WILEY | Hindawi

Research Article

Hierarchical Neural Network and Simulation Based Structural Defect Identification and Classification

Divya Shyam Singh [ID]**, G. B. L. Chowdary, and D. Roy Mahapatra** [ID]

Department of Aerospace Engineering, Indian Institute of Sciences, Bangalore 560012, India

Correspondence should be addressed to D. Roy Mahapatra; roymahapatra@iisc.ac.in

A vibration data-driven structural defect identification and classification technique is developed using frequency response under random excitation and a hierarchical neural network. A system of artificial neural networks (ANNs) is trained using finite element simulation-based synthetic data to reduce the need for many sensor measurements required otherwise. Principal component analysis (PCA) is employed to compress the high dimensionality of the vibration response data and eliminate the noise effect in the training and testing. Frequency responses data dimension for the structure with defects such a crack from stress concentration, rivet hole expansion, and attached foreign object mass such as ice accumulation in aircraft wing or fuselage are reduced using PCA and fed to a classifier network. The probabilistic decision output from the classifier network and the compressed data are then fed to the next levels of estimator networks, where each network is dedicated to the individual type of defect for the estimation of the defect parameters corresponding to that class of defect. The methodology is applied to a stiffened panel structure. The cracks and rivet hole expansions are introduced in the rivet line of the stiffener, and the foreign object mass is attached to the panel surface. The results show that it is possible to classify the defects and further estimate the defect parameters with good accuracy and reliability. It was observed that the damage classification network had an accuracy of roughly 95%. The damage localization network for crack as well as rivet expansion had average absolute error of around 2. The damage severity network was also able to perform well with a mean absolute error of about 0.34 for crack length detection and 0.22 for expanded rivet damage. However, the damage localization and severity prediction networks were quite challenging to train in the presence of multiple damages and need further development in the network architecture.

1. Introduction

Defect and damage detection in mechanical structures have emerged as an important area of research. This is considering the requirements of structural health monitoring (SHM) for next-generation aerospace, marine, and civil structures. Here, we use the term defects as a general term where damage is a particular type of irreversible change induced by a loading process resulting in the performance degradation or measurable change in the material and or geometry. Increasing emphasis is being placed on damage-tolerant structural design with improved safety standards through better manufacturing, quality control, and a paradigm shift from purely off-line inspection to on-line monitoring with integrated sensors [1, 2]. Understanding the structural response under various loads is central to this idea beyond the load-case analysis that is used mostly during the design. The vibration-based defect identification approach reviewed by Doebling et al. [3] pointed out the modern developments toward vibration-based damage detection. The approaches evolved from the initial studies in the off-shore oil industry [4]. The damage detection methods in mechanical structures can broadly be divided into two main groups. One employs physics-based models, and the other employs numerical/measured data-driven models. Both methods aim to find changes in the structural response and determine the change in the system properties such as stiffness, mode shape, and frequency change [5].

With the advancement of machine learning-based algorithms, damage detection in structures is also carried out

with the help of vision-based systems [6, 7]. Vision-based system has been used for crack detection in concrete using SDDNet [8], real-time detection using fast CNN [9], encoder-decoder based crack detection [10], crack detection in roads [11, 12] along with the emergence of several neural networks focusing on crack detection such as CrackNet [13–15] etc. However, this type of monitoring approach cannot be applied to structures with damages where placing a camera might not be feasible or with damages which are not easily visible from outside. For such systems, vibration-based approach is more suited that employs methods for extracting information based on changes in the vibration data to determine the damage parameters. Moreover, these are generally more easier to place (such as embedded sensors [16]) as well as a cheaper alternative.

Vibration-based damage detection technique often converts the vibration data into the frequency domain. Frequency response is easiest to obtain in real time with a moving time window as they require only a small number of sensors at locations having a high sensitivity (observability) to the probable damages [17]. Numerous different frequency response-based algorithms to identify damage have been reported in the literature and evaluated in practical applications. Useful results have been obtained in damage identification from numerically simulated frequency response data [18]. The literature related to damage detection using the shifts in the natural frequency is also quite extensive [19, 20]. However, there are practical limitations of applying frequency change as a damage indicator when the mode shapes around the excitation frequencies are not sensitive to the local distortion in the displacement or strain field due to localized damage as well [21–23]. Unexpected changes in the natural frequencies are often attributed to the additional mechanisms of local stiffening, material hardening, and nonlinear harmonics produced due to contact dynamics [24, 25]. The use of a higher-order gradient beyond the order of strain shows some advantages in amplifying the localizing damage fingerprint, which is otherwise not observable [26, 27]. This is helpful in a theoretically ideal situation. However, this could be limited by the observability or the spatial resolution of the reconstructed field from sensor data or the fidelity of a simulation, and sometimes can create spurious results due to higher-order spatial derivative calculations or spectral decomposition (e.g., FFT or wavelet transformation) [28]. Using such response data in a machine learning process may deteriorate the prediction capabilities unless special strategies based on physics-based criteria on filtering/reducing the data are evolved [29, 30]. In summary, the use of entire frequency response spectra directly along with an effective dimensional reduction of the data to eliminate unrelated features and remove noise/artifacts can have an advantage in machine learning-based damage detection strategies.

Formulating an analytical model with an acceptable level of accuracy often requires considerable effort, making such approaches less attractive. Data-driven methods for damage detection have increasingly become a practical choice as it is easier to circumvent the complexity of physics model-driven processes. Another advantage of choosing a data-driven approach is that it can be easily automated. Over the years, there has been a substantial effort toward developing structural health monitoring (SHM) algorithms for mechanical structures. Several techniques have been applied for damage detection, classification, and parameter estimation, such as support vector machines [31–34], neural networks [35–42], and hierarchical neural networks [43–45]. Data compression strategies along with hierarchical networks have also been reported [46–48].

The neural networks must be first trained using training datasets. In most real-world applications, training data is hard to make available initially, and the input to the system is complicated in nature and contains a lot of uncertainties. The input cannot be measured at every location except for a few points where the sensors are placed. The system in a real-world situation is generally complex in nature. The system might require to be reduced, and system properties might have to be assumed in a manner they can be represented at par with synthetic data or model data. Thus, the model order reduction is an essential aspect of representing system parameters in the context of a live dataset. Also, the data collected using sensors such as accelerometers and strain gauges may contain a lot of noise. Because of the variabilities in the manufacturing process, the system response will have certain statistical characteristics. Thus, collecting highly accurate live data for the training of neural networks is cumbersome.

Synthetic data can be produced through numerical simulation, controlled experiments, field trials, or using machine learning algorithms. However, in the controlled environment of a laboratory, the difficulty to control the damage parameters, the cost, and the time required to gather adequate training data makes these methods unfeasible. Synthetic data generation through finite element software or analytical formulation is cost-effective, and the damage parameters are much easier to control. With the improvement in computational power in recent years, it is easier to generate artificial data for several damage cases in a short period of time [49–52]. In the recent years, generative adversarial networks (GAN [53]) have gained a lot of popularity in generating synthetic data, and there have been studies [54, 55] on the possibility of using GANs to generate synthetic data. They deal with topics such as generating synthetic images for damage detection concrete structure [56], for road damage detection [57], generation of vibration data for damages [54], multiclass detection of damage in civil structures [58], regeneration of lost data to make up for sensor malfunctions [59], and detection of anomalous data from malfunctioning sensors [60]. GANs are still under active research, and their applicability is limited by the fact that they need real data to generate more synthetic data. So, they cannot be applied in areas where it is not possible to carry out experiments due to the high costs involved. Also, there is only a scarce amount of recorded data which is publicly available for the researchers that make the use of GANs quite restricted.

In our present study, the proposed damage identification method is based on a simulated frequency response generated. The synthetic data are generated from finite

element software, and the process of generation of training data for multiple damage cases was automated from MATLAB code. PCA has been used to reduce the frequency response data dimension. A further challenge in utilizing frequency response data for damage classification and identification is the influence of measurement points [61]. In our present study, we assumed all the damages are likely to develop around the stiffener, and sensors have been placed in the vicinity of these points. A random Gaussian white noise signal is used as a vibration input force signal to simulate a typical aircraft fuselage stiffened panel vibration in real time. The number of such different random input signals is limited to ten to limit computational time for generation of the synthetic data. The central idea of this work is the hierarchical network of ANNs which has not been reported in the literature so far. There are many different methods when it comes to the individual ML algorithms for damage detection, and there are comparisons of these ML algorithms in the literature as well. Most the papers in this context deal with either damage detection [62–64] or damage localization/severity prediction [65] aspect individually. The novelty of this paper is to propose a way of organizing simple ANN networks in a way that they can detect multiple damage types followed localization and severity prediction altogether within a unified framework. There are a total of six neural networks used in the current framework, some of which are regression networks, and some are classifiers. At first, the compressed data are fed to a damage classifier network to classify the type of damage (rivet loosening, crack, and mass addition). Based on the damage classification, the frequency response data are fed to the respective identifier network to estimate the location and severity of the probable damage class. Thereby, a more robust and improved damage identification scheme is obtained.

