34th Congress of the International Council of the Aeronautical Sciences
ICAS
Florence
CON GRE SS
September 9-13
2024
ITALY

TRAIN ON SYNTHETIC - TEST ON REAL: DOMAIN ADAPTATION FOR STRAIN-BASED DAMAGE DETECTION ON AN AIRCRAFT WING

Philipp Conen[1], Florian Raddatz[1] & Gerko Wende[1]

[1]German Aerospace Center, Hein-Saß-Weg 22, 21129 Hamburg, Germany

Abstract

There is an increasing development towards sustainable aviation. Thus, there are multiple approaches to decrease emissions. One of the possible solutions is mass reductions of structural elements.

For these structural elements, mass reduction leads to an increasing need for precise maintenance. To fulfill this need, present research focuses on predictive or prescriptive maintenance. Nevertheless, predictive maintenance requires the as-is condition of a component for effective planning.

Therefore, the train on synthetic - test on real approach is considered for structural health monitoring of the components. The approach is examined on the prediction of crack-like damages on an aircraft wing using strain-based data.

Hence, a methodology is outlined that is used for the present paper. This methodology covers the development of a suitable test scenario. In the next step, a virtual and a physical representation of the scenario is built to generate a data set for each domain. Virtually, structural finite element model simulations are prepared and automated in Salome Meca using the Code_Aster solver. Physically, a test stand consisting of a cantilever with applied strain gauges is constructed. The objective is to train a machine learning model with the virtual data set and test it with the physical data set. For this, different supervised machine learning models from the Python library Scikit-Learn are compared.

The classification of a damaged or undamaged structure works well among all models and all data sets. For the regression of the damage position success is recognized inside the virtual and the physical data set. Training the models in the virtual and testing them in the physical domain leads to problems for this initial investigation. A similar behavior appears for the multi-output regression. This requires a deeper understanding of both data set characteristics as well as continuing to exploit further machine learning opportunities like neural networks and dedicated domain adaptation methods.

Keywords: Train on synthetic - Test on real; Sim to real; Domain adaptation; Supervised machine learning; Structural health monitoring

1. Introduction

Sustainable aviation is of growing interest to industry, politics, and society. The Paris Agreement [1] underlines these interests with the global motivation to limit global warming to $1.5\,^{\circ}$. In addition, the German Federal Government [2] sticks to the Paris Agreement, especially for transportation, which includes also the aviation sector. Furthermore, Destination 2050 [3] plans to reduce $50\,\%$ CO_2 emissions compared to 2005, with the future goal of net zero European aviation. Finally, the Air Transport Action Group adheres to the United Nations Sustainable Development Goals (SDGs) [4, 5]. In particular, SDG-12 (Responsible consumption and production) and SDG-13 (Climate change) are highly relevant for the purpose of this paper. For SDG-12 the state of the art is a recycling rate of $85\,\%$ to $90\,\%$ of an aircraft (measured by weight) [4]. With regard to SDG-13, CO_2 today's emissions are already $54\,\%$ lower than in 1990 [4].

Several strategies and technologies are present to achieve these goals [6]. First, sustainable aviation fuels emit less CO_2 than kerosene [7]. Second, hydrogen only emits water and no CO_2 [8]. Finally, mass reduction lowers the demand for fuel [6].

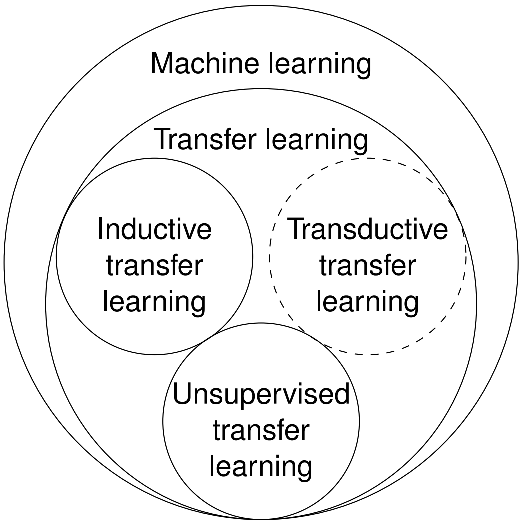
Machine learning
Transfer learning
Inductive transfer learning
Transductive transfer learning
Unsupervised transfer learning

Figure 1 – Correlation of machine learning, transfer learning, unsupervised transfer learning, inductive transfer learning, and transductive transfer learning respectively domain adaptation inspired by [9].

This work focuses on the scenario of mass reduction. Mass reduction is challenging as engineers push the structures closer to their physical limits. As a result, they possibly become more vulnerable or less damage-tolerant. Therefore, new methods and concepts are necessary for maintenance, repair and overhaul.

Predictive and prescriptive maintenance are an essential part of these new concepts. Only with additional as-is information, these concepts can work accurately. To know the exact condition of a structure, structural health monitoring is required. Current research on structural health monitoring or damage detection relies on data-driven approaches. However, data that represents damage is rare. A possible solution to gather data is the train-synthetic-test-real approach [10] because simulation data is easy to access [11].

The research in this paper investigates the train on synthetic - test on real approach for a specific application example. The use case is the prediction of crack-like damages on an aircraft wing using strain-based data for a supervised machine learning model. For this, a suitable test case is required with less complexity. As a result, a well-known cantilever is chosen for initial investigations. Furthermore, a virtual and a physical model of the test scenario is necessary. In the next step, virtual and physical data sets are generated with the corresponding model. The approach provides for training with the virtual data set and testing with the physical data set. The key outcome of the resulting model is the prediction of the damage existence, position, and width.

