Available online at www.eccomasproceedia.org
Eccomas Proceedia SMART (2023) 1863-1875

ECCOMAS
Proceedia

X ECCOMAS Thematic Conference on Smart Structures and Materials
SMART 2023
D.A. Saravanos, A. Benjeddou, N. Chrysochoidis and T. Theodosiou (Eds)

A DIGITAL TWIN OF A FIBER BRAGG GRATING NETWORK IN THE LEADING EDGE OF A ROTATING COMPOSITE FAN BLADE SUBJECTED TO BIRD STRIKE

GIANNIS FLOROS[*], DIMITRIS SOTIROPOULOS[†], KONSTANTINOS TSERPES[*] AND DIMITRIOS TSOUROUNIS[ℵ]

[*] Laboratory of Technology & Strength of Materials, Department of Mechanical Engineering & Aeronautics, University of Patras, Rio, 26504, Greece

[†] Department of Electrical and Computer Engineering, University of Peloponnese, Patras, 26334, Greece

[ℵ] Electronics Laboratory, Department of Physics, University of Patras, Rio, 26504, Greece
email: dtsourounis@upatras.gr
ORCID I.D. : 0000-0003-2249-2790
Scopus Author I.D. : 57195940702

Abstract. In the last two decades, an intensive effort is in being made to manufacture smart composite structures with embedded fiber optic sensors that can be exploited to detect, localize, and classify damage. In this context, the use of digital twins for the virtual design and optimization of a sensors' network may give insight to complex phenomena and significantly reduce the development time and cost of the smart structures. Composite engine fan blades are prone to bird strike damage and thus, early detection of bird strike damage is crucial for the efficient maintenance of the engines and the prevention of critical incidents. In this work, a Fiber Bragg Grating (FBG) network in the leading edge of a rotating composite fan blade subjected to bird strike is developed. The fan blade is made of 3D woven composite material and its leading edge is protected by an adhesively bonded layer made of steel while the FBG network is placed into the bondline. The Digital Twin (DT) of the above description integrates a Finite Element (FE) model and Artificial Neural Network (ANN) models. In the FE model, progressive damage in the composite material and debonding of the steel layer have been simulated. Several bird strike simulation scenarios were performed at different strike locations. At the time of the bird strike, the blade is rotating with a rotation speed of 3000 rpm. ANNs were trained using the numerical strain data in order to predict the existence of debonding of the steel layer and the area of debonding. ANNs exploit Convolutional Neural Networks (CNNs) to capture spatial dependencies in input signals. Experimental results show ANNs provide low root mean square error (RMSE) in both regression tasks, effectively predicting the bird strike location and the damage in terms of the relevant affecting area.

Key words: Finite element analysis, SHM, Fiber optic sensors, Smart composite structures,

doi: 10.7712/150123.9956.445571

Fan blade, Bird strike

1 INTRODUCTION

Nowadays, composites materials are widespread in various applications and fields due to their advantages in terms of strength and mechanical behavior, compared to traditional metallic materials. Over the last 20 years, extensive research is being performed to implement smart composite structures which integrate advanced features for the detection, localization and classification of damages during their operation. This could be performed by using embedded fiber optic sensors like FBGs which are used to monitor the operating conditions. Data provided by FBGs can be used to detect possible damages. To this end, DT play an important role in various tasks. On one hand, DT could be a useful tool for designing and optimize the sensors' network, in terms of topology and density. On the other hand, a DT which is implemented by means of a FE model could be used to generate numerous data to subsequently train and evaluate an ANN network. The ANN network, during operation or maintenance conditions could be useful to classify, localize and quantify damages in critical structures.

Yagawa and Okuda [1] reviewed the neural network applications to the computational mechanics fields and afterwards detailed their contribution in various task such as crack growth analysis of welded specimens, structural design etc. Hernandez-Gomez et al. [2] presents an ANN for locating defects using dynamic strain analysis. An ANN approach for multiple defects localization have been proposed by Farley et al. [3] A computational inverse technique for the localization and classification of defects using a hybrid data base have been performed by Luna-Avilés et al [4]. Lu et al. [5] investigated a damage identification algortithm using FBG sensor's data.

In aircraft structures, phenomena which cause important damage are various foreign object damage (FOD) impacts such as hail, debris, and bird strike, which is the most frequent. This could cause several damages to all the outer components of the aircraft's nose, leading edge of the wings, as well as the fan blades of the engine. Due to the severity of the bird strike impact, aviation authorities have defined strict certification regulations concerning the capacity of the critical components to withstand bird strike events.

Lamberti et al. [6] investigated the feasibility to utilize the FBG network data to localize and quantify impact events. Sarego et al. [7] among others, used a large number of spectral components of the signal have been used to reconstruct the force due to small mass impacts and measure the uncertainty of the prediction of the impact position. Gomes et al. [8] proposed a damage identification method on a helicopter's main rotor blade by combining a FE model with a bat optimization algorithm to An ANN for damage detection, trained using in situ distributed strained have been implemented by Califano et al. [9]. Xu et al. [10] developed a DT modeling method for hierarchical stiffened panel. Based on a pre-trained deep neural network, the model has been fine-tuned and finally, the DT was capable to visualize the full-field strength of the structure. Finally, Moradi et al. constructed an intelligent health indicator of composite structures using data provided by deep neural network and SHM data [11].

In this paper, the DT integrates a FE model of a rotating fan blade, subjected to bird impact and an ANN, trained and tested using data provided from the FE model, after a parametrical analysis concerning the impact point. The modeled geometry of the rotating blade consists of the main CFRP part, and an adhesively bonded steel leading edge ANN output is the impact location and the debonded area after the bird strike phenomenon.

2 PRINCIPLES OF THE CONCEPT

An embedded FBG network is installed in the bondline between the main composite part of the panel and the adhesively bonded steel leading edge. The FBG network sends current strain values in a processing unit. In this unit, where an ANN have already been implemented (trained and tested), the strain values from the FBGs are used as input and the system respond about possible damage creation due to an impact, the impact location as well as the extent of the damage in the bondline.

This concept could facilitate scheduled maintenance procedure through damage detection. Furthermore, the system could be used after an impact phenomenon to quantify the damage and, in combination with the DT, localize the damage. Afterwards, the maintenance procedure could be focused the on the damaged area, which will be predicted. The above procedure is schematically shown in Figure 1(a).

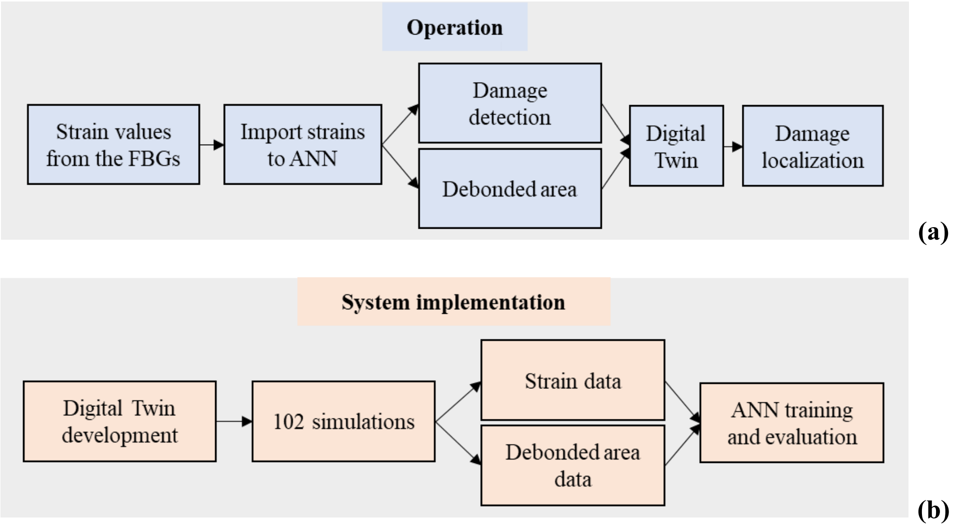

Figure 1: (a) Proposed operation of the developed system. **(b)** Schematic represantation of the workflow for the implementation of the system.

For the implementation of the concept, a DT by means of a FE model is developed to simulate bird strike scenarios in various positions on the steel leading edge bonded on a rotating CFRP fan blade. Numerous data have been created by performing analyses for 102 bird strikes on different impact locations of the leading edge. The numerical results have been used as data to train and evaluate an ANN. As already mentioned, after the successful training procedure, strain values are imported as input to the ANN whose output is the impact location and extent of damage (value of debonded area). A schematic representation of the system's implementation is depicted in Figure 1(b).

3 DIGITAL TWIN

3.1 Finite Element Model

The DT implemented using a FE model, developed in the LS-DYNA explicit code. The simulation concerns an artificial bird impact on a rotating part of an FOD panel. The model consists of:

- The main body of the FOD panel which represents a part of a turbine blade.
- A steel cover at the leading edge.
- An adhesive layer which bonds the main body of the FOD panel with the leading edge.
- An artificial bird impactor, modeled as a cylindrical object with hemispherical ends.
- An FBG network in the bondline between the main part of the FOD panel and the leading edge.

The main body of the FOD panel is a 3D woven CFRP and has been modeled with MAT-162_COMPOSITE_MSC_DMG material model which implements a progressive damage model. For the steel leading edge, MAT_003-PLASTIC_KINEMATIC has been used, which is a material model suitable for simulating materials with kinematic hardening plasticity behavior. Furthermore, MAT_138-COHESIVE_MIXED_MODE with bilinear mixed-mode traction-separation has been used to simulate the mechanical behavior of the adhesive. Additionally, the FBG network has been modeled with cylindrical beam elements and MAT_001-ELASTIC. Concerning the artificial bird which modeled with SPH particles, MAT_010-ELASTIC_PLASTIC_HYDRO and the Gruneisen equation of state has been adopted.

The artificial bird has a mass 3.6 kg and an initial velocity of 10.8 m/sec resulting in an impact energy of 210J on the 800mm-wide rotating blade with rotation speed of 300 rpm. The different parts of geometry are presented in Figure 2. In Figure 3 the entire geometry of the panel, as well as the geometry of the FBG network which is integrated in the bondline are depicted. The axial strain values derived from the beam elements and the debonded area due to the bird strike will be used for the following ANN training and evaluation. Details about the developed FE model (boundary conditions, mechanical properties etc.) can be found in previous work performed by the authors [12].

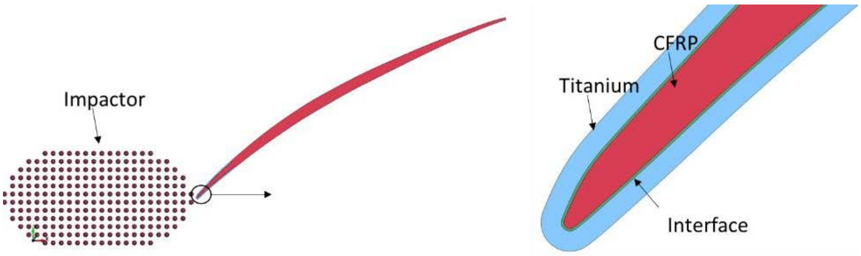

Figure 2: Parts of the modeled geometry.

3.2 Simulation scenarios

Numerous simulation scenarios had to be performed in order to obtain strain and debonded area values data which will be used to train and evaluate the ANN. More specifically, 102 simulations have been executed for 102 different impact locations along the line across the width of the FOD panel (Figure 3(a)), ranging from 0 to 800 mm.

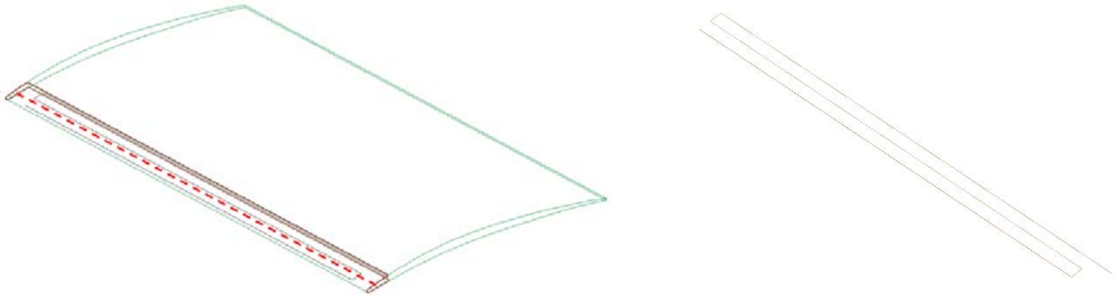

Figure 3: (a) Impact locations across the width of the FOD panel, (b) Schematic representation of the FBG network in the bondline.

4 NUMERICAL RESULTS

After the completion of every simulation scenario, the numerical results that were extracted were the axial strain of the beam elements that form the FBG network and the value of the debonded area (in terms of mm^2) of the adhesive that bonds the CFRP part of the panel with the steel leading edge. For the sake of brevity, the numerical results for the scenario where the impact location is 352.95 mm are presented.

In Figure 4 and Figure 5, the contour plot and the distribution of axial strains that derived from the 244 elements that form the FBG network are presented. For comparative reasons, the predicted axial strains from the model with impact location 588.25 mm has been added to the figure. As can be seen, from these different impact locations, the FBG network is quite sensitive and predicts important difference in axial strains along the network.

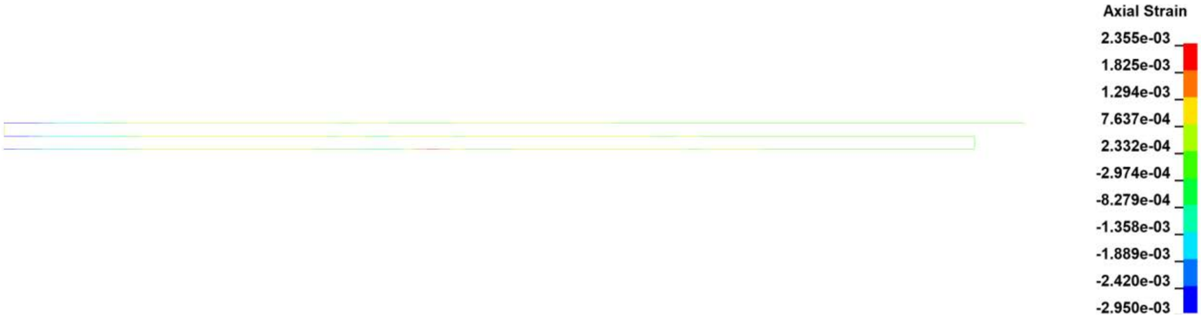

Figure 4 : Contour plot of axial strains of beam elements along the FBG network.

Additionally, concerning the damage of the adhesive, for the analysis with 352.95 mm impact location, the debonded elements (in blue color) from both sides of the adhesive, are shown in Figure 5. The debonding occurs close to the impact area as well as on the area that are close to the rotation axis. From all performed analyses, the debonded area (in mm^2) has been extracted.

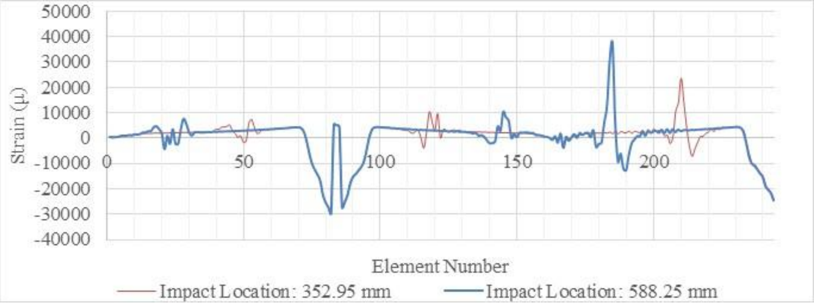

Figure 5: Axial strains distribution of beam elements along the FBG network.

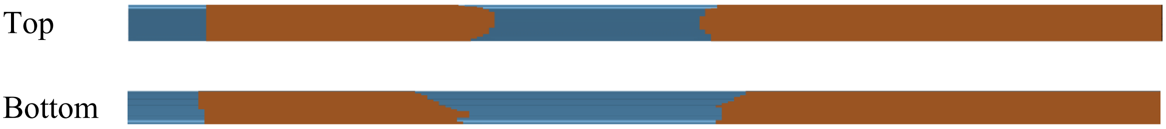

Figure 6 : Debonded elements (in blue color) for both sides of the adhesive.

5 ARTIFICIAL NEURAL NETWORKS

5.1 Dataset

A total of 102 simulations were carried out to investigate the effects of bird strikes on the FOD panel. These simulations covered 102 distinct impact locations along the width of the panel. The strain vector serves as the input for the machine learning models, which predict either the impact location (in millimeters) or the damaged area (in square millimeters) as output. All the input feature vectors are z-score normalized prior to being fed into the machine learning models.

5.1.1 Experimental protocol

In machine learning projects, it is common practice to split the available data into three subsets: the training set, the validation set, and the test set. The training set is used to train the model by feeding it inputs and their corresponding outputs. The model learns from this data and adjusts its parameters to improve its performance. The validation set is then used to evaluate the model's performance and tune its hyperparameters. Hyperparameters are parameters that cannot be learned from the data and need to be set before training, such as the learning rate or the number of layers in the model. The test set is reserved to evaluate the model's performance on unseen data. It is important to have a separate test set to detect overfitting, which is when a model performs well on the training data but poorly on new data. By evaluating the model's performance on the test set, we can get an estimate of its generalization ability. In this work, the test set is generated from the one-third of data while the other two-thirds of data are used both for the training and the validation set. More specifically, from the 102 simulation scenarios, 34 scenarios are utilized for the test set, 60 scenarios are used for the training set, and the validation set is formed with 8 scenarios.

To split the data into the test set, training set, and validation set, we used a systematic random sampling technique since the bird strikes are simulated on different impact locations of the leading edge. The simulations were executed across the width of the FOD panel, and the corresponding data were ordered in this direction. Hence, every third sample is selected to create the test set. From the remaining samples, every eighth sample is selected to create the validation set and finally, the rest of the samples were used for the training set. The starting point of the sampling technique varied, so we could develop three different protocols to ensure that the test set contained one-third of the data. This approach is more appropriate than using a random sample to split the available data, as it allows the test set to represent all the possible cases evenly.

5.1.2 Evaluation metrics

In a regression problem, the aim is to predict a continuous value as the output. To evaluate the performance of a regression model, two commonly used metrics are Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). RMSE measures the average distance between the predicted and actual values, and it is calculated as the square root of the average of the squared differences between the predicted and actual values. RMSE is more sensitive to outliers compared to MAE, as it gives higher weights to larger errors. MAE measures the absolute average distance between the predicted and actual values, and it is calculated as the average of the absolute differences between the predicted and actual values. MAE is less sensitive to outliers and provides a more robust estimate of the model's performance. Both RMSE and MAE are useful metrics for assessing the performance of the regression models in our work. The average RMSE and average MAE metrics for the three experimental executions are reported in the next tables, taking into account the three different protocols used.

5.1.3 Training augmentations

Feature augmentation for neural networks involves generating additional training data by creating new features from the existing ones. This can help improve the performance of the artificial neural network (ANN) by increasing the quantity and diversity of the data available for training. There are several ways to perform feature augmentation, but one common approach is to apply transformations to the original data. Two possible augmentation techniques to our regression problem are adding Gaussian noise and shifting the input feature vectors. Adding Gaussian noise involves randomly generating noise from a normal distribution and adding it to the original vector to create a new data vector. This process helps to create more variability in the data and can make the model more robust to noise in the input data. Shifting the vector left or right involves moving the elements of a vector to earlier or later positions, creating a new vector with a different starting point. This process can help the model learn to be more invariant to small shifts in the input data. By applying these augmentation processes, we increase the amount of training data and so, we can improve the performance of the machine learning models and reduce the risk of overfitting to the limited training data. The first augmentation entails adding white Gaussian noise to the vector signal with Signal-to-Noise Ratio (SNR) values of 5dB, 10dB, 15dB, and 20dB. This generates four augmented feature vectors for every original sample. The second augmentation includes shifting the feature vector 5 and 10 elements either to the left or right by repeating border elements. This also generates four augmented feature vectors for every original sample. An example of the augmentations applied to one original input vector is demonstrated at the Figure 7.

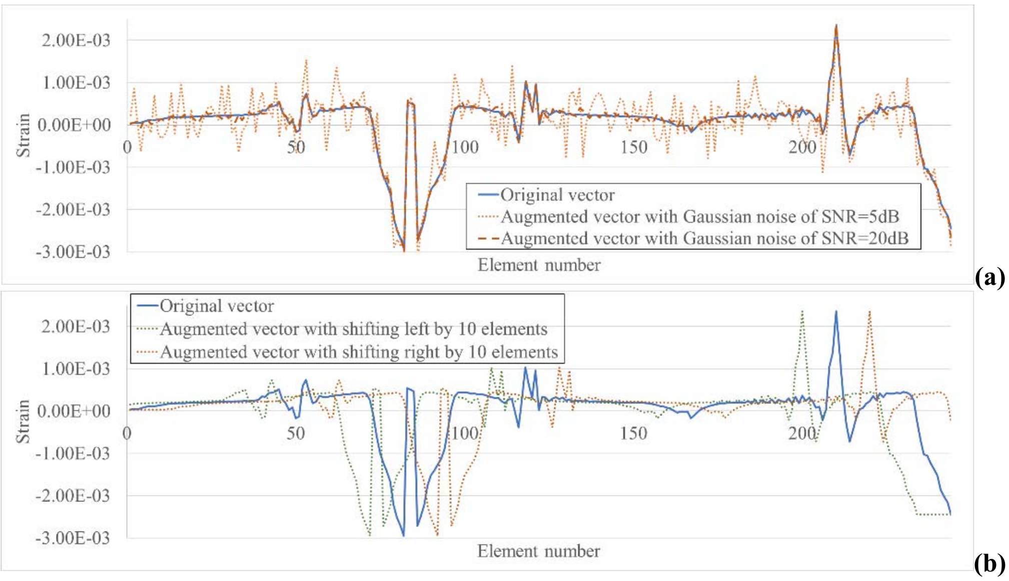

Figure 7: Feature augmentations include (a)adding white Gaussian noise and (b)shifting the original vector.

5.2 Architecture

Convolutional Neural Networks (CNNs) are a type of Artificial Neural Network (ANN), where their core building block is the convolutional layer that applies a set of learnable filters to a local region of the input signal. Each filter slides over the input signal, and at each position, it performs a dot product between its weights and the corresponding values of the input signal. This operation results in a feature map that highlights certain patterns or motifs in the input signal that are relevant to the task at hand. The convolutional layer is important because it enables the network to learn hierarchical representations of the input signal, where each layer captures more abstract and complex features than the previous one. Each layer takes the output of the previous layer as its input, resulting in a deep and powerful model that can effectively extract meaningful information from the input data.

We developed a custom CNN architecture for both regression tasks, which takes a feature vector of 244 strain values as input and outputs either impact location or damaged area. Our

CNN architecture includes several convolutions and max pooling layers, and does not use fully connected layers to reduce the number of parameters in the model. This results in only 1207 learnable parameters, achieved by using four convolution layers and two max pooling layers. Batch normalization layers and rectified linear unit (relu) functions are applied after each convolutional layer. Our proposed CNN architecture is summarized in Table 1.

Table 1: Summary of the CNN layers

Layer	Kernel Size	Parameters	Activation Size
Input	-	-	244 x 1
Convolution (Conv1)	20 x 1	Stride = 2, pad = 0	113 x 4
Convolution (Conv2)	10 x 1	Stride = 2, pad = 0	52 x 8
Max Pooling (Pool1)	4 x 1	Stride = 2, pad = 0	25 x 8
Convolution (Conv3)	5 x 1	Stride = 2, pad = 0	11 x 16
Max Pooling (Pool2)	3 x 1	Stride = 2, pad = 0	5 x 16
Convolution (Conv4)	5 x 1	Stride = 1, pad = 0	1 x 1
Output	-	-	1 x 1

The optimization process in our study involves minimizing the RMSE function through the use of Stochastic Gradient Descent (SGD) with Nesterov Momentum. A minibatch size of 64 and a momentum factor of 0.9 were utilized, with a weight decay of 10-4 as L2 regularization to prevent overfitting. We applied Xavier Glorot's initialization for the model weights while the initial learning rate was set to 10-3 and reduced by a factor of 10 every 600 iterations from the total of 1000 iterations during the training process to facilitate better convergence.

5.3 Results

Experimental results obtained from the proposed method are presented and discussed in this section. First, we present the experiments conducted with some baseline classifiers that were trained directly on the numerical strain vectors. Subsequently, we demonstrate the performance of the custom CNN architecture that exploits the augmentations during training.

5.3.1 Investigating baseline regression classifiers

We examined four common regression classifiers that predict the impact location or the damaged area, given a feature vector with strain values as input. The classifiers were trained on the 60-training data, while their hyperparameters were defined using the validation set. The results were calculated on the test set and are presented in Table 2 as the average values along the three experimental protocols. First, the Linear regression classifier models the linear relationship between input features and output, fitting a line to the data points that minimizes the sum of squared errors. Secondly, the exponential Gaussian Process Regression (GPR) is a non-parametric regression technique that models the input-output relationship as a Gaussian distribution. It uses the kernel function to measure the similarity between input feature vectors and predicts the output based on the similarity of the training data. Thirdly, the Support Vector Machine classifier (SVM) aims to find the hyperplane that best separates the data, with the Gaussian SVM applying the Gaussian kernel function to map the input feature vectors into a higher dimensional space. Finally, the Fully Connected Neural Network (FC-NN) is a classifier that uses three dense layers, connecting every neuron in one layer to every neuron in the next layer. The FC-NN used in this study consists of a first layer with 128 neurons and second and third layers each with 64 neurons, resulting in approximately 40,000 learnable parameters.

Despite their popularity and versatility, the above four regression classifiers did not achieve satisfactory performance on the given tasks. The linear regression model is limited by its assumption of a linear relationship between the input and output variables and fails to capture the complexity of the data. On the other hand, the GPR can suffer from overfitting when dealing with high-dimensional data, while the SVM with a Gaussian kernel function can model complex non-linear relationships, but still lead to large errors. Lastly, the FC-NN is a powerful model, but requires a large amount of training data to be effective. Overall, the poor performance of these models highlights the challenge of regression problems with high-dimensional input features and underscores the need for more advanced and specialized models such as the custom CNN proposed in this study.

Table 2: Comparisons of state-of-the-art classifiers applied on the feature vectors with strain values.

Metho d	**Impact Location Prediction**		**Damage Area Prediction**	
Classi fier	**avg RMSE**	**avg MAE**	**avg RMSE**	**avg MAE**
Linear	444.1667	286.6500	21040	13021.7
GPR	134.5467	94.9070	1586.6	1184.3
SVM	124.0343	88.8990	1970.7	1379.9
FC-NN	129.2300	68.2907	2580.6	1898.6

5.3.2 Investigating custom CNN for regression

CNNs use filters to extract relevant features from input data through convolution operations, allowing them to perform feature extraction and regression simultaneously. This makes them capable of learning complex hierarchical representations of input data and capturing relationships between input features and output variables. Our proposed custom CNN outperformed baseline regression classifiers on both tasks of impact location prediction and damage area prediction, as shown in Table 3. This can be attributed to the CNN's ability to directly process raw input data without extensive preprocessing, which is typically required by traditional regression classifiers. CNNs automatically learn relevant features at multiple scales and orientations, making them advantageous over traditional classifiers.

We evaluated two data augmentation techniques, shifting and adding Gaussian noise to the original strain feature vectors. Table 3 shows that shifting outperformed the Gaussian noise technique in both regression tasks. Shifting augmented the original examples, providing variations that were still informative for the task and increasing the model's ability to generalize to new data. The approach allowed the model to focus on specific patterns that distinguished different bird strike scenarios. In contrast, adding Gaussian noise did not significantly improve performance as it simply added white noise without altering the data's underlying structure. Combining both techniques led to decreased performance compared to using only the shifting augmentation, potentially because the added Gaussian noise was unnecessary. Nevertheless, using either of the augmentation techniques resulted in better performance than not using any augmentation at all, indicating the effectiveness of data augmentation in improving the performance of the model.

The custom CNN was trained on both tasks using the RMSE loss function to measure the difference between predicted and true output. Figure 8 displays the training and validation loss curves during training for both tasks, including original samples and their shifted versions. The loss significantly decreased throughout training, indicating the model was learning and improving. Figure 9 presents predicted vs. true response diagrams, showing closely aligned responses, indicating the model accurately predicted output for both tasks. These figures provide a comprehensive evaluation of the custom CNN's performance on both tasks.

Table 3: Comparisons of the proposed method using different augmentations during training the custom CNN.

Augmentations	**Impact Location Prediction**		**Damage Area Prediction**	
	avg RMSE	**avg MAE**	**avg RMSE**	**avg MAE**
-	64.722	44.94367	1695.667	1249
Gaussian noise	54.57467	29.24467	1231.333	915.3333
Shifting	**30.61633**	**18.70833**	**1118.333**	**814.6667**
Both	41.553	22.63633	1148.333	828.3333

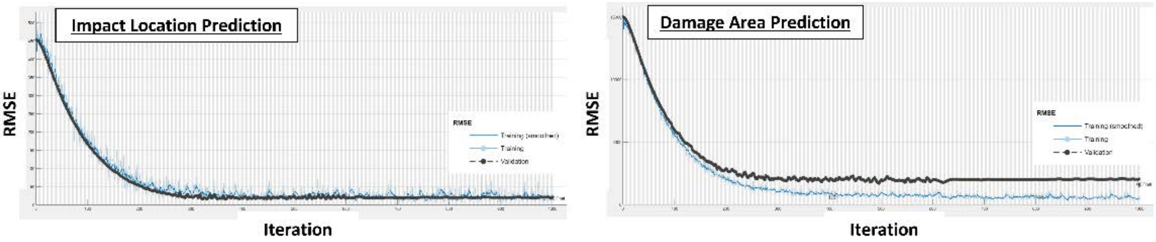

Figure 8: Monitoring the training progress via the training and validation loss curves for the task of impact location detection (Left) and the damage area detection (Right).

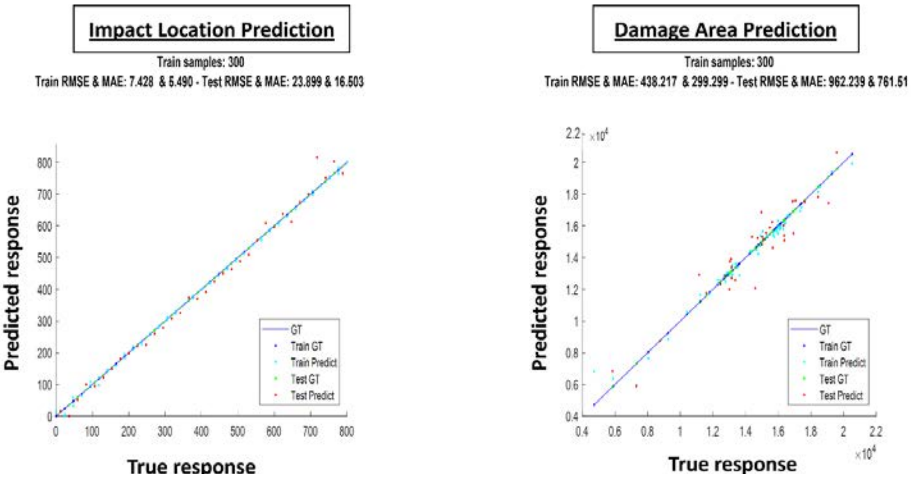

Figure 9: Evaluating the performance via the diagrams of predicted response against the true response (also called ground-truth (GT)) for the task of impact location detection (Left) and the damage area detection (Right).

6 CONCLUSIONS

In this work a DT which includes a FE model and ANNs has been developed. The ANN, for given strain data as input learn to predict the impact location as well as the damaged area of the adhesive caused by a bird strike. Additionally, from the implementation of the present study the following conclusions can be extracted:

- Simulation of a complex phenomenon by means of a FE model can produce valuable data in order to train an ANN which could be used for the monitoring of the status of critical parts during the operation of an aircraft.
- ANNs, depending on the training procedure can predict with relatively small discrepancies the impact location and the damaged area on a rotating blade. This could contribute to the optimization of the maintenance procedure.
- The proposed custom ANN captures the spatial dependencies present in strain vectors, allowing the model to effectively learn to extract relevant features from the input data.
- Data augmentations increase the quantity and diversity of the training data, leading to a more generalized behavior of the ANN and improved performance.

REFERENCES

[1] G. Yagawa and H. Okuda, Arch. Comput. Methods Eng., vol. 3, no. 4, pp. 435–512, 1996, doi: https://doi.org/10.1007/BF02818935.

[2] L. H. Hernandez-Gomez, J. F. Durodola, N. A. Fellows, and G. Urriolagoitia-Calderón, Appl. Mech. Mater., vol. 3–4, pp. 325–330, 2005, doi: 10.4028/www.scientific.net/AMM.3-4.325.

[3] S. J. Farley, J. F. Durodola, N. A. Fellows, and L. H. Hernández-Gómez, Appl. Mech. Mater., vol. 13–14, pp. 125–131, 2008, doi: 10.4028/www.scientific.net/AMM.13-

14.125.

[4] A. Luna-Avilés et al., J. Phys. Conf. Ser., vol. 305, no. 1, pp. 1–10, 2011, doi: 10.1088/1742-6596/305/1/012121.

[5] S. Lu, M. Jiang, Q. Sui, Y. Sai, and L. Jia, Compos. Struct., vol. 125, pp. 400–406, 2015, doi: 10.1016/j.compstruct.2015.02.038.

[6] A. Lamberti, G. Luyckx, W. Van Paepegem, A. Rezayat, and S. Vanlanduit, Sensors (Switzerland), vol. 17, no. 4, pp. 1–13, 2017, doi: 10.3390/s17040743.

[7] G. Sarego, M. Zaccariotto, and U. Galvanetto, IEEE Aerosp. Electron. Syst. Mag., vol. 33, no. 8, pp. 38–47, 2018, doi: 10.1109/MAES.2018.170157.

[8] A. Califano, N. Chandarana, L. Grassia, A. D’Amore, and C. Soutis, Appl. Compos. Mater., vol. 27, no. 5, pp. 657–671, 2020, doi: 10.1007/s10443-020-09829-z.

[9] G. Ferreira Gomes, J. A. Souza Chaves, and F. A. de Almeida, Mech. Syst. Signal Process., vol. 145, p. 106932, 2020, doi: 10.1016/j.ymssp.2020.106932.

[10] Z. Xu, T. Gao, Z. Li, Q. Bi, X. Liu, and K. Tian, Aerospace, vol. 10, no. 1, 2023, doi: 10.3390/aerospace10010066.

[11] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, Eng. Appl. Artif. Intell., vol. 117, no. November 2022, p. 105502, 2023, doi: 10.1016/j.engappai.2022.105502.

[12] K. Papadopoulos, I. Floros and Konstantinos Tserpes, , J. Phys. Conf. Ser., *Article in press*.