In the next section, the modelling of the system and the scheme used for generation of synthetic data along with data compression technique is discussed. The third section discusses in detail about the architecture of the hierarchical system of neural network used, their configuration, and training and testing aspect of each of them. The fourth section presents the results and discussion of the current work followed by the conclusion.

2. Modelling and Simulation Setup

A stiffened Aluminum panel (Figure 1) geometry is discretized using the finite element meshing software Hypermesh. The panel of dimension 1500 mm × 1150 mm × 2 mm has an L-angle placed halfway and riveted with the help of 34 rivets. We used CQUAD4 shell finite elements for the mesh and rigid 1D element (RBE3 connector) for modelling rivet connection around the holes in the plate and the L-angle. The plate is assumed to be on all-edge fixed support. The panel was excited at the center through 10 different random excitations generated through MATLAB. The acceleration and strain values from 8 nodal locations (sensor locations) on the panel were used for machine learning.

2.1. Damage Modelling.

We consider three cases of damage in the present study: crack emanating from a rivet hole, rivet hole expansion, and mass addition on the panel. The crack was simulated by releasing the element edge nodes along the crack path. For simulating the rivet hole expansion, the diameter was changed by mesh morphing. In these ways, the base model was kept the same.

Multiple rivet hole expansion was also simulated using the same way. Mass addition (e.g., ice formation) was simulated by calculating the equivalent thickness of aluminum which would have equal mass as that of mass added on the panel and adding to the panel thickness (mass addition has been assumed to take place on the complete area of the panel).

2.2. Generation of Synthetic Data.

A function was written in MATLAB, which automated the generation of synthetic data. The MATLAB function fed the value of the damage parameter corresponding to the damage case into Hypermesh, following which the model was updated accordingly. This updated model was analyzed in finite element solver MSC NASTRAN, and the generated output data were exported back into MATLAB environment. The simulation output data consisted of the acceleration and strain at the preselected node location, and the corresponding frequency response function was computed. Each simulation took about 40 minutes of CPU time to run in a quad core Xeon processor. This was repeated for all the damage cases for ten sample random force signatures as listed in Table 1 with the run time for each damage case. The overall time required to generate the complete simulation data was 730 hours. The location of nodes from which the acceleration and strain data were collected was chosen based on expected sensitivities by analyzing the finite element data. A flowchart of the synthetic data generation steps is shown in Figure 2.

For the stiffened panel structure, full-size frequency response data, which covers, for example, a frequency range of 0 to 1000 Hz, contain 2,00,000 frequency samples. This would mean 2,00,000 input nodes in the neural network for each measurement point. Such large numbers of input points cause severe problems in training convergence. Therefore, PCA is applied to the damage fingerprints to reduce size and filter noise.

2.3. PCA of the Frequency Response Data.

The large size of frequency response data is a major obstacle for neural network training. In order to use only a few damage parameters that are feasible for neural network training, the input data dimension must be greatly reduced. In the current work, principal component analysis has been used. Principal component analysis (PCA) [66, 67] is a statistical procedure that identifies the principal directions or basis space in which the data vary. A useful application of PCA is reducing the problem's dimensionality in cases where eigenvalues spread over a wide range. It allows identifying the principal directions in which the data vary. Principal components are entirely equivalent to finding the eigenvectors of the covariance matrix. There is a principal component
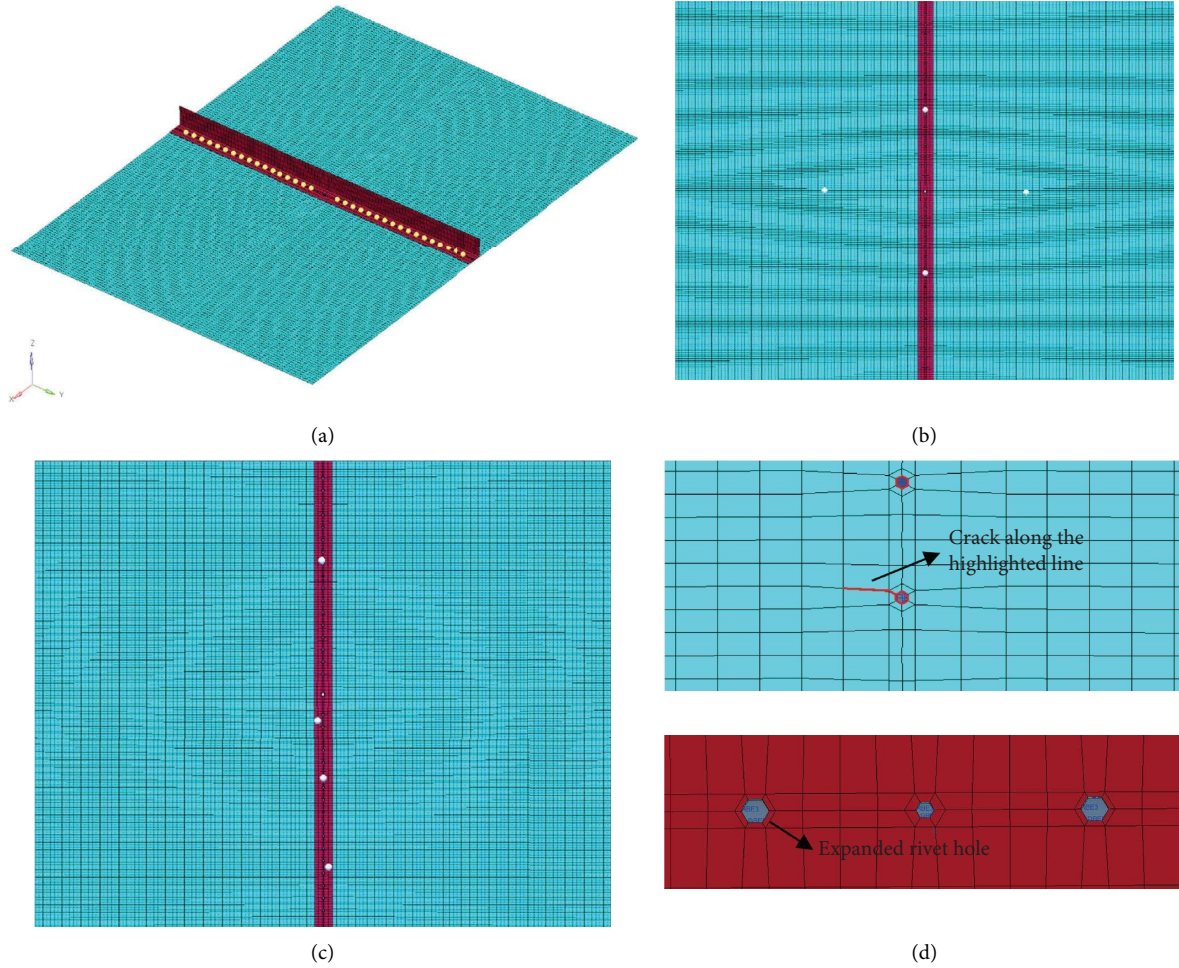
(a)

(b)

(c)

(d)

Figure 1: (a) Finite element model of a stiffened panel, (b) nodal points for collection of the acceleration signal, (c) nodal points for collection of strain signal, and (d) crack emanating from rivet hole and expanded rivet hole.

Table 1: Specification of damage scenarios.

Damage case	Number of simulation samples	Damage parameter set	Computational runtime (hours)
Rivet expansion	340	17 locations × 2 diameters × 10 random input signals	227
Crack	680	34 locations × 2 lengths × 10 random input signals	454
Uniform mass addition	60	6 variations in thickness × 10 random input signals	40
Healthy panel	10	10 random input signals	7

corresponding to every eigenvalue. The corresponding eigenvalue depicts the variance of that principal component, and the principal components with the largest variances are the most important (Figure 3).

Using PCA, the original set of variables in an M-dimensional space can be transformed into principal components in a P-dimensional space ($P < N$) by considering only the first few eigenvalues. Let $H(\omega)$ denote the $n \times m$-dimensional frequency response matrix where n is the number of samples and m is the number of variables that is already centered. To calculate and reduce the data using PCA, we first calculate the correlation matrix. We can define a correlation matrix of size $m \times m$ such that

$$\mathbf{C} = \frac{1}{(n-1)}\mathbf{H^{T}H}. \tag{1}$$

The frequency response matrix $\mathbf{H}$ can then be decomposed into the singular value decomposition as follows:

$$\mathbf{H} = \mathbf{U\Lambda V^{T}}, \tag{2}$$

where $\mathbf{U}$ and $\mathbf{V}$ are matrices of eigenvectors and are orthogonal, and Λ is a diagonal matrix with eigenvalues λ_i arranged in descending order. Putting this back into (1), we obtain
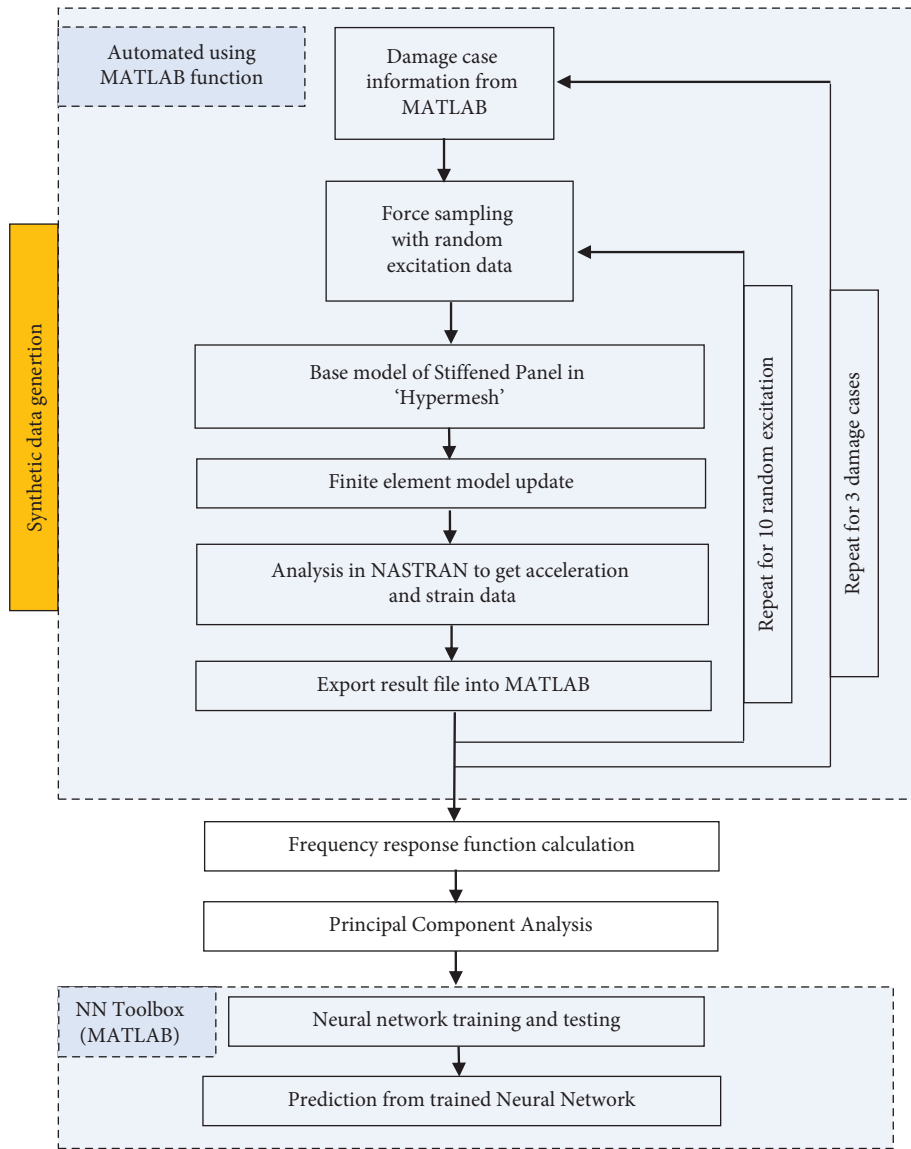
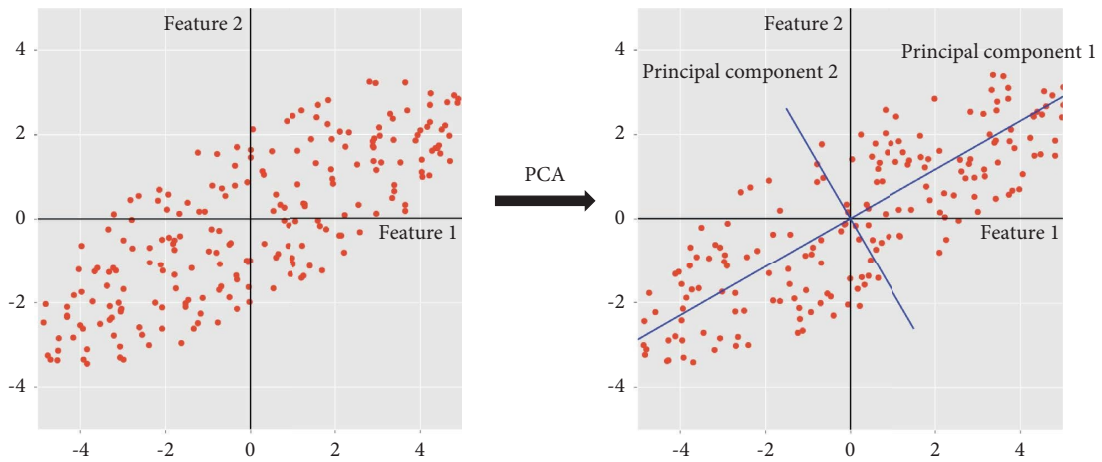FIGURE 2: Flowchart of frequency response data-driven damage detection scheme.

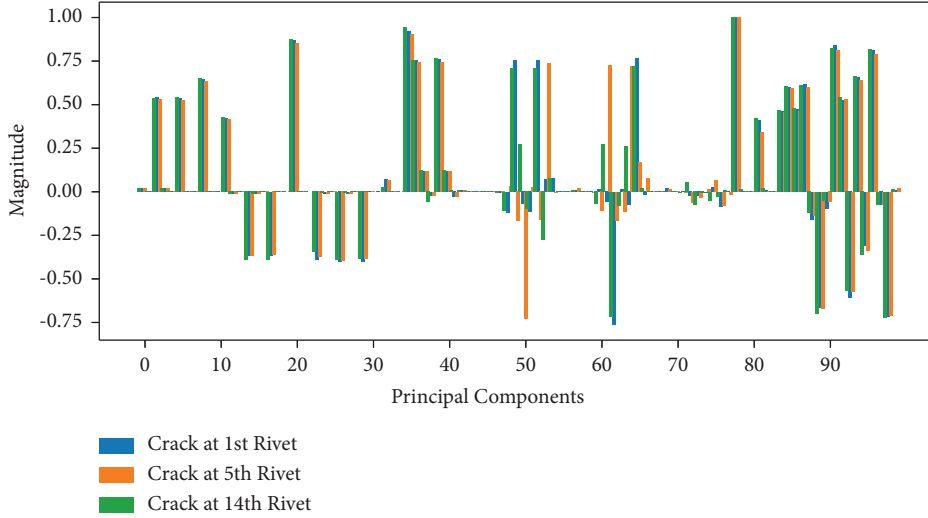FIGURE 3: PCA graphical representation.

FIGURE 4: Principal components for crack at different rivet locations.

$$\mathbf{C} = \frac{1}{(n-1)}\left(\mathbf{U\Lambda V^T}\right)^T\left(\mathbf{U\Lambda V^T}\right)$$

$$= \frac{1}{(n-1)}\mathbf{V\Lambda^2 V^T}. \tag{3}$$

The matrix Λ^2 contains the eigenvalues at the diagonal positions. To extract the first p principal components out of the total m principal components, the first p columns of $\mathbf{U}$ and the first $p \; x \; p$ entries of the matrix Λ can be selected. The final PCA matrix can be written as follows:

$$\mathbf{P} = \mathbf{U_P\Lambda_P}. \tag{4}$$

The original signal can also be reconstructed from reduced principal components as follows:

$$\mathbf{X_P} = \mathbf{U_P\Lambda_P V_P^T}. \tag{5}$$

The "*princomp*" function in MATLAB is utilized to project the damage fingerprints onto their principal components. The number of measurement points on the stiffened panel is eight, which includes four triaxial accelerometers and 4 unidirectional strain gauges. Triaxial accelerometer gives translational acceleration at the point of measurement in all three Cartesian directions, and unidirectional strain gauges measure strain in the direction of its length. Therefore, the total no of measurement points sums up to 16 (12 accelerations, 4 strains). For the damage scenario, the measurement matrix would be of dimension $2,00,000 \times 16$. Such a huge amount of data is reduced into 100×1 damage vector with the use of PCA. The most significant Eigenvalues and corresponding eigenvectors are considered, and the remaining values are discarded. The first seven principal values contribute to 99.999% of the variability in the data, thus projecting huge measurement data of dimension $2,00,000 \times 16$ onto just 100 principal components, where the first 84 components relate to the acceleration, and the last 16 correspond to strain. Figure 4 shows the principal components for crack at different rivet

locations. The differences are clearly observable. Figure 5 shows principal components variation for different types of damage.

3. ANN Architecture and Computational Scheme

The ANN-related theoretical details are reported in the literature [68]. The most popular class of multilayer feed-forward neural networks is the multilayer perceptron, in which each computational unit employs either the threshold function or the sigmoid function. Multilayer perceptron [69] can form arbitrarily complex decision boundaries and represent any Boolean function. The development of the backpropagation learning algorithm to determine weights has made these networks the most popular. Figure 6 shows the proposed perceptron network for damage detection (with classification) and damage identification (with parameter estimation). For each data category, the input samples are divided into three sets: training, validation, and testing datasets. This is done according to a partitioning system termed chessboard selection. Each damage case is divided into sets according to Table 2. While the network adjusts its weights from the training samples, its performance is supervised by utilizing the validation set to avoid overfitting. The network training stops when the validation set error reaches a minimum and begins to increase. At this point, while the error of the training set continues to decrease, the network's generalization ability is lost, and overfitting occurs.

The sample details for each damage category type are listed in Table 3. Before each dataset is presented to the network, the input data are scaled to range from −1 to +1. Data scaling is important to ensure that the distance measure accords equal weight to each sample. It also helps the network to converge in a shorter time. Furthermore, the input and output data must comply with the transfer function of the hidden layer and output layer. For this study, the hyperbolic tangent sigmoid transfer function, which
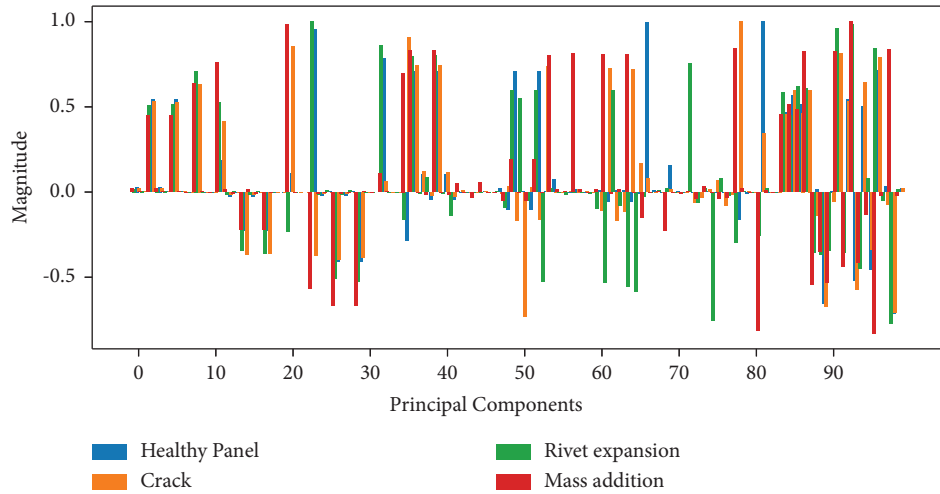
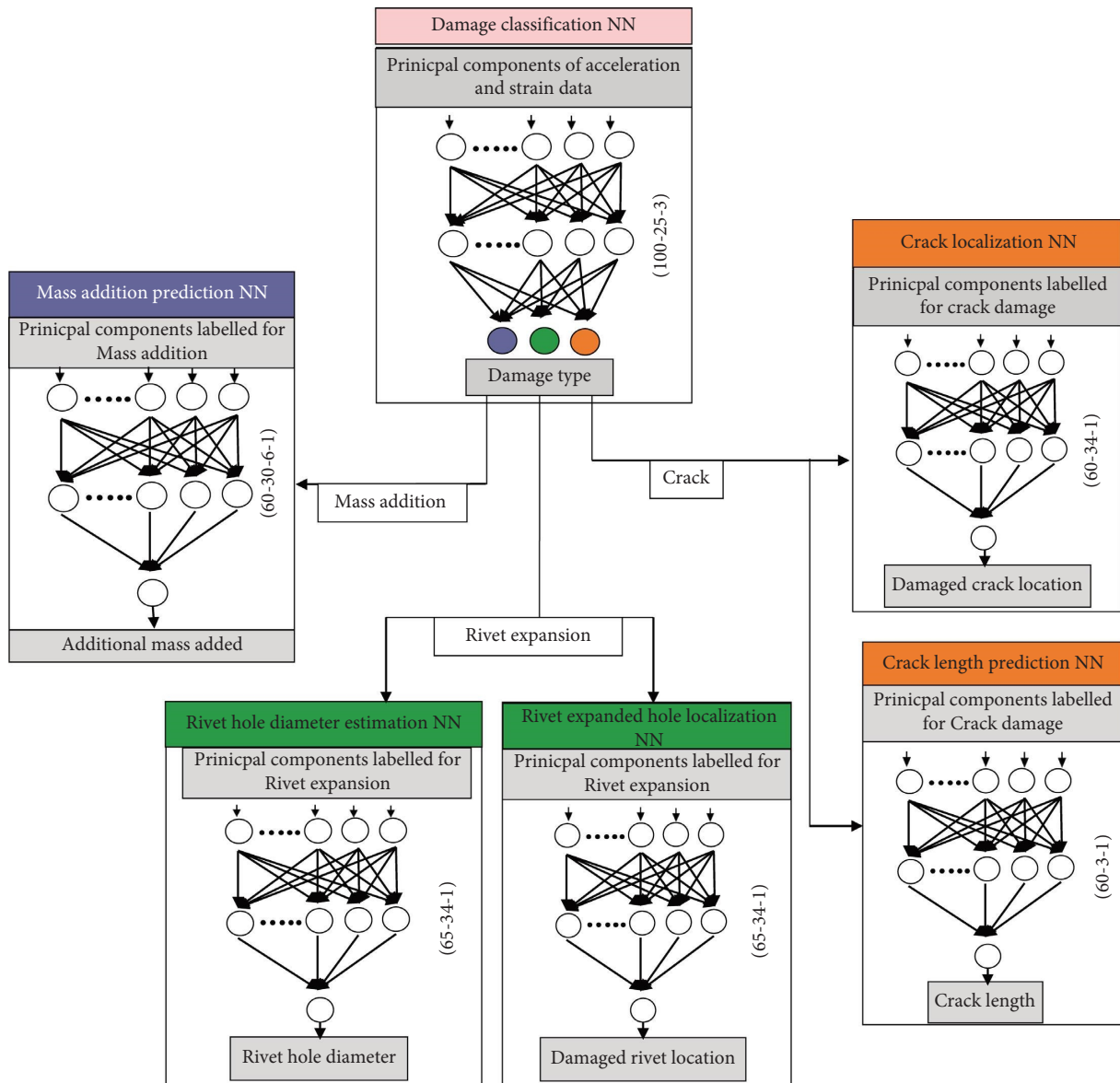Figure 5: Principal components for different types of damage at the same location.

Figure 6: Proposed neural network system.

TABLE 2: Percentage of data for use for training and testing the ANN.

Training (%)	Validation (%)	Testing (%)
60–65	15	20–25

operates in the range of −1 to +1, is selected for hidden layers, and a linear transfer function is chosen for the output layer.

The number of neurons in the hidden layer is chosen following the geometric pyramid rule; neurons in the hidden layer nodes should decrease in number from the input layer to the output layer. We design the detection/classifier network with 100-25-3 distribution of hidden layer nodes and employ the conjugate gradient learning algorithm. For the identifier/estimator networks for crack and rivet expansion localization, we designed a 100-10-34 and 100-50-27 distribution of hidden layer nodes, respectively. Both crack and rivet localization networks are trained with the Bayesian regularization learning algorithm. The identifier/estimator network for mass addition prediction is designed with 100-10-6 distribution of hidden layer nodes and the scaled conjugate gradient learning algorithm. For the identifier/estimator network for crack and rivet expansion severity prediction, we designed a 100-25-2 distribution of hidden layer nodes, with the scaled conjugate gradient learning algorithm. To obtain the best results, each network is trained up to fifteen times with different initial weights and bias values. Furthermore, iteration reduces the absolute error for the training set and increases the error of the validation set, which implies that the network starts overfitting the data and thereby loses its generalization ability.

3.1. Detector/Classifier Network.

The compressed frequency response via PCA is fed to the classifier network (Figure 6). The output of this neural network is in the form of binary data corresponding to the type of damage detected (Table 4). The training performance of the classifier network is shown in Figure 7. The learning algorithm converged after 1500 epochs with the least mean square validation error of 0.09. The number of retrainings is arbitrary and selected on a trial-and-error basis till convergence is satisfactory.

3.2. Identifier/Estimator Network.

The hierarchical network architecture described above was used to predict the location and severity of the damages in the last level of networks based on the input from the classifier network. For the rivet damage severity representation purpose, a binary vector of size $n \times 1$ characterizes the location of rivet expansion damage, where $n = 34$ indicates the number of rivets. Adams learning algorithm is used for training all the above neural networks. The network training performance graphs for the crack localization and expanded rivet localization are shown in Figure 8. The localization algorithm takes about 10000 epochs to find the best solution. The performance parameter monitored is the mean absolute error, which is approximately 2 indicating the capability of the network to estimate

the cracked rivet with ±2 location. The crack length prediction network took 800 epochs to reach the best performance. To test the performance of both networks, the data samples are randomly divided into training and validation. It is observed from Table 5 that both the individual estimator networks for crack localization and rivet expansion localization can map the compressed frequency response data to damage characteristics with reasonable accuracy. However, the network was found to be lacking in its capabilities to learn multiple rivet expansion localization or crack location. The networks may not always pinpoint the exact location, but it can direct to a location of rivet close to the actual damage.

Next, after the damage is localized, the compressed data are fed to the other estimator networks for the respective damage severity prediction. Figure 9 shows the training performance graphs for crack length prediction and expanded rivet diameter prediction networks. The minimum absolute error for crack length prediction network was 0.3 and 0.2 for expanded diameter prediction network. Table 6 shows the predicted parameter value for damage severity from the validation dataset for crack length, and the network was able to estimate the crack length with good accuracy. Table 7 shows the predicted values for the rivet expansion. In this case, the predicted values are quite close to the actual values. The undamaged reference rivet diameter is 3 mm. The rivet diameter of the rivets 16 and 21 was changed from 3 mm to 3.3 mm and 3 mm to 4 mm, respectively. The predicted values are quite close to the actual values.

The estimator network for mass addition on the stiffened panel is trained and tested in terms of uniform layer thickness as the parameter. This classification of damage is analogous to the frost/icing conditions on the aircraft wing or fuselage panels. The allowable thickness of the ice layer in our dataset is limited to 3 mm [70]. The thickness from 0 (healthy) to 3 mm (max. allowable) in steps of 0.5 mm is trained and tested. The thickness of layer variation is adjusted to the panel density. The network output is a binary form, as shown in Table 8.

The performance of the neural network is shown in Figure 10. The Adams optimizer algorithm with the mean absolute error as the performance index shows the optimal performance of the network after 2000 epochs. An examination of five random sample data from the training dataset is given in Table 9 and used for testing. All the sample data are classified appropriately, except sample number 4.

4. Results and Discussion

In this section, we analyze the results of damage detection, classification, and parameter estimation using both simulated data from the stiffened panel on the hierarchical compressed learning network as trained above. As discussed in the previous section, the subnetworks in the layers are trained to sufficient accuracy to predict the damage to the structure when subjected to new data from simulations. Figure 11 shows the output of the classifier network. The classification of the different damage type was an easy task resulting in only a few misclassifications.

TABLE 3: Data samples for each category of damage.

Data category	Data set	Sample vol.	Remarks
Classification network	Training	826 (65%)	34 crack × 2 crack len. × 10 signals 26 rivet exp. × 2 dia. × 10 signals 10 healthy, 6 mass × 10 signals
	Validation	190 (15%)	
	Testing	254 (20%)	
Crack network	Training	544 (80%)	34 crack × 2 crack lengths × 10 signals
	Validation	68 (10%)	
	Testing	68 (10%)	
Rivet expansion network	Training	364 (70%)	26 rivet exp. × 2 dia. × 10 signals
	Validation	78 (15%)	
	Testing	78 (15%)	
Mass addition network	Training	52 (80%)	6 mass additions × 10 signals
	Validation	7 (10%)	
	Testing	7 (10%)	

TABLE 4: Damage state representation in binary form.

Damage type	Binary form of damage		
Crack	1	0	0
Rivet expansion	0	1	0
Mass addition	0	0	1

TABLE 5: Testing performance of the estimator networks.

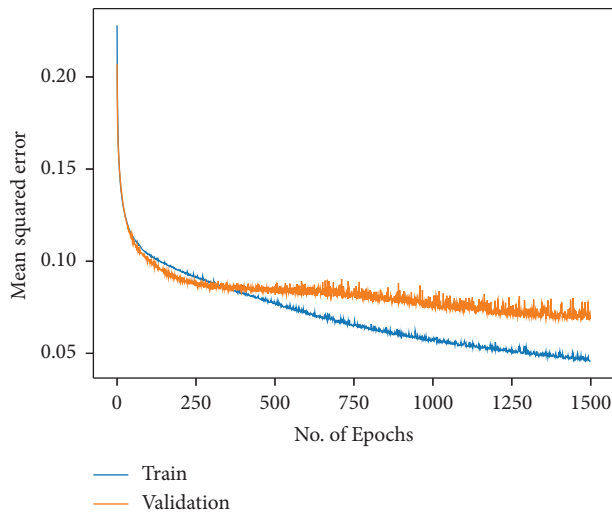| Data sample	Actual rivet no.	Estimator network output
(a) Crack location identification (crack originating from rivet)		
01	0	0
02	6	8/5
03	11	14
04	20	24
05	28	32
(b) Rivet expansion location identification		
01	0	2
02	10	9
03	24	19
04	15	13/17
05	31	27

FIGURE 7: Training error minimization characteristics of classifier network.

The output of the crack location prediction estimator is shown in Figure 12. The output of the Figure 12(a) shows that the crack locations could be detected around the region of the damage. The average error between the predicted and actual crack location was around ±2 rivet positions. In the presence of more than one crack, the location estimation task complicates even further resulting in lower accuracy. After this, the crack size estimator network was trained. The crack was length set to be either 4 mm or 5 mm along with some undamaged cases also. The output of the estimator network for crack size shown in Figure 12(b) indicates that the crack length estimation was satisfactory. The dataset consisted of only one crack at a time. The mean absolute error on the validation dataset was 0.34. Similar procedure

can also be adopted multiple cracks as well with various crack lengths.

The trained networks are also tested for multiple rivet expansion on the stiffened panel, as shown in Figure 13. Figure 13(a) shows the predicted location of expanded rivet compared with actual location of the damage. As compared to the earlier case of crack, detection of expanded rivet was found to be more difficult, and thus, a higher value of the absolute mean error was present. The absolute mean error for the validation dataset was around 2 which indicated that, on average, the network was able to localize the damage within ±2 rivet position of the actual damage. Figure 13(b) indicates the result from the estimator network. The rivets were either expanded to 4 mm or 5 mm compared to the reference diameter of 3 mm. The estimator network was able to predict the diameter with good accuracy resulting in mean absolute error 0.22 for the validation dataset. As with the crack case, in this case also, it was observed that the presence of two expanded rivet proved to be difficult to train with the given number of data points.

A 1.5 mm thick mass layer was added uniformly to generate simulation data for mass addition to the panel and test the network. The classifier network was able to predict the correct class of damage with high probability (Figure 11). Since the uniformity of mass addition on the panel is considered in the present network, only the thickness of the additional mass is used as the parameter for estimation, and
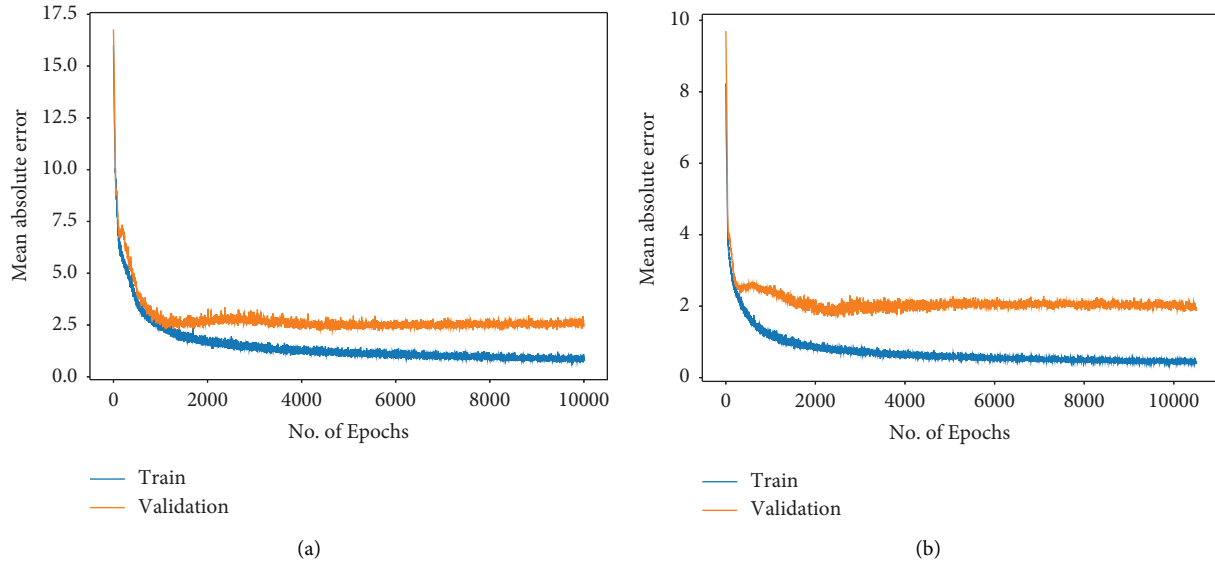
(a)

(b)

FIGURE 8: Training error minimization characteristics of estimator network for (a) crack localization and (b) expanded rivet localization.

TABLE 6: Testing performance of the estimator networks for crack length prediction (crack size measured from rivet hole edge).

Data sample no.	Actual crack size (mm)	Estimated crack size (mm)
01	4	4.5489
02	5	5.3627

TABLE 7: Testing performance of the estimator networks for rivet diameter prediction (reference diameter is taken as 3 mm).

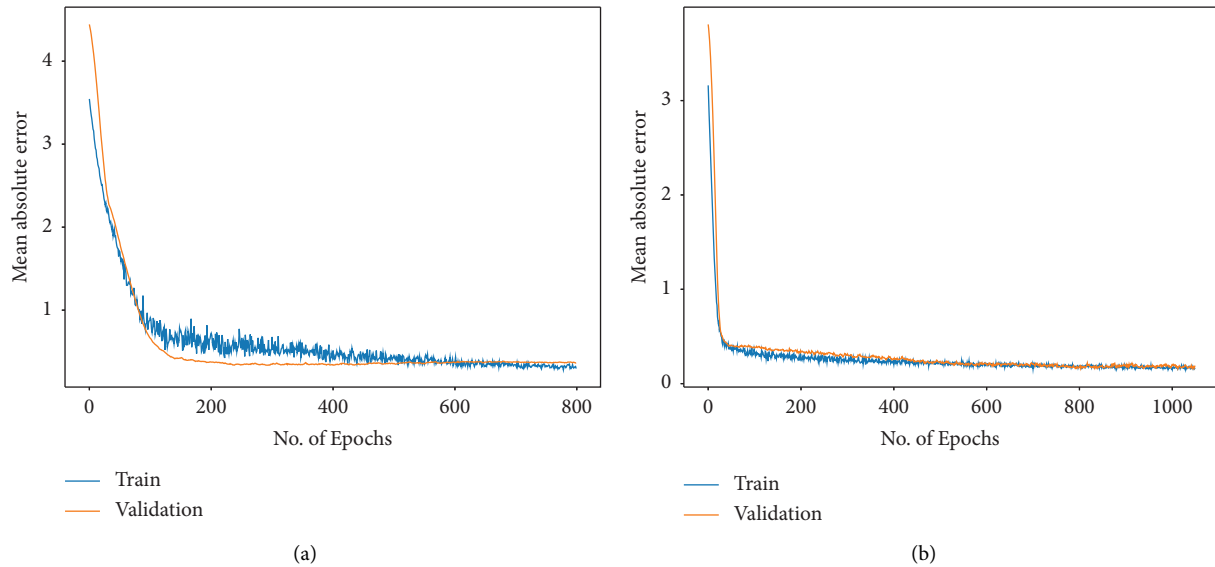| Data sample no.	Actual rivet size (mm)	Estimated rivet dia. (mm)
01	3.3	3.2311
02	4	4.1079

(a)

(b)

FIGURE 9: Training error minimization characteristics of estimator network for (a) crack length prediction and (b) expanded rivet diameter estimation.

TABLE 8: Binary representation of mass addition in terms of additional layer thickness.

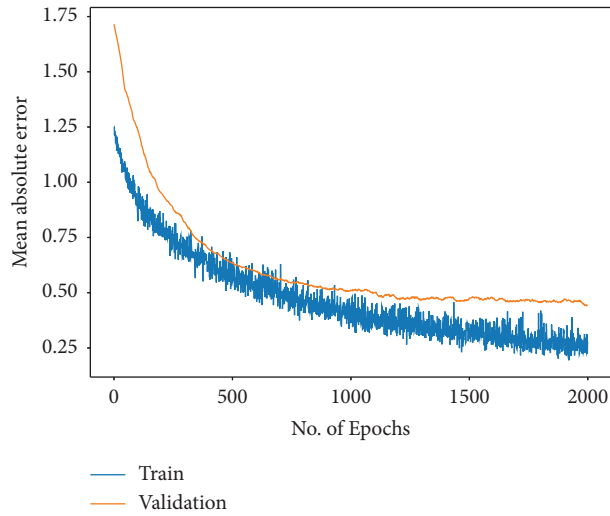| Thickness (mm)	Binary form					
0.0	0	0	0	0	0	0
0.5	1	0	0	0	0	0
1.0	0	1	0	0	0	0
1.5	0	0	1	0	0	0
2.0	0	0	0	1	0	0
2.5	0	0	0	0	1	0
3.0	0	0	0	0	0	1

FIGURE 10: Training error minimization characteristics of mass addition network.

TABLE 9: Random sample from the training dataset used for testing the estimator network for additional mass thickness.

Data sample no.	Actual thickness (mm)	Predicted thickness (mm)
1	0	0.06
2	0.5	0.63
3	1.5	1.74
4	1.5	1.68
5	2	1.8
6	2	0.86
7	3	2.75

location-related parameters were not used. As discussed in the previous section, the damage index for mass thickness parameter was presented in binary with 0.5 mm interval. Therefore, for 1.5 mm mass thickness, the damage index was plotted as shown in Figure 14 for damage states on x-axis. This result demonstrates that for 6 out of the 7 damage cases, the network was able to predict the added mass value with

acceptable level of accuracy. Overall, all these predictions appear quite accurate even after considering the probability of detection and likelihood estimation aspects.

There were total of 6 different simple learners used in the current work for various tasks. The accuracies of these networks are shown at a glance in Table 10. Except for the first classification network, the accuracy has been shown in

Confusion Matrix

	Class 1	Class 2	Class 3	
Class 1	331 36.0%	21 2.3%	3 0.3%	93.2% 6.8%
Class 2	9 1.0%	495 53.8%	4 0.4%	97.4% 2.6%
Class 3	5 0.5%	0 0.0%	52 5.7%	91.2% 8.8%
	95.9% 4.1%	95.9% 4.1%	88.1% 11.9%	95.4% 4.6%

FIGURE 11: Confusion matrix of the classifier network. Class 1 damage is crack, class 2 damage is rivet expansion, and class 3 damage is mass addition.
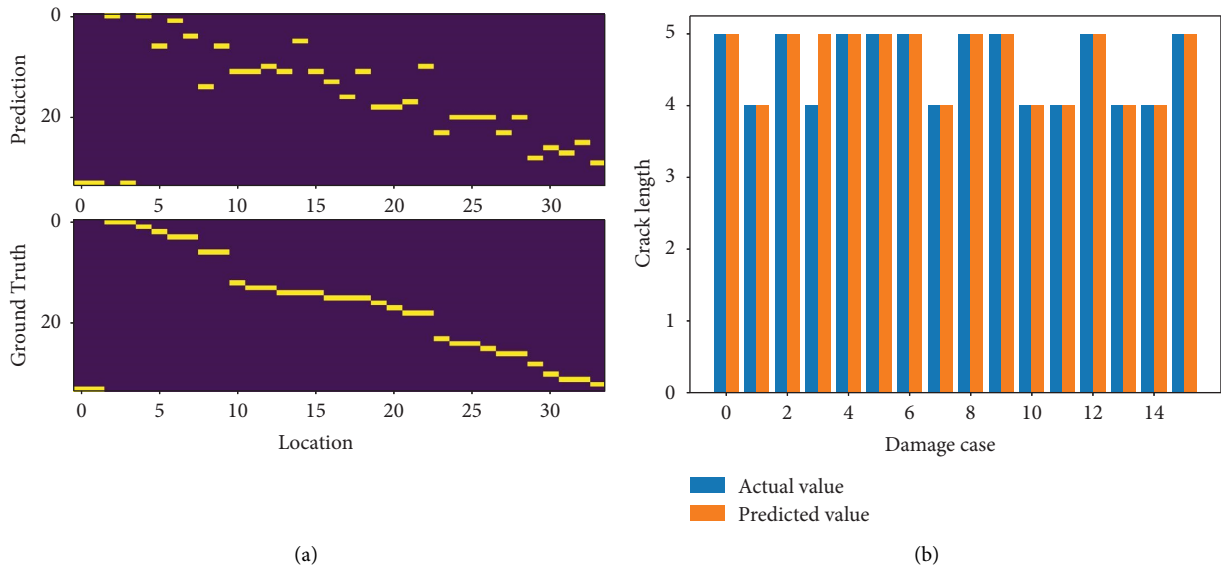

(a)

Actual value
Predicted value

(b)

FIGURE 12: (a) The output of crack location prediction network is shown for a validation dataset. The lower half of the figure shows the location of actual crack damage, and the upper half indicates the predicted location. (b) Bar plot of the crack size prediction.
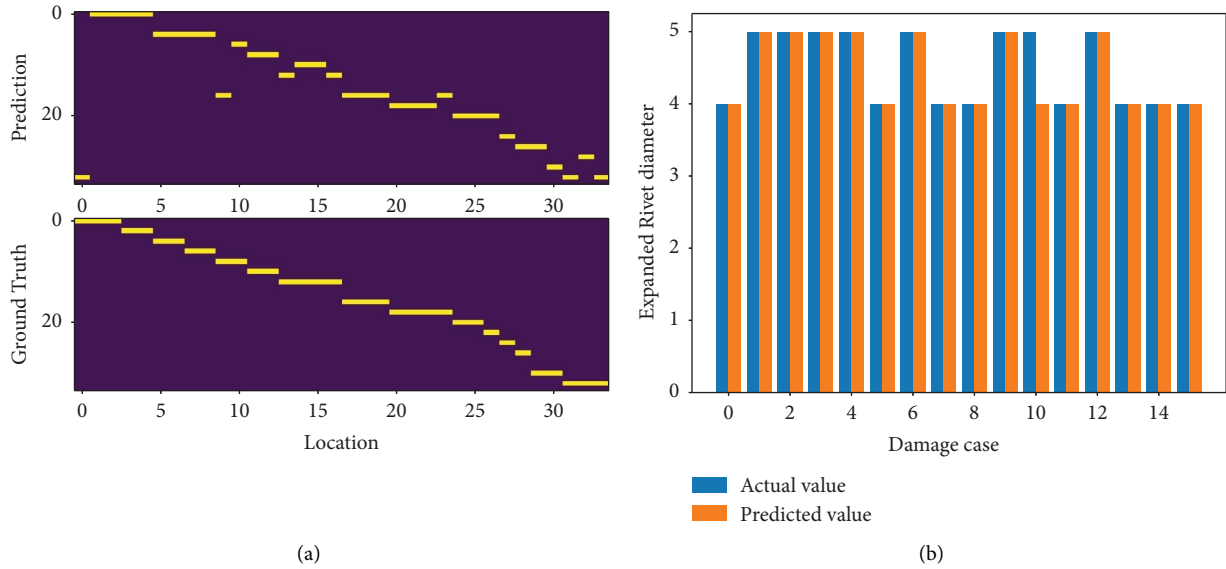
(a)

(b)

FIGURE 13: (a) The output of rivet expansion location prediction network is shown for a validation dataset. The lower half of the figure shows the location of actual rivet expansion damage, and the upper half indicates the predicted location. (b) Bar plot of the expanded rivet diameter prediction for 15 different cases compared with the actual value.
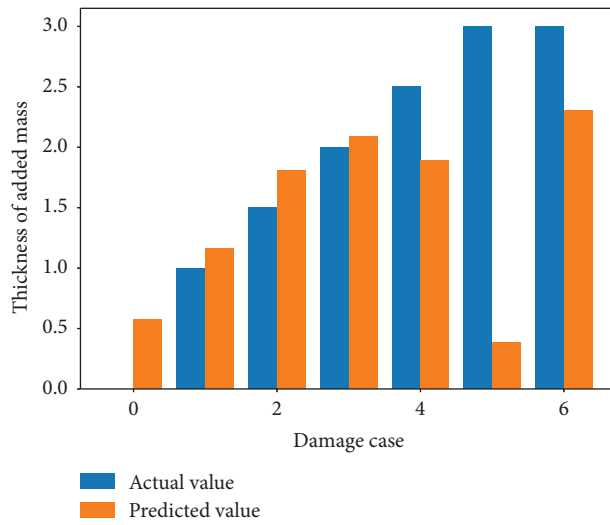

FIGURE 14: Mass addition prediction on the panel.

TABLE 10: The accuracy metric of all the neural networks used.

Neural net category	Accuracy metric
Classification network	95.4%
Crack location	1.5 (mean absolute error)
Crack length prediction	0.34 (mean absolute error)
Rivet location	2 (mean absolute error)
Rivet diameter expansion	0.22 (mean absolute error)
Mass addition	0.32 (mean absolute error)

terms of mean absolute error. The reported values are calculated using a validation dataset which was not used for training the network.

5. Conclusion

This study demonstrates the feasibility of a hierarchical sensing-based classifier-estimator network designed for simulation data-driven learning regarding structural damages, which are usually hard to detect due to nonuniqueness and nonstationarities in high-dimensional vibration data. We first analyze the data compression characteristics and principal components derived from random vibration frequency response data. The data are fed in a classifier network and parallelly provided to further layers of estimator networks for damage parameter estimation. A simultaneous classification and parameter estimation approach leads to interesting ways of developing probabilistic detection and likelihood estimation of the damage parameters. The network architecture and data representation are designed so that the demonstrative examples give useful insight into the synthetic data-driven training at various levels of the hierarchical network. We also show a detailed analysis of the network test outcome and capabilities to localize specific damages and estimate the damage parameters by the network.

Different learning rules are used for different neural networks within the developed architecture to obtain enhanced performance. The Bayesian regularization learning rule produces a good network performance even with a small training dataset but at the expense of computational power during training. It was observed that retraining the networks several times would improve their performance characteristics. It was observed that damage parameter estimation (damage location and severity) is a challenging task that needs further work to improve accuracy. The simulation is also required for the damage at each location and damage severity which does not scale well in cases when the damage can occur at any location. In the current study, a single damage case is considered at a time. For future work, multiple simultaneous damage scenarios will be needed, potentially with a probabilistic approach, and a comparative study of the performance of different ML algorithms should be considered. The current study also indicates the limitations in generating simulation data based on supervised learning algorithms. The present study opens up the need for further investigation on various synthetic data generation-related aspects such as introducing uncertainties and noise in the simulation dataset, scaling and transforming in PCA for learning across completely different structural geometries, mesh granularities, different damage cases etc., and transitioning to recursive training and testing processes as the network expands in dimension and the learning space.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] C. Boller, D. R. Mahapatra, R. Sridaran Venkat et al., "Integration of Non-Destructive Evaluation-based Ultrasonic Simulation: a means for simulation in structural health monitoring," *Structural Health Monitoring: International Journal*, vol. 16, no. 5, pp. 611–629, 2017.

[2] L. Qiu, F. Fang, S. Yuan, C. Boller, and Y. Ren, "An enhanced dynamic Gaussian mixture model–based damage monitoring method of aircraft structures under environmental and operational conditions," *Structural Health Monitoring*, vol. 18, no. 2, pp. 524–545, 2019.

[3] S. W. S. Doebling, C. R. C. Farrar, M. B. M. Prime, and D. W. D. Shevitz, "Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review," *Los Alamos National Lab*, 1996.

[4] R. Nataraja, "Structural integrity monitoring in real seas," *The Day*, vol. 1983, pp. 221–228, 1983.

[5] C. Zang and M. Imregun, "Structural damage detection using artificial neural networks and measured frf data reduced via principal component projection," *Journal of Sound and Vibration*, vol. 242, no. 5, pp. 813–827, 2001.

[6] Y. Liu, L. Guo, H. Gao, Z. You, Y. Ye, and B. Zhang, "Machine vision based condition monitoring and fault diagnosis of machine tools using information from machined surface texture: a review," *Mechanical Systems and Signal Processing*, vol. 164, Article ID 108068, 2022.

[7] C.-Z. Dong and F. N. Catbas, "A review of computer vision–based structural health monitoring at local and global levels," *Structural Health Monitoring*, vol. 20, no. 2, pp. 692–743, 2021.

[8] W. Choi and Y. J. Cha, "SDDNet: real-time crack segmentation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 8016–8025, 2020.

[9] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 731–747, 2018.

[10] D. H. Kang and Y. J. Cha, "Efficient attention-based deep encoder and decoder for automatic crack segmentation," *Structural Health Monitoring*, vol. 21, no. 5, pp. 2190–2205, 2022.

[11] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, "Road damage detection and classification using deep neural networks with smartphone images," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 12, pp. 1127–1141, 2018.

[12] G. Ciaparrone, A. Serra, V. Covito, P. Finelli, C. A. Scarpato, and R. Tagliaferri, "A deep learning approach for road damage

classification," *Advanced Multimedia and Ubiquitous Engineering: MUE/FutureTech 2018 12*, pp. 655–661, 2019.

[13] A. Zhang, K. C. P. Wang, B. Li et al., "Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 10, pp. 805–819, 2017.

[14] A. Zhang, K. C. P. Wang, Y. Fei et al., "Deep learning--based fully automated pavement crack detection on 3D asphalt surfaces with an improved CrackNet," *Journal of Computing in Civil Engineering*, vol. 32, no. 5, Article ID 4018041, 2018.

[15] Y. Fei, K. C. P. Wang, A. Zhang et al., "Pixel-level cracking detection on 3D asphalt pavement images through deep-learning-basedCrackNet-V," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 273–284, 2020.

[16] P. M. Ferreira, M. A. Machado, M. S. Carvalho, and C. Vidal, "Embedded sensors for structural health monitoring: methodologies and applications review," *Sensors*, vol. 22, no. 21, p. 8320, 2022.

[17] X. Fang, H. Luo, and J. Tang, "Structural damage detection using neural network with learning rate improvement," *Computers & Structures*, vol. 83, no. 25-26, pp. 2150–2161, 2005.

[18] Z. Wang, R. M. Lin, and M. K. Lim, "Structural damage detection using measured FRF data," *Computer Methods in Applied Mechanics and Engineering*, vol. 147, no. 1-2, pp. 187–197, 1997.

[19] S. Das and K. Roy, "A state-of-the-art review on FRF-based structural damage detection: development in last two decades and way forward," *International Journal of Structural Stability and Dynamics*, vol. 22, no. 2, Article ID 2230001, 2022.

[20] R. K. Munian, D. R. Mahapatra, and S. Gopalakrishnan, "Lamb wave interaction with composite delamination," *Composite Structures*, vol. 206, pp. 484–498, 2018.

[21] C. R. Farrar and K. M. Cone, *Vibration Testing of the I-40 Bridge before and after the Introduction of Damage*, Los Alamos National Lab, Los Alamos, NM USA, 1994.

[22] F. Ismail, A. Ibrahim, and H. R. Martin, "Identification of fatigue cracks from vibration testing," *Journal of Sound and Vibration*, vol. 140, no. 2, pp. 305–317, 1990.

[23] C. H. J. Fox, "The location of defects in structures-A comparison of the use of natural frequency and mode shape data," *10th International Modal Analysis Conference*, vol. 1, pp. 522–528, 1992.

[24] G. Kolappan Geetha and D. Roy Mahapatra, "Modeling and simulation of vibro-thermography including nonlinear contact dynamics of ultrasonic actuator," *Ultrasonics*, vol. 93, pp. 81–92, 2019.

[25] G. Kolappan Geetha, R. K. Munian, D. R. Mahapatra, and D. A. Raulerson, "Ultrasonic horn contact-induced transient anharmonic resonance effect on vibro-thermography," *Journal of Sound and Vibration*, vol. 525, Article ID 116786, 2022.

[26] J. G. Chen, N. Wadhwa, Y.-J. Cha, F. Durand, W. T. Freeman, and O. Buyukozturk, "Modal identification of simple structures with high-speed video using motion magnification," *Journal of Sound and Vibration*, vol. 345, pp. 58–71, 2015.

[27] J. Zhao and J. Tang, "Amplifying damage signature in periodic structures using enhanced piezoelectric networking with negative resistance elements," *Journal of Intelligent Material Systems and Structures*, vol. 24, no. 13, pp. 1613–1625, 2013.

[28] R. Gangadharan, C. R. L. Murthy, S. Gopalakrishnan, M. R. Bhat, and D. R. Mahapatra, "Characterization of cracks and delaminations using PWAS AD Lamb wave based time-

frequency methods," *International Journal on Smart Sensing and Intelligent Systems*, vol. 3, no. 4, pp. 703–735, 2017.

[29] A. Nag, D. Roy Mahapatra, S. Gopalakrishnan, and T. S. Sankar, "A spectral finite element with embedded delamination for modeling of wave scattering in composite beams," *Composites Science and Technology*, vol. 63, no. 15, pp. 2187–2200, 2003.

[30] D. Roy Mahapatra, S. Suresh, S. N. Omkar, and S. Gopalakrishnan, "Estimation of degraded composite laminate properties using acoustic wave propagation model and a reduction-prediction network," *Engineering Computations*, vol. 22, no. 7, pp. 849–876, 2005.

[31] K. Worden and A. J. Lane, "Damage identification using support vector machines," *Smart Materials and Structures*, vol. 10, no. 3, pp. 540–547, 2001.

[32] P. Konar and P. Chattopadhyay, "Bearing fault detection of induction motor using wavelet and Support Vector Machines (SVMs)," *Applied Soft Computing*, vol. 11, no. 6, pp. 4203–4211, 2011.

[33] A. Tabrizi, L. Garibaldi, A. Fasana, and S. Marchesiello, "Early damage detection of roller bearings using wavelet packet decomposition, ensemble empirical mode decomposition and support vector machine," *Meccanica*, vol. 50, no. 3, pp. 865–874, 2015.

[34] Q. Xu, "Impact detection and location for a plate structure using least squares support vector machines," *Structural Health Monitoring*, vol. 13, no. 1, pp. 5–18, 2014.

[35] J. W. Lee, G. R. Kirikera, I. Kang, M. J. Schulz, and V. N. Shanov, "Structural health monitoring using continuous sensors and neural network analysis," *Smart Materials and Structures*, vol. 15, no. 5, pp. 1266–1274, 2006.

[36] X. Wu, J. Ghaboussi, and J. H. Garrett, "Use of neural networks in detection of structural damage," *Computers & Structures*, vol. 42, no. 4, pp. 649–659, 1992.

[37] W. B. Spillman, D. R. Huston, P. L. Fuhr, and J. R. Lord, "Neural network damage detection in a bridge element," *North America Conference Smart Structure Mater*, vol. 1918, pp. 288–299, 1993.

[38] J. Rhim and S. W. Lee, "A neural network approach for damage detection and identification of structures," *Computational Mechanics*, vol. 16, no. 6, pp. 437–443, 1995.

[39] X. Zeng, Y. Liao, and W. Li, "Gearbox fault classification using S-transform and convolutional neural network," in *Proceedings of the International Conference on Sensing Technology*, pp. 0–4, Nanjing, China, November 2016.

[40] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-D convolutional neural networks," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7067–7075, 2016.

[41] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "1-D CNNs for structural damage detection: verification on a structural health monitoring benchmark data," *Neurocomputing*, vol. 275, pp. 1308–1317, 2018.

[42] G. Vu, J. J. Timothy, D. S. Singh, L. A. Saydak, E. H. Saenger, and G. Meschke, "Numerical simulation-based damage identification in concrete," *Modelling*, vol. 2, no. 3, pp. 355–369, 2021.

[43] M. L. Mavrovouniotis and S. Chang, "Hierarchical neural networks," *Computers & Chemical Engineering*, vol. 16, no. 4, pp. 347–369, 1992.

[44] K. Salahshoor, M. Kordestani, and M. S. Khoshro, "Fault detection and diagnosis of an industrial steam turbine using fusion of SVM (support vector machine) and ANFIS (adaptive

neuro-fuzzy inference system) classifiers," *Inside Energy*, vol. 35, no. 12, pp. 5472–5482, 2010.

[45] C. Shen, L. Wang, and Q. Li, "Optimization of injection molding process parameters using combination of artificial neural network and genetic algorithm method," *Journal of Materials Processing Technology*, vol. 183, no. 2-3, pp. 412–418, 2007.

[46] H. Song, L. Zhong, and B. Han, "Structural damage detection by integrating independent component analysis and support vector machine," *International Journal of Systems Science*, vol. 37, no. 13, pp. 961–967, 2006.

[47] D. R. Mahapatra, S. Suresh, S. N. Omkar, and S. Gopalakrishnan, "Estimation of degraded composite laminate properties using acoustic wave propagation model and a reduction-prediction network," *Engineerings with Computing*, 2005.

[48] Y. Cheng, H. Yuan, H. Liu, and C. Lu, "Fault diagnosis for rolling bearing based on SIFT-KPCA and SVM," *Engineering Computations*, vol. 34, no. 1, pp. 53–65, 2017.

[49] J. Shu, Z. Zhang, I. Gonzalez, and R. Karoumi, "The application of a damage detection method using Artificial Neural Network and train-induced vibrations on a simplified railway bridge model," *Engineering Structures*, vol. 52, pp. 408–421, 2013.

[50] S.-F. Jiang, C. M. Zhang, and S. Zhang, "Two-stage structural damage detection using fuzzy neural networks and data fusion techniques," *Expert Systems with Applications*, vol. 38, no. 1, pp. 511–519, 2011.

[51] P. Quirke, C. Bowe, E. J. Obrien, D. Cantero, P. Antolin, and J. M. Goicolea, "Railway bridge damage detection using vehicle-based inertial measurements and apparent profile," *Engineering Structures*, vol. 153, pp. 421–442, 2017.

[52] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *Journal of Sound and Vibration*, vol. 388, pp. 154–170, 2017.

[53] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[54] F. Luleci, F. N. Catbas, and O. Avci, "Generative adversarial networks for labelled vibration data generation," *Special Topics in Structural Dynamics \& Experimental Techniques*, vol. 5, pp. 41–50, 2023.

[55] Ş Y. Selçuk, P. Ünal, Ö. Albayrak, and M. Jomâa, "A workflow for synthetic data generation and predictive maintenance for vibration data," *Information*, vol. 12, no. 10, p. 386, 2021.

[56] R. Ali and Y.-J. Cha, "Attention-based generative adversarial network with internal damage segmentation using thermography," *Automation in Construction*, vol. 141, Article ID 104412, 2022.

[57] H. Maeda, T. Kashiyama, Y. Sekimoto, T. Seto, and H. Omata, "Generative adversarial network for road damage detection," *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 1, pp. 47–60, 2021.

[58] K. Dunphy, A. Sadhu, and J. Wang, "Multiclass damage detection in concrete structures using a transfer learning-based generative adversarial networks," *Structural Control and Health Monitoring*, vol. 29, no. 11, Article ID e3079, 2022.

[59] X. Lei, L. Sun, and Y. Xia, "Lost data reconstruction for structural health monitoring using deep convolutional generative adversarial networks," *Structural Health Monitoring*, vol. 20, no. 4, pp. 2069–2087, 2021.

[60] J. Mao, H. Wang, and B. F. Spencer, "Toward data anomaly detection for automated structural health monitoring: exploiting generative adversarial nets and autoencoders," *Structural Health Monitoring*, vol. 20, no. 4, pp. 1609–1626, 2021.

[61] I. Trendafilova, W. Heylen, and P. Sas, "Damage localization in structures . A pattern recognition perspective," *Pattern Recognit*, pp. 99–106, 1998.

[62] Z. Wang and Y. J. Cha, "Unsupervised machine and deep learning methods for structural damage detection: a comparative study," *Engineering Reports*, 2022.

[63] H. Khodabandehlou, G. Pekcan, and M. S. Fadali, "Vibration-based structural condition assessment using convolution neural networks," *Structural Control and Health Monitoring*, Article ID e2308, 2018.

[64] D. S. Singha, G. B. L. Chowdarya, and D. R. Mahapatraa, "Structural Damage Identification Using Artificial Neural Network and Synthetic Data," 2017, http://arxiv.org/abs/1703.09651.

[65] Y. Yu, C. Wang, X. Gu, and J. Li, "A novel deep learning-based method for damage identification of smart building structures," *Structural Health Monitoring*, vol. 18, no. 1, pp. 143–163, 2018.

[66] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[67] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–441, 1933.

[68] C. M. Bishop, "Neural networks and their applications," *Review of Scientific Instruments*, vol. 65, no. 6, pp. 1803–1832, 1994.

[69] A. Papert Seymour, *Perceptrons*, 2022, https://www.simplilearn.com/tutorials/deep-learning-tutorial/perceptron#:%7E:text=A%20neural%20network%20link%20that,logic%20gates%20with%20binary%20outputs.

[70] Manual and Aircraft Maintenance, *Of Airbus A320 Flight Data Recording Parameter Library*, FDRPL Airbus, Paris, France, 2005.