2. State of the art

The subcategory for the above mentioned approach is also known as transfer learning. It is a non-classical machine learning challenge since the source and target data do not share the same domain. Possible domains are the feature space, label space or distribution space. [9, 11, 12, 13]

Transfer learning is categorized into inductive transfer learning, transductive transfer learning, and unsupervised transfer learning (see Figure 1). Unsupervised transfer learning is the case if source and target distributions as well as the tasks from the source and target domain, are not matching. In inductive transfer learning, the source and target distributions are matching, but the tasks from the source and target domains do not. For transductive transfer learning, the tasks from the source and target domain are matching, but the source and target distributions are not, which is the case for the investigations in the present paper. It is also known as domain adaptation. [9, 12, 13]

Following, a literature review is done for the fields of domain adaptation and crack detection using Scopus. Here it is to remark that domain adaptation and domain adaption partly are used

Table 1 – Confusion matrix for the search strings: ML (Machine learning), TL (Transfer learning), DA (Domain adaptation), SHM (Structural health monitoring), DD (Damage detection), and CD (Crack detection) on Scopus with standard settings in December 2023.

	ML	TL	DA	SHM	DD	CD
ML	634006					
TL	24445	103591				
DA	3268	4382	42145			
SHM	1692	242	64	44349		
DD	4432	718	90	12505	114048	
CD	899	269	16	3131	10153	29048

incorrectly. The dominant version is domain adaptation. As a starting point, Table 1 gives a general overview of the related research fields. Ongoing to the general overview, the search string *(TITLE-ABS-KEY (domain AND (adaptation OR adaption) AND (crack AND detection)))* on Scopus delivers 16 publications. Nevertheless, none of these publications depicts the work presented in this paper. From a methodology perspective, eight publications refer to unsupervised learning [14, 15, 16, 17, 18, 19, 20, 21] and three to self-supervised learning [17, 21, 20]. Some of these authors see self-supervised learning as a subcategory of unsupervised learning. From the application side, ten publications contain image recognition [22, 23, 17, 14, 20, 24, 18, 15, 25, 21] and four to vibrations, acoustics or guided waves [26, 16, 27, 19, 28]. In addition, only three of 16 publications handle the domain shift from the synthetic domain to the physical domain [28, 18, 19]. Xiang et al. investigate rotor fault diagnosis based on vibro-acoustics [19]. Wang et al. evaluate fatigue crack detection based on lamb waves [28]. Lin et al. work on crack detection for roads based on image recognition [18]. As a result, there is still a research gap for domain adaptation in crack detection using synthetic strain data.

The review above returns only a few cases for training with synthetic data. Thus, the search string gets further constraints and extensions. The refined search string results in *(TITLE-ABS-KEY (domain AND (adaption OR adaptation) AND ((crack OR damage OR fault) AND detection) AND synthetic))*. This search returns eight publications. Nevertheless, there is agreement on the necessity of the train-synthetic-test-real approach [29, 30, 31]. Ha et al. [30] detect faults of a planetary gearbox using deep learning and vibration data. Ghorvei et al. [31] use unsupervised learning for bearing fault diagnosis while Wang et al. [29] use deep learning. Difficulties for training machine learning models for structural health monitoring are different operating conditions [30]. When dealing with synthetic data, the data is often not realistic enough [29]. Furthermore, global and local distribution gaps have to be distinguished [31]. Another problem is the lack of damage data [30, 29], especially when it is infeasible, expensive, or for safety critical systems [29]. Synthetic data overcomes this problem since it is easily accessible and pre-labeled [29]. In addition, it is less expensive than generating big data sets of experimental data. There are two outstanding works for an overview of domain adaptation and transfer learning that are not included in the above research strings. These are Redko et al. [9] and Kouw et al. [12].

When dealing with domain adaptation, generalization is a crucial task. Generalization describes the complexity and flexibility of a classifier. Models can be overfitted, underfitted, or well-fitted. Overfitted models can not generalize to new data. In this case, regularization is a solution. Nevertheless, high regularization can also lead to underfitted models. [12]

For generalization, it is essential to identify a suitable training set and include as much realistic features into the training data as possible. Possible features are correlated and uncorrelated noise. Linear operators like crosscorrelation, autocorrelation, and convolution can introduce these features into a data set. [11]

The challenge of generalization can also be expressed as the probability distribution of the source and target domains which are not equal.

$$P_\mathrm{s}(x_\mathrm{s}, y_\mathrm{s}) \neq P_\mathrm{t}((x_\mathrm{t}, y_\mathrm{t}) \tag{1}$$

P refers to the probability distribution, x to the feature space, and y to the label space of the source s and the target t. One possible solution for the generalization issue is the subspace mapping

$$P_\mathrm{s}(T(x_\mathrm{s}) = P_\mathrm{t}((x_\mathrm{t}) \tag{2}$$

with a suitable transformation T. Neural networks can find such transformations [32, 33, 34]. [11]

To conclude, research is already done in the field of domain adaptation and damage detection, but it still lacks in amount, depth, and application.

3. Methodology

This section outlines the methodology that is followed in the present paper. First, the process towards the train on synthetic - test on real approach is shown. Second, the data gets explained and categorized into features and labels. Finally, the architectures of the different machine learning models that come into use for this project are outlined.

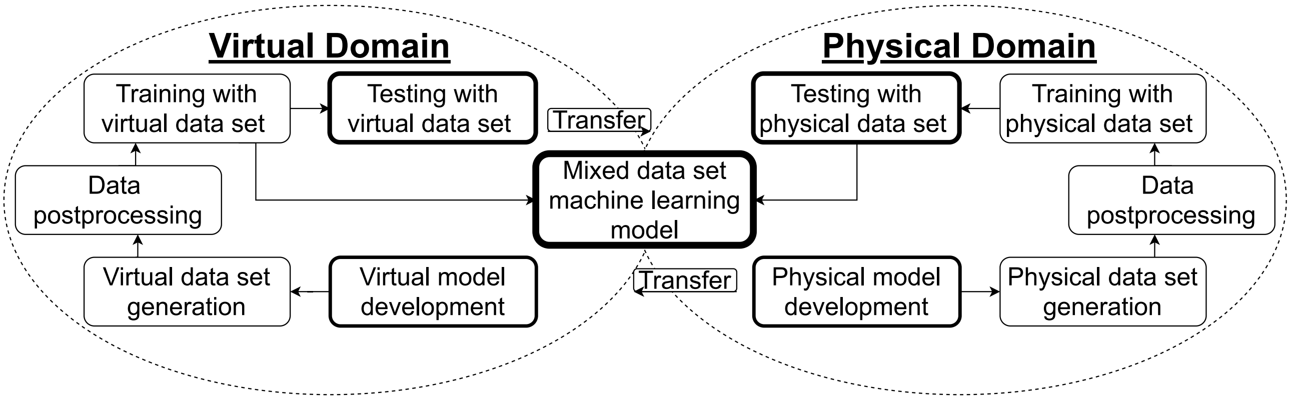
Virtual Domain
Physical Domain
Training with virtual data set
Testing with virtual data set
Transfer
Testing with physical data set
Training with physical data set
Data postprocessing
Mixed data set machine learning model
Data postprocessing
Virtual data set generation
Virtual model development
Transfer
Physical model development
Physical data set generation

Figure 2 – Process description for a train on synthetic - test on real approach.

3.1 Process description

Figure 2 displays a general description of the train on synthetic - test on real approach that is followed in the present paper. The starting point is the design of a suitable test scenario (section 4.) with the possibility of modeling an equal physical and virtual representation. Next, this requires the development of a simulation model (subsection 4.2). The simulations can be anything reproducible in the physical domain. In the specific case of this work, these are strain-based structural simulations, that generate data for different load-damage-scenario. Load-damage-scenario means different combinations of loads or displacements under different damage characteristics. To prove the independence of the data against the mesh, a sensitivity analysis is required (see subsubsection 4.2.2). With the existence of a valid simulation model a parameter study needs to be done to generate a data set (see subsubsection 4.2.3) that contains the features and labels (see subsection 3.2). In most cases, the data needs some kind of post-processing after the generation phase. At least this means conversion into suitable formats, normalization of data, interpolation between data points, and concatenation of single result files to a common database. The fixed locations of the physical sensors require interpolations between some virtual data points to have matching data points in both domains. Equal work is required for the physical domain (see subsection 4.1).

With the existence of a common and complete data set for both domains, a pre-analysis for both data sets is carried out (see section 5.). This contains data analysis on the one hand and training of machine learning models on the other hand. The machine learning models are trained inside their corresponding domain first, to predict the damage existence, position, and width. In the transfer phase, the machine learning model is trained with the virtual and tested with the physical data set (see section 6.). The evaluation is split into multiple stages. First, a classification model predicts the existence of damages (see subsection 6.1). Second, a regression model predicts the position of a damage (see subsection 6.2). Finally, a multi-output regression model predicts the position and the width of the damage (see subsection 6.3).

All models face a performance evaluation. The evaluation metrics are the accuracy, precision, recall, and F1 score for the classification models as well as the R2 score, the mean squared error, and the mean absolute error for the regression models. Assuming a perfect scenario, no additional changes are required inside the synthetic machine learning model. In reality, a perfect scenario seems unlikely, due to noise and environmental conditions. This is why the train on synthetic - test on real models are compared to the models that are trained in their corresponding domain. The focus of the present paper is on supervised machine learning models only, since the labels are well-known from the simulations.

3.2 Data

This subsection explains the data that is used for the training and testing of the machine learning models. The data is further categorized into features and labels. For the train on synthetic - test on real approach it is important to have equal features and labels in the physical and virtual data sets. Table 2 gives an overview of the features and labels used in both data sets.

3.2.1 Features

The main features are the strain values from discrete points. These can be physical sensor values or virtual sensor values. The physical sensor points have a fixed location since they are applied permanently. To match the virtual sensor points with the physical sensor points, interpolation between two mesh nodes of the virtual model is required for some of these points. In this paper six features of this type are available theoretically (see subsubsection 4.1.2). Practically, only three of them are relevant yet, since for constant damage length of $80\,\mathrm{mm}$ (cantilever width) there will be no difference in the sensor values of row A and row B. Further restrictions for the strain features are the in-existence of damages on sensor positions for the physical domain since this would destroy the sensors. As a result, these damage positions are excluded from the synthetic data set, since this could influence the results of the later machine learning model. Additional features could be the load and the displacement. For linear structural mechanics, load and displacement are strongly interconnected. This means, that for machine learning models that represent a linear structural mechanics problem, there shouldn't be any benefit using both features. Above all, measuring very small displacements of multiple points is a non-trivial task in the physical domain. In contrast, the load is easy to estimate in the physical and the virtual domain. Thus, for this paper, the load is considered as a feature, while the displacement is out of scope. When it comes to non-linear material behavior, the material characteristics change. For this case, it could be a potential benefit to include displacement as an additional feature.

3.2.2 Labels

Possible damage characteristics are the existence, the position, the width, and the length. The present paper tackles the prediction of the existence, the position, and the width (see Table 2). The prediction of the length is out of scope. Regarding the damage existence, classification is used. The damage position and damage width are predicted using a regression model. Corresponding to these two methods the data sets (see Table 2) are reduced to the relevant labels only.

4. Test scenario

As pointed out in section 3., the first step is to find a suitable test scenario that can be equally represented in the physical and virtual domains. The first part of the test scenario requires an investigation object. To use the train on synthetic - test on real approach for damage detection on an aircraft wing, a strongly simplified representation of an aircraft wing is developed. For this initial investigation, a bending beam in the form of a rectangular aluminum cantilever is chosen. This eliminates geometrical and material complexity. Table 3 shows the material properties (left) and the geometrical properties of the cantilever (right). At this point, it is to remark that this scenario isn't a realistic in-flight scenario since the aerodynamic loads would lead to an upward bending instead of an additional downward bending as presented here. An upward bending would result in much more experimental effort within the given boundary conditions. Nevertheless, the downward bending is still a relevant scenario, e.g. during refueling. A corresponding application in the real world would benefit the eliminated complexity of in-flight structural health management. Moreover, it would promote the approach of human centricity. The bending beam itself is a well-known structural scenario and is analytically solvable. The validation of the numerical model takes place with the analytical displacements of a gravitational load with an additional single load. The maximum displacement under gravitational load

$$f_{\mathrm{GL}} = \frac{\rho \cdot V \cdot g \cdot l^3}{8 \cdot E \cdot I} \tag{3}$$

with the gravitation $g = 9.81\,\mathrm{m\,s^{-2}}$, the Young's modulus E, the area moment of inertia I, the density ρ, the volume V, and the length of the cantilever l_{c} [36]. The maximum displacement under a single

Table 2 – Overview of the features and labels used in the physical and virtual data set.

Load	A-0	A-1	A-2	Damage existence	Damage position	Damage width
..	..	..	..	..	..	..

Table 3 – Geometrical and material properties of the EN AW-6060 cantilever [35].

Material characteristics	Value	Unit	Geometrical characteristics	Value	Unit
Density	2.7	$\mathrm{kg\,dm^{-3}}$	Length	550	mm
Young's modulus	$70\,000$	$\mathrm{N\,mm^{-2}}$	Width	80	mm
Tensile strength	$215-260$	$\mathrm{N\,mm^{-2}}$	Height	8	mm
Magnesium	0.5	$\%$	Volume	$352\,000$	$\mathrm{mm^3}$
Silicon	0.5	$\%$	Area moment of inertia	3413.3	$\mathrm{mm^4}$
Iron	0.2	$\%$			

load

$$f_{\text{SL}} = \frac{F \cdot l^3}{3 \cdot E \cdot I} \tag{4}$$

with the force F, and the lever arm of the force l_{f} [36]. The maximum displacement under gravitational load and single load

$$f_{\text{CL}} = f_{\text{GL}} + f_{\text{SL}} \tag{5}$$

[36]. The area moment of inertia

$$I = \frac{b \cdot h^3}{12} \tag{6}$$

with the width b and thickness h of the cantilever [37]. Under the conditions mentioned in Table 3 and a single load of $9.81\,\mathrm{N}$ that is applied at a lever arm of $540\,\mathrm{mm}$ from the clamping this results in a displacement of $f_{\text{CL}} = 2.849\,\mathrm{mm}$.

The second part of the test scenario requires a damage object. The initial idea of this damage is a crack. For natural cracks, there are two main problems. First, natural cracks with certain characteristics are hard to reproduce, since they occur randomly as a result of overload or fatigue. Second, cracks are irreversible damages. This would require multiple cantilevers for testing, which means high effort. Thus, reversible artificial crack-like damages are introduced. For this, rectangular profiles are placed underneath the cantilever, which act as additional support trestle. This increases the local stiffness instead of decreasing it like natural cracks would do. The rectangular profiles have a constant length that matches the cantilever widths ($80\,\mathrm{mm}$). Possible damage widths are $3\,\mathrm{mm}$ and $5\,\mathrm{mm}$. The artificial damages are made of the same material as the cantilever (EN AW-6060) to avoid damages, either on the cantilever or on the artificial damages.

The following section is split into subsections for the physical and the virtual representation of the test scenario. Physically, this contains a test stand. Here, the construction, the data acquisition system, the data generation as well as measurement uncertainties and error sources are pointed out. Virtually, the development of the numerical model, a sensitivity analysis, and the data generation are described.

4.1 Physical model: Test stand

The following subsection covers the physical representation of the test scenario. This contains the construction, the data acquisition system, the data generation of the test stand as well as measurement uncertainties and error sources.

4.1.1 Construction

Figure 3 shows a schematic of the test stand. It is built out of aluminium profiles. A robust ground plate works as the baseline of the test stand. On the left side of the ground plate, there is an equally robust support mounted. The robustness is essential to eliminate errors that occur from the bending of the test stand itself. The clamp fixes the cantilever on the support. An adapter on the free end of the cantilever allows applying loads without harming the cantilever. The adapter has a length of $80\,\mathrm{mm}$ (cantilever width) and a width of $20\,\mathrm{mm}$. Thus, the center of the load is at $540\,\mathrm{mm}$ from the clamping. In addition, the ground plate carries two rows of linear guides of type MRT30V from Rollon [38]. This allows the two wagons to slide underneath the cantilever. The wagons carry a stack of

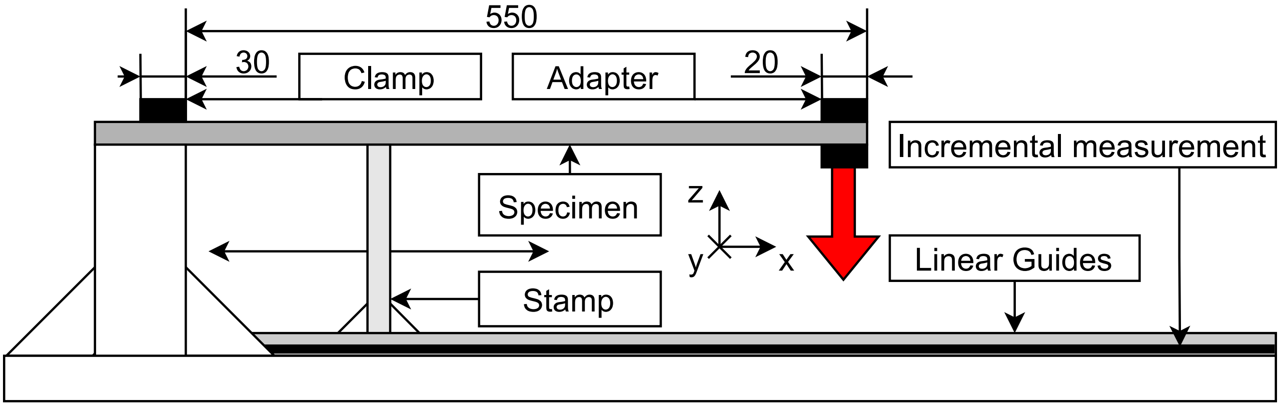

Figure 3 – Test scenario: Cantilever with fixed clamping, support as well as gravitational and single load.

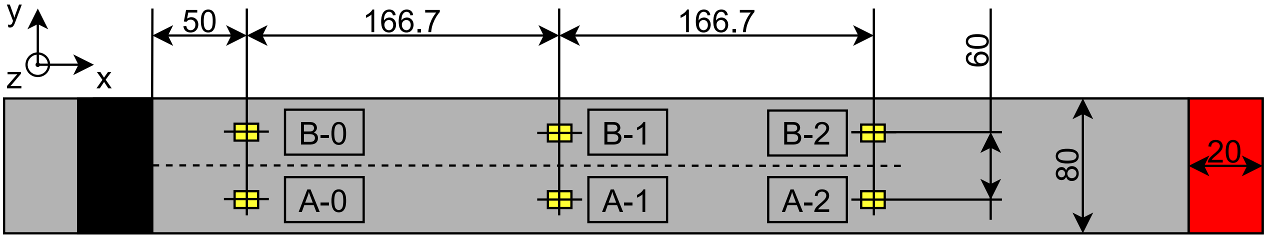

Figure 4 – Cantilever with applied strain gauges in yellow, with the clamp location in black, and the adapter in red.

aluminium profiles (stamp). On top of these profiles, there is an additional assembly that allows to fine adjust the height of the artificial damages to the exact height of the cantilever. This assembly also allows to exchange the artificial damages in y direction. That is necessary since sliding the stamp in x direction would destroy the applied sensors on the lower side of the cantilever. Furthermore, it allows the use of different damage widths. At the edge of the linear guides, a magnetic band is applied on the ground plate.

4.1.2 Data acquisition system

As known from subsection 3.2, there are four features and three labels to span the required data set for the investigations. The most important features are the strain gauges. The criteria for choosing the right strain gauges are the uni-axial strain condition, the measurement of precise local stresses instead of averaged global stresses, the homogeneous material properties, and the laboratory conditions. This results in sensors of type K-CXY3-0030-3-350-4-030-N from HBM (see Figure 4) [39]. This sensor type covers two measurement grids that are rotated by $90°$ to each other on one carrier foil (see Figure 5). Figure 4 displays the locations of the different sensors on the cantilever. The letters A and B indicate the distribution in y direction, while the numbers reaching from 0 to 2 indicate the distribution in x direction. The strain gauges are applied on both sides, in two rows and three columns over the cantilever. In total, this results in 12 strain gauge carrier foils and 24 single strain gauge measurement grids. Each pair on the upper and lower sides of the cantilever is interconnected as a resistive full-bridge that eliminates everything except the bending strain from the main load direction. This results in six strain values for the sensors A-0, A-1, and A-2 as well as B-0, B-1, and B-2. The strain values from row B are not relevant for this investigation. Since there are only damages that correspond to the full width of the cantilever, there won't be any differences in strain between strain values from row A and row B. Nevertheless, the second row is necessary for further investigations of the damage length. These are not covered within the present paper but are still considered for the cantilever already. The acquisition itself takes place with the measurement amplifier MX1615-B from

Characteristic	Value	Unit
Measurement grid length	3	mm
Measurement grid width	3.3	mm
Carrier length	10.9	mm
Carrier width	7.6	mm
Nominal resistance	350	Ω
Max. voltage	10	V
k-factor	2	-
Temperature response	23×10^{-6}	ppm/K
Measurement grid material	Constantan	-
Carrier foil material	Polyimide	-

Table 4 – Characteristics of K-CXY3-0030-3-350-4-030-N T-rosette strain gauge from HBM [44].

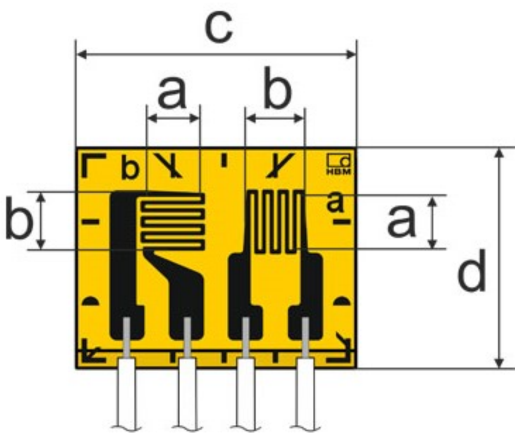

Figure 5 – K-CXY3-0030-3-350-4-030-N strain gauge from HBM [39].

HBM [40]. The last feature is the load that is applied at the adapter (see red elements in Figure 3 and Figure 4). At this stage, the load is considered as a static weight. The options for the load feature include 0 kg, 0.5 kg, and 1.0 kg. The acquisition of this information takes place manually.. Regarding the labels, the damage position and the damage width are relevant. A magnetic band allows the extraction of the position of the artificial damage by incremental measurements with a resolution of 0.01 mm. The magnetic band is of type MB20-50-10-1-R from ELGO [41]. In addition, the LMIX2 sensor from ELGO evaluates the position [42]. Finally, the Z50 module from ELGO displays the values on a screen [43]. The digital interface isn't used yet. Thus, the acquisition is done manually at this stage. The last label is the crack width. The crack width depends on the artificial crack that is inserted into the assembly on top of the stamp. Since the insertion is done manually, the acquisition takes place manually, too.

4.1.3 Data generation

To generate the physical data set, experiments are made with different combinations of features and labels. Since the data set will be small compared to the virtual data set it is important to have random values on the one hand, but a good distribution on the other hand. To generate a single value of one load-damage-scenario the strain values are recorded for 1 s with a sampling rate of 4.8 kHz. The resulting single value is the mean of all the 4800 values. A slight delay between adjusting and measuring guarantees to eliminate vibrations. The adjusting and recording of all load-damage scenarios is done as follows. A zeroing of the sensor can not be done by lying the cantilever flat on the ground. The reason is that this would first, possibly harm the sensors and second, result in an inaccurate zero state. As a result, the sensors are zeroed with the cantilever fixed on the test stand, but without additional load and no damage applied. Next, the stamp with the artificial damage of 3 mm width is positioned randomly at the free end underneath the cantilever. After taking all the required information, the stamp is positioned four more times in the first section between the free end of the cantilever and the sensor A-2. This is repeated for the section between the sensors A-2 and A-1 as well as A-1 and A-0. There are no values for the section between the sensor A-0 and the clamp since it was not possible to place the artificial damage underneath the cantilever due to the pre-load force of the artificial damage. This procedure is repeated for all combinations of loads and crack widths. Due to completeness, sensor values are taken also for all load cases, without applying damages. The final post-processing takes place in Python. It includes the extraction of the mean of each file which represents a load-damage-scenario. All features and labels are stored in a single data file.

4.2 Virtual model: Numerical simulation

The following subsection covers the virtual representation of the test scenario. This contains the development of the geometry, the mesh and the numerical model, the mesh automation, a sensitivity analysis, and the data generation.

4.2.1 Development of the model

The virtual model is a physically accurate reproduction of the physical model. For the creation of the geometry and the mesh the software Salome is used. Salome offers the possibility of dumping studies. This functionality generates Python scripts based on the inputs inside Salome. The development of the finite element model takes place in Salome Meca which is a wrapper for the solver Code_Aster. Salome Meca supports an equal functionality to dump the study. With both scripts, further automation is made for the sensitivity analysis and the parameter study.

As a starting point, a three-dimensional geometrical model is built. For this, multiple points are defined. From these points, further geometrical objects like lines, faces, and a volume arise. In addition, groups of lines, faces, and the volume are created The groups are required for the later mesh and the finite element model. Geometrical inputs are the overall dimensions of the cantilever, the load face, and the reaction face. The damage is numerically represented as an additional fixation in z direction (support trestle), that is applied on the reaction face. The geometrical groups split into one single group for the volume, groups of faces for the load face, the reaction face, and the fixation face as well as groups of edges for all edges in x direction, all edges in y direction, all edges in z direction, and an edge along the x axis of the sensors of row A.

The geometry is used as input for the mesh. Like the geometry, the mesh is also three dimensional. Initial settings are hexahedron mesh conditions for the third dimension and mapped quadrangle mesh for the second dimension. For the boundaries of this research, a quadrangle mesh is essential for a smooth and accurate numerical representation of the damage. Regarding the settings for the first dimension, wire distances in mm are defined for the groups that contain all the edges in x, y, and z directions. As for the geometry, groups are defended for the mesh. Salome imports all existing groups automatically. Additionally, some groups are converted to groups of nodes. These are groups that are not meant as an option for boundary conditions of the finite element model, but for the export of strain data, e.g. the edge along the x axis of the sensors of row A.

The mesh is the first essential input for the numerical model. This requires the Code_Aster functionality LIRE_MAILLAGE. In the next step, the model is linked to the mesh (AFFE_MODELE). This contains the 3D element modeling method and the mechanical phenomena. The definition of a material (DEFI_MATERIAU) requires the Young's Modulus, Poisson ratio, and the density (see Table 3. Now the material is linked to the model (AFFE_MATERIAU). Essential boundary conditions (AFFE_CHAR_MECA) are displacement impositions in x, y, and z direction for the fixation face as well as displacement impositions in z direction for the reaction face. Natural boundary conditions are a negative gravitational load of $9.81\,\mathrm{m\,s^{-2}}$ and (initially) a pressure of $6131.25\,\mathrm{N\,m^{-2}}$ that acts as a single load. The gravitational load is assigned to the complete model, while the pressure is only assigned to the load face. In the context of the parameter study, the pressure can take different values. Next, the model, the material, and the boundary conditions are assigned to a static mechanical analysis (MECA_STATIQUE). Code_Aster calculates the displacement automatically. The calculation of the strain requires additional post-processing (CALC_CHAMP). Both, the displacement and the strain, are the raw values of the calculation nodes (ELNO). Regarding the strain, the only relevant component is the bending strain in x direction (EPSI_XX) For the extraction of strain and displacement, the data is converted into table format (POST_RELEVE_T) and exported to a .csv file (IMPR_TABLE). Finally, the entire data is exported to a .rmed file. This allows to open the results in a post-processor like ParaVis to view the results directly on the geometry. The file is not necessary for the later data generation, since this leads to unnecessary high data storage. Nevertheless, it is helpful for investigations during the development phase.

4.2.2 Sensitivity analysis

Sensitivity analyses guarantee accurate results under the condition of acceptable computational effort. At a certain point, relevant parameters e.g. the maximum displacement become independent of the cell sizes. This point indicates suitable mesh options. The sensitivity analysis takes the gravitational load and the maximum single load of $6131.25\,\mathrm{N\,m^{-2}}$ into account. Damages are not covered in the sensitivity analysis. Available mesh options reach from $0.5\,\mathrm{mm}$ to $10\,\mathrm{mm}$ in x direction, $0.5\,\mathrm{mm}$ to $8\,\mathrm{mm}$ in y direction, and $1\,\mathrm{mm}$ to $4\,\mathrm{mm}$ in z direction. In total, this results in 75 possible mesh options.

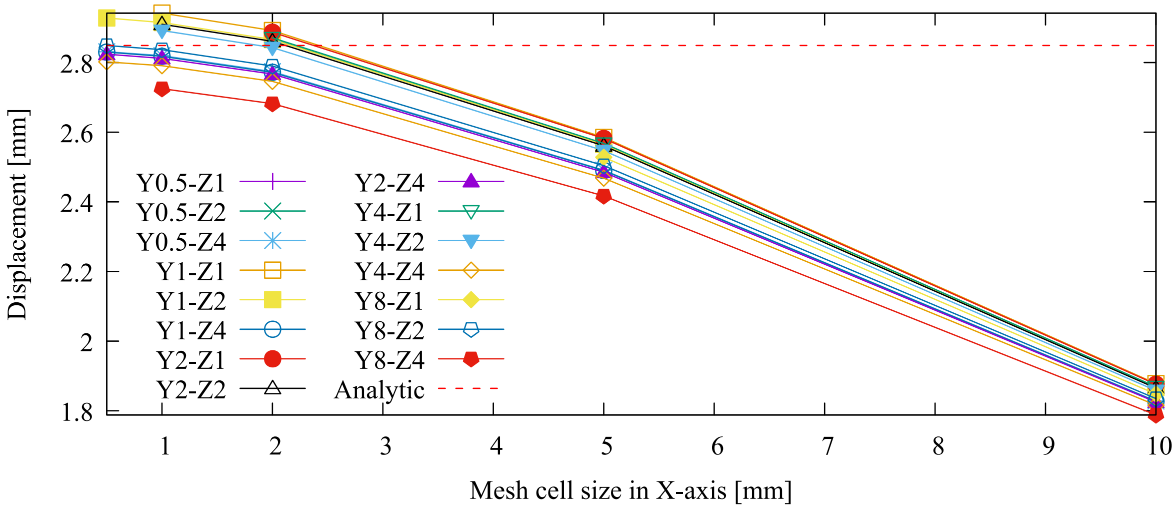

Figure 6 – Sensitivity analysis for a cantilever of size $550\,\mathrm{mm} \times 80\,\mathrm{mm} \times 8\,\mathrm{mm}$ with gravitational load and single load of $6131.25\,\mathrm{N\,m^{-2}}$.

Because of the high amount of possible mesh options, the sensitivity analysis itself as well as the later parameter study requires automation for meshing and solving. As mentioned above, Python can automate the generation of geometries and meshes with dumped scripts from Salome. The dumped script contains three additional variables, that allow an easy manipulation of the cell sizes in the three dimensions x, y, and z. An additional function exports the mesh in the .med format and with a representative file name to a desired directory. The simulations can be run from a lightweight sub-tool of Salome Meca called ASTK. After exemplary preparing the .csv, .comm, .export, .med, .mess, and .rmed files as well as the .base folder with ASTK, the simulation can be run also from a Linux shell. For this, additional software is written to automate the generation of all the cases in the form of folders with the required .comm, .export, and .med files. After the generation, the script executes all simulations one after another. Finally, the script post-processes the resulting .csv files and extracts the maximum displacements at the free end of the cantilever. Furthermore, it extracts the strains for all calculation nodes along the x axis of the sensor row A for the synthetic data set. For the sensitivity analysis, the maximum displacements are concatenated with the cell sizes of the mesh in each dimension.

Figure 6 shows the sensitivity analysis. In contrast to common sensitivity analysis, there isn't only one global variable for the cell sizes in all dimensions, but each for one single dimension. As a result, the displacements are plotted over the mesh size in x direction. Multiple lines indicate the mesh options in y and z direction. Additionally, a horizontal line introduces the analytical solution mentioned in section 4. . There aren't results for every mesh combination. The reason for this is increasing residuals in some simulations that cause errors when reaching a certain criterion. In the present paper, the criterion takes a value of 1×10^{-5}. An example are the highly accurate meshes with cell sizes of $0.5\,\mathrm{mm}$ in y direction. Only the combination with a cell size of $4\,\mathrm{mm}$ in z direction delivers good results for fine cell sizes in x direction. As expected, a general trend is lower displacements for courser cell sizes. This has the most influence for cells in x direction, medium influence for cells in z direction, and less influence for cells in y direction. In general, acceptable convergence is present for x cell sizes of $\leq 1\,\mathrm{mm}$ A closer look at the displacements for x cell sizes of $0.5\,\mathrm{mm}$ to $2\,\mathrm{mm}$ shows that there are three clusters. One cluster underestimates the analytical solution, while another clearly overestimates the analytical solution. The results from the overestimated cluster show unexpected behavior. It contains mesh options with cell sizes of $1\,\mathrm{mm}$ and $2\,\mathrm{mm}$ in z direction, which should deliver the most accurate results. This could indicate problems either with the analytical solution or with the model. The third cluster reaches the analytical solution most closely for x cell sizes of $0.5\,\mathrm{mm}$. The mesh options with the best performance are cell sizes of $0.5\,\mathrm{mm} \times 8\,\mathrm{mm} \times 2\,\mathrm{mm}$ in x, y, and z direction. In contrast to the best mesh options mentioned above, another mesh is chosen for the data generation phase, because the simulations failed for most load-damage scenarios. There are two possible reasons for this. First, there is the damage-based non-linearity that comes into play, since damages have not been covered during the sensitivity analysis. This non-linearity leads to high residuals. Second, there is the aspect ratio, which is a geometrical bias of a cell face. Since the aspect ratio of the best mesh has a four times higher aspect ratio (16) compared to the now chosen mesh, this is a possible reason for the high residuals. As a note, the recommended aspect ratio is ≤ 10 [45]. The chosen mesh options for the data generation are $1\,\mathrm{mm} \times 4\,\mathrm{mm} \times 2\,\mathrm{mm}$. Nevertheless, noise and variance between the synthetic and physical data sets are omnipresent. Thus, for these initial experiments, the high-end accuracy of the numerical model is of less importance. This potential source of variance between data sets of both domains can be seen as an additional robustness test for the train on synthetic - test on real approach.

4.2.3 Data generation

In comparison to the sensitivity analysis, the parameter study for the synthetic data set generation needs further automation. This includes two adaptations. First, additional loads are considered as mentioned in subsection 4.1. Valid options are $0\,\mathrm{kg}$, $0.5\,\mathrm{kg}$, and $1\,\mathrm{kg}$. The gravitational load remains the same. This adaptation requires changes in the overlaying wrapper, to generate additional case structures. The second adaptation is about the reaction face. Since the overall goal is to predict the position and the width of a damage, the damage needs to be shifted along the x axis of the cantilever. Additional variables are the damage width and the damage position. A function loops over the variables and spans a reaction face in the geometry and the mesh at a certain position and with a certain width. As for the physical model, the options for the damage widths are $3\,\mathrm{mm}$ and $5\,\mathrm{mm}$. Regarding the damage position, the damage gets shifted every $1\,\mathrm{mm}$ from the clamping to the free end of the cantilever. The only exceptions are locations of applied sensors as pointed out in subsection 3.2. In total, this results in 2345 combinations where two of them represent an undamaged scenario with $0.5\,\mathrm{kg}$ and $1\,\mathrm{kg}$ single load. The case with gravitational load only acts as a baseline.

5. Data pre-analysis

The following section introduces a pre-analysis of the synthetic and physical data sets. It covers general data analysis, but also investigations that focus on physics, like the distributions of the strain data investigated among features and labels. This shows the feasibility of distinguishing damage positions and damage width based on the strain data. Correlations and differences between the synthetic and physical data sets are also under investigation.

A first investigation shows that there are fewer cases than expected for the virtual data set. The reason for this are high residuals for simulations with a damage that is located near the clamp (see subsubsection 4.2.2).

Second, a correlation analysis between features and labels in the form of heat maps is done with the Seaborn Python library. The heat maps for the virtual and the physical data sets show some differences. While there is an (equally) strong correlation between each sensor and the damage position for the virtual data set, there is only a medium correlation between sensor A-0, and a low correlation between sensors A-1 and A-2 for the physical data set. In contrast, there is a low correlation between the load and all sensors for the virtual data set, but an increasing correlation from sensor A-0 to sensor A-2 for the physical data set. Regarding the damage width, there is no correlation recognized for all features.

In the next step, histograms of all features and labels are made and analyzed. These histograms are again created with the Seaborn Python library. Inside the physical data set both sensor rows are recorded during the experiments. For the (physical) sensor pair A-0 and B-0, the results are equal, while for the sensor pair A-1 and B-1 as well as A-2 and B-2 there are some slight deviations. This could be just because of the threshold of the histograms since the general characteristic is similar. In comparison to the virtual data set, there are two outcomes. The first one is, that the distributions for all three virtual sensors are much more equal than within the physical data set. The second is the distribution itself, which seems to be a beta distribution. For the damage position, the distributions of the virtual and physical data sets are contrary, since for the virtual data set data points are missing near the clamp. Regarding the load and the width, the distributions are even and equal among both data sets.

Finally, the regression plots from Seaborn allow insides into the characteristic between all features and a single label. As observed already in the heat maps, there is a trivial outcome regarding the damage width, which results in a linear behavior with nearly horizontal correlations for both data sets. The load seems just to have a scaling effect, which underlines the linearity of the material. Regarding the damage position, a complex characteristic appears that is strongly different among both data sets. For the virtual data set, there is a direction of high negative strains for damages that are near the clamping and low negative strains near the free end of the cantilever. The distributions are high for the (virtual) sensor A-0 and decrease towards the sensor A-2. Within the physical data set a partly similar behavior is recognised. Additionally, there are also low negative values for damage positions near the clamping. The characteristic is changing further when comparing the strains from A-0 to A-1 and A-2. In general, the strains from the virtual data sets have more linearity than the strains from the physical data set.

6. Train on synthetic - test on real

Table 5 – Classification - Existence of a damage: Accuracy, precision, recall, and f1 scores for the RandomForestClassifier (RFC), KNeighborsClassifier (KNC), LogisticRegression (LR), and RidgeClassifier (RC) from the virtual data set (VDS), physical data set (PDS), and the mixed data set (MDS).

		Accuracy	Precision	Recall	F1 score
VDS	RFC	0.9989	0.9989	1.0000	0.9995
	KNC	0.9989	0.9989	1.0000	0.9995
	LR	0.9989	0.9989	1.0000	0.9995
	RC	1.0000	1.0000	1.0000	1.0000
PDS	RFC	0.9714	0.9714	1.0000	0.9855
	KNC	0.9714	0.9714	1.0000	0.9855
	LR	1.0000	1.0000	1.0000	1.0000
	RC	0.9714	0.9714	1.0000	0.9855
MDS	RFC	0.9996	0.9996	1.0000	0.9998
	KNC	0.9987	0.9987	1.0000	0.9994
	LR	1.0000	1.0000	1.0000	1.0000
	RC	0.9996	0.9996	1.0000	0.9998

This section covers the investigation of the train on synthetic - test on real approach. It splits into the three sub-investigations of classification, regression, and multi-output regression. The classification is the starting point to see if a damage is present or not. Next, a regression model predicts the damage position. In the last stage, a multi-output regression model predicts the damage position and damage width. For these tasks, different models are compared for the corresponding disciplines. All models work with classical supervised machine learning methods. The investigation does not cover neural networks. For all sub-investigations, the corresponding models are trained in their corresponding domain first. This requires data set splitting of $80\,\%$ of training data and $20\,\%$ of testing data for the physical data set and the virtual data set. Common recommendations reach from $20\,\%$ to $30\,\%$ testing data set size [46]. Afterwards, the virtual data set spans a training data set, while the physical data set acts as a testing data set. In the transfer phase data normalization is necessary to eliminate deviations between the virtual and the physical data set. Since positive as well as negative strain values are present for the feature A-0, A-1, and A-2, the normalization reaches from -1 to 1.

6.1 Classification: Damage existence

The classification task is about identifying whether the structure has a damage or not. Compared to the regression and the multi-output regression, this is assumed as a trivial task. Nevertheless, it is an important first step in a possible process chain for structural health monitoring. The models under investigation are the RandomForestClassifier, KNeighborsClassifier, LogisticRegression, and RidgeClassifier. All models work with standard settings, except the KNeighborsClassifier. For the KNeighborsClassifier an ideal amount of 5 neighbors results from an evaluation of settings between $2 \leq$ neighbors ≤ 20. The evaluation metrics accuracy, precision, recall, and F1 score are compared to the different machine learning models. In turn, these different models are again compared to the machine learning model with the virtual data set, the physical data set, and the mixed data set.

Table 5 shows the results for the evaluation of the classification task. In general, the tests show overall good results. The virtual data set shows slightly better results for all models, compared to the physical data set. A reason for this could be the data set size of the virtual data set, which is more than 25 times bigger than the physical data set. While the virtual data set performs best with the RidgeClