else:
            r[i]=f[i]-f[i-1]
    return r
```

B.7 Newmark’s Beta Method

```
def initMdofSolver(Ndof,N):
    an=np.zeros(Ndof)
    vn=np.zeros(Ndof)
    dn=np.zeros(Ndof)
    un=np.zeros(Ndof)
    fn=np.zeros(Ndof)
    rn=np.zeros(Ndof)
    agn=np.zeros(Ndof)
    A=np.zeros((Ndof,N))
    V=np.zeros((Ndof,N))
    D=np.zeros((Ndof,N))
    U=np.zeros((Ndof,N))
    F=np.zeros((Ndof,N))
    R=np.zeros((Ndof,N))
    return an,vn,dn,un,fn,rn,agn,A,V,D,U,F,R


def prepareConstant(N,M,C,K,beta,dt):

    Mh=np.zeros((N,N))
    Aa=np.zeros((N,N))
    Av=np.zeros((N,N))

    Mh=M+0.5*C*dt+K*beta*dt*dt
    Aa=C*dt+0.5*K*dt*dt
    Av=K*dt

    Mhi=np.linalg.inv(Mh)
    return Mhi, Aa, Av

def incrementalstepmdofNL(M,dag,K,Mhi,Aa,Av,dt,dn,vn,an,beta):
    dFh=np.matmul(M,dag)-np.matmul(Av,vn)-np.matmul(Aa,an)
    da=np.matmul(Mhi,dFh)
    a=an+da
    v=vn+an*dt+0.5*da*dt
    d=dn+vn*dt+0.5*an*dt**2+beta*da*dt*dt
    return a,v,d

def corrector(a,v,d,f,fn,dn,K,Mhi,dt,beta):
    df=f-fn
    dd=d-dn
    dfc=df-np.matmul(K,dd)
    dac=np.matmul(Mhi,dfc)
    a=a-dac
    v=v-0.5*dac*dt
    d=d-beta*dac*dt*dt
    return a,v,d


def newmarkbetaMDOFNL(Ndof,M,C,K,dt,beta,wave,k,Tr, Nr, Ae, G,alf,qc):
    N=len(wave)
    Mhi,Aa,Av=prepareConstant(Ndof,M,C,K,beta,dt)
    an,vn,dn,un,fn,rn,agn,A,V,D,U,F,R=initMdofSolver(Ndof,len(wave))

    for i in range(N):
        ag=np.ones(Ndof)*wave[i]
        dag=ag-agn
        a,v,d=incrementalstepmdofNL(M,dag,K,Mhi,Aa,Av,dt,dn,vn,an,beta)
        u=d2u(d)
        f,r=Mdofbilinear(Ndof,fn,u,un,k,Tr,Nr,Ae,G,alf,qc)
        a,v,d=corrector(a,v,d,r,rn,dn,K,Mhi,dt,beta)
        an=a
        vn=v
        dn=d
        agn=ag
        fn=f
        rn=r
        un=u
        A[:,i]=a
        V[:,i]=v
        D[:,i]=d
        U[:,i]=u
        F[:,i]=f
        R[:,i]=r
    return A,V,D,U,F,R
```

Appendix-C: ANN Model and Training for Nonlinear Model Classification Program

C.1 Import the Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import glob
import tensorflow as tf
from keras.models import Sequential
from keras.layers import Dense
from tensorflow import keras
from tensorflow.keras.utils import to_categorical
import tensorflow.compat.v2 as tf
```

C.2 Loading of Data

```
BL=pd.read_csv('Demo.csv')
MBL=pd.read_csv('Demo.csv')
BW=pd.read_csv('Demo.csv')
MBW=pd.read_csv('Demo.csv')
CL=pd.read_csv('Demo1.csv')
TR1=pd.read_csv('Demo.csv')
TR2=pd.read_csv('Demo.csv')

Y_train=np.vstack((BL[:a],MBL[:a],BW[:a],MBW[:a],CL[:a],TR1[:a],TR2[:a]))
Y_test=np.vstack((BL[a:b],MBL[a:b],BW[a:b],MBW[a:b],CL[a:b],TR1[a:b],TR2[a
:b]))
Xtrain=np.concatenate((c[:a],d[:a],e[:a],f[:a],g[:a],h[:a],i[:a]),axis=0)
X_train=np.vstack(Xtrain)
Xtest=np.concatenate((c[a:b],d[a:b],e[a:b],f[a:b],g[a:b],h[a:b],i[a:b]),ax
is=0)
```

```
X_test=np.vstack(Xtest)
```

C.3 ANN Training

```
model = Sequential()
model.add(Dense(100, activation='relu', input_dim=120))
model.add(Dense(50, activation='relu'))
model.add( Dense(7, activation='softmax'))
model.compile(
    optimizer = tf.keras.optimizers.Adam(learning_rate=0.001),
    loss = 'categorical_crossentropy', #(classess that are greater than 2)
    metrics = ['accuracy'],
)
model.summary()

history=model.fit( Y_train,
      to_categorical(X_train),
      validation_split=0.1,
      epochs=50,
      )

model.evaluate(
    Y_train,
    to_categorical(X_train)
)
```

C.4 Accuracy and Loss Visualization

```
history_dict=history.history
loss_values=history_dict['loss']
acc_loss_values=history_dict['accuracy']
plt.plot(loss_values,label='training loss',color='black')
plt.plot(acc_loss_values, 'r', label='training accuracy',color='black')
```

```
val_loss_values=history_dict['val_loss']
val_acc_loss_values=history_dict['val_accuracy']
plt.plot(val_loss_values,label='validation
loss',color='black',linestyle='dashed')
plt.plot(val_acc_loss_values, 'b', label='validation
accuracy',color='black',linestyle='dashed')
#plt.legend( fontsize=14)
plt.ylabel('Accuracy and Loss', color='black', fontsize=18)
plt.xlabel('Epoch',color='black', fontsize=18)
```

C.5 Prediction and Confusion Matrix

```
predictions = model.predict(Y_train)
pred_cls=np.argmax(predictions,axis=1)

true_cls=X_train[:,0]

print(pred_cls,true_cls)

from sklearn.metrics import confusion_matrix as cfsmx
mycm=cfsmx(true_cls,pred_cls)
print(mycm)

import itertools
def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion matrix, without normalization')

    print(cm)

    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45, fontsize=16)
    plt.yticks(tick_marks, classes,fontsize=16)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.rcParams["font.family"] = "serif"
    plt.rcParams["font.serif"] = "Times New Roman"
    plt.ylabel('True label',fontsize=20)
    plt.xlabel('Predicted label',fontsize=20)
    plt.savefig('confutionmatrix.jpg')

plot_confusion_matrix(mycm,classes =
["BL","MBL","BW","MBW","CL","TR1","TR2"])
```

Appendix-D: ANN Model and Training for Nonlinear Parameter Identification Program

D. 1 Import the Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import copy
import random
import math

import glob
import tensorflow as tf
from keras.models import Sequential
from keras.layers import Dense
from tensorflow import keras
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()
```

D.2 Read the Strain Values

```
Dn = []
with open('250D.txt', 'r') as f:
    Di = f.readlines()
    for i in Di:
      di = i[:-1]
      Dn.append(di)
Ds = np.array(Dn, dtype=float)
print(Ds.dtype)
```

D.3 Data Generation with Random Parameter's Selection

```
Alpha=[]
K=[]
Bi=[]
G=[]
```

```
I=[]
bi=[]

al=25 #Alf lower limit
au=35 #Alf upper limit
kl=50 #K1 lower limit
ku=80 #K1 upper limit
bl=31 # B lower limit
bu=37 # B upper limit
gl=-30 # QC lower limit
gu=-27 # QC upper limit
bil= 35
biu= 40

with open('output.txt', 'a') as f:
    for i in range(0,1000):  # Number of Samples
    # Random Parameters with Specific Range
        alpha=random.randint(al,au)/100
        k=random.randint(kl,ku)/100
        B=random.randint(bl,bu)/10
        gama=random.randint(gl,gu)/10
        b=random.randint(bil,biu)/100

        simsig0=SimByMPW(Ds,alpha,k,B,gama,b)

        df=pd.DataFrame(simsig0)
        Alpha.append(alpha)
        K.append(k)
        Bi.append(B)
        G.append(gama)
        bi.append(b)

        M=np.concatenate((Ds, simsig0), axis=0)
        N=np.vstack((M))
        P=N.T
        I.append(P)
        np.savetxt(f,P)
        print(Alpha[i],K[i],Bi[i],G[i],bi[i])

        #Save Images in every loop
        plotresult(Ds, simsig0)
```

D.4 ANN Training

```
a=900
b=1000
Y_train=Y[:a]
Y_test=Y[a:b]
X_train=X[:a]
X_test=X[a:b]

e=900
f=400

model = tf.keras.models.Sequential()
model.add(tf.keras.layers.Dense(e, input_dim=120, activation='relu'))
model.add(tf.keras.layers.Dense(f,activation='relu'))
model.add(tf.keras.layers.Dense(5))

optimizer = tf.train.RMSPropOptimizer(0.001)

model.compile(loss='mse',
              optimizer=optimizer,
              metrics=['mae'])

model.summary()

history=model.fit(Y_train, X_train, validation_split=0.1, epochs=100)
```

D.5 Loss Visualization

```
score=model.evaluate(Y_test, X_test)
score=model.evaluate(Y_train, X_train)

history_dict=history.history
loss_values=history_dict['loss']
val_loss_values=history_dict['val_loss']
plt.plot(loss_values,label='training loss')
plt.plot(val_loss_values, 'r', label='training loss val')
plt.xlabel("Epochs", color='black', fontsize='18')
plt.ylabel("Loss",color='black', fontsize='18')
plt.legend()
```

D.6 ANN Parameter Prediction

```
g=20
pred=model.predict(Y_test[:g])
print(pred)
print(X_test[:g]) # True Labels
```

D.7 Experiment Data Nonlinear Parameter Prediction

```
Dn = []
with open('D.txt', 'r') as f:
    Di = f.readlines()
    for i in Di:
      di = i[:-1]
      Dn.append(di)
Ds = np.array(Dn, dtype=float)
print(Ds.dtype)

Fn = []
```

```
with open('250F.txt', 'r') as f:
    Fi = f.readlines()
    for i in Fi:
      fi = i[:-1]
      Fn.append(fi)
Fs = np.array(Fn, dtype=float)
print(Fs.dtype)

M=np.concatenate((Ds,Fs), axis=0)
N=np.vstack((M))
I=N.T
pred=model.predict(I)
print(pred)
```

D.8 Contribution Rate Calculation

```
def ContributionRate(Fex,Fax):
    num=0
    den=0
    FeA=np.average(Fex)
    for fe,fa in zip(Fex,Fax):
        num=num+(fe-fa)**2
        den=den+(fe-FeA)**2
    R2=1-num/den
    if R2<0:
        R2=0
    R=np.sqrt(R2)
    return R

R=ContributionRate(Fs,simsig1)
print(R)
```

Appendix-E: Bridge Seismic Isolation Design Program

E.1 Input Earthquake

```
import os
os.chdir("/content/drive/MyDrive/EQ Data")

sheets=['L2T2']
filename='/content/drive/MyDrive/EQ Data/level2-type2.xls'

wave1=np.array([])

for sheet in sheets:
    df = pd.read_excel(filename, sheet_name=sheet)
    df_value=df.values
    D=df_value[1:2001,4]
    ag=D/100
    wave1=np.append(wave1,ag)


dt = 0.01
Fs = 100
N = len(wave1)
t = np.arange(N)/Fs
print(wave1.shape)
print(N)
print(t.shape)

plt.plot(t,wave1,color='black')
plt.title("Input Wave",fontsize=20)
plt.xlabel("Time(s)",fontsize=18)
plt.ylabel("Acceleration (m/ss)",fontsize=18)
plt.show()
plt.show()
```

E.2 Initial Parameters

```
#Superstructure and HDR-S

VP =[]    #Vertical Pressure
BR = []   #HDR-S Cross Section
M1 =[]    #Mass of the Super-structure
TEI=[]    #Thickness of the Rubber
NR=[]     #Number of Rubber Layers
TR= []    #Total Thickness of Rubber
S1R=[]    #Primary Shape Factor
S2R=[]    #Secondary Shape Factor

# Pier

KP= []      #Pier's Stiffness
M2=[]      # Mass of the Pier

#Nonlinear Parameters

G1=[]
G2=[]
QCi=[]
ALF=[]

K11=[] #N/mm
K22=[]  #N/mm
KE =[]
QC=[]


#Structure's Response

UMAX = []       #Maximum Displacement
U0MAX = []      #Maximum Strain of the HDR-S
T01=[]          #Period of the Structure
T02=[]          #Period of the Structure
```

```
Nr = 6        #Number of the HDR-S Bearing (pcs)
vpl= 6        #MPa
vpu= 9        #MPa
brl= 60       #cm
bru= 120      #cm
tel= 15       #mm
teu= 40       #mm
nel = 6       #pcs
neu = 11      #pcs
```

E.3 HDR-S Bearing Generation

```
def generate_bearing (Nr,vpl,vpu,brl,bru,nel,neu,tel,teu):
  while True:
    vp = np.random.randint(vpl,vpu)        #vertical pressure MPa N/mm2
    br=np.random.randint(brl,bru)*10       #cross sectional width of the 
rectangular bearing mm
    Aei=br*br                              #Cross sectional Area
    wb = Aei*vp                            #assumed weight of the 
superstructure Kg

    m1 = wb/9.81/Nr                       #mass of the superstructure kg
    m2= m1/5                             #mass of the pier kg

    ni = np.random.randint(nel,neu)       #No. of rubber in pcs
    Tri=np.random.randint(tel,teu)        #HDR-S Rubber thickness in mm
    Tr= Tri*ni

    #Check the Shape Factor
    s1R= Aei/(2*2*br*Tri)               #Primary Shape Factor > 6.0
    s2R = br/Tr                         #Secondary Shape Factor > 4.0
    #print (s1R,s2R)
    if s1R>6.0 and s1R<12.0 and s2R>4.0 and s2R<8.0:
      break
  return vp,br,Aei,wb,m1,m2,ni,Tri,Tr,s1R,s2R
vp,br,Aei,wb,m1,m2,ni,Tri,Tr,s1R,s2R = generate_bearing
(Nr,vpl,vpu,brl,bru,nel,neu,tel,teu)
print(vp,br,Aei,wb,m1,m2,ni,Tri,Tr,s1R,s2R )



def set_paras(g1,g2,ge,Tr,Aei):
  td= ub*(ge-g2)
  ty=td*(g1/(g1-g2))

  qc = ty
  alf= g2/g1

  k11=g1*Aei/Tr
  k22=g2*Aei/Tr
  qcc=qc*Aei

  K1= k11*1000                            #stiffness of the bearing N/m
  K2 =K1*(3+np.random.random()*1)         #Pier's Stiffness  N/m

  #Newmarks Beta Method
  beta = 1/6 #New Marks Beta Methdod
  Ndof = 2 # 2 degree of freedom
  h = 0.05 # Damping

  m = [m1,m2]
  k=[K1,K2]
  h=[h,h]
  return m,k,h,Ndof,beta,k11,k22,qcc,alf,qc,K1,K2

def get_Gs(gamma):
  r=gamma
  a=np.array([35.13,-54.83,43.51,-16.44,3.019,-0.2114])
  b=np.array([3.345,-5.221,4.144,-1.565,0.2872,-0.0201])
  c=np.array([5.128,-7.971,6.227,-2.331,0.4162,-0.02762,])
  rs=[]
  for i in range(6):
    rs.append(r**i)

  g1 = np.dot(a,rs)
  g2 = np.dot(b,rs)
  ge = np.dot(c,rs)
  return g1, g2, ge
```

E.4 Nonlinear Time History Analysis Simulation

```
from numpy.lib.shape_base import kron

eps0=0.03
N_case=5

for i in range(0,N_case):  # Number of Samples

  vp,br,Aei,wb,m1,m2,ni,Tri,Tr,s1R,s2R=generate_bearing
(Nr,vpl,vpu,brl,bru,nel,neu,tel,teu)

  VP.append("%.3f"%vp)
  BR.append("%.3f"%br)
  TR.append(Tr)
  TEI.append(Tri)
  NR.append(ni)
  S1R.append(s1R)
  S2R.append(s2R)

  ubi = 2.5

  while True:
    ub= ubi
    g1,g2,ge= get_Gs(ub)

    m,k,h,Ndof,beta,k11,k22,qcc,alf,qc,K1,K2= set_paras(g1,g2,ge,Tr,Aei)

    M,C,K,w,f,T=BuildMatrix(Ndof,m,k,h)


A,V,D,U,F,R=newmarkbetaMDOFNL(Ndof,M,C,K,dt,beta,wave1,k,Tr,Nr,Aei,g1,alf,
qc)

    Umax1=max(abs(U[0,:]*1000)) #maximum displacement mm

    U0max1=(Umax1)/Tr #maximum strain

    ubi=U0max1

    if ubi > 4:
      break
    error=np.abs(ubi-ub)
    print('erro=%.5f,ub=%.5f,ubi=%.5f,umax1=%.5f'%(error,ub,ubi,Umax1))
    if error<eps0:
      break

  UB1=Umax1

  ubi = 2.5

  while True:

    ub= ubi
    g1,g2,ge= get_Gs(ub)

    m,k,h,Ndof,beta,k11,k22,qcc,alf,qc,K1,K2= set_paras(g1,g2,ge,Tr,Aei)

    M,C,K,w,f,T=BuildMatrix(Ndof,m,k,h)
```

```
A,V,D,U,F,R=newmarkbetaMDOFNL(Ndof,M,C,K,dt,beta,wave2,k,Tr,Nr,Aei,g1,alf,
qc)

    Umax1=max(abs(U[0,:]*1000)) #maximum displacement mm

    U0max1=(Umax1)/Tr #maximum strain

    ubi=U0max1

    if ubi > 4:
      break
    error=np.abs(ubi-ub)
    print('erro=%.5f,ub=%.5f,ubi=%.5f,umax1=%.5f'%(error,ub,ubi,Umax1))

    if error<eps0:
      break

  UB2=Umax1
  ubi = 2.5

  while True:
    ub= ubi
    g1,g2,ge= get_Gs(ub)

    m,k,h,Ndof,beta,k11,k22,qcc,alf,qc,K1,K2= set_paras(g1,g2,ge,Tr,Aei)

    M,C,K,w,f,T=BuildMatrix(Ndof,m,k,h)


A,V,D,U,F,R=newmarkbetaMDOFNL(Ndof,M,C,K,dt,beta,wave3,k,Tr,Nr,Aei,g1,alf,
qc)

    Umax1=max(abs(U[0,:]*1000)) #maximum displacement mm
    U0max1=(Umax1)/Tr #maximum strain

    ubi=U0max1

    if ubi > 4:
      break
    error=np.abs(ubi-ub)
    print('erro=%.5f,ub=%.5f,ubi=%.5f,umax1=%.5f'%(error,ub,ubi,Umax1))

    if error<eps0:
      break

  UB3=Umax1

  G1.append("%.3f"%g1)
  G2.append("%.3f"%g2)
  QCi.append("%.3f"%qc)
  ALF.append("%.3f"%alf)

  K11.append("%.3f"%k11)
  K22.append("%.3f"%k22)
  KE.append("%.3f"%K1)
  QC.append("%.3f"%qcc)
  KP.append("%.3f"%K2)

  M1.append("%.3f"%m1)
  M2.append("%.3f"%m2)

  UB=(UB1+UB2+UB3)/3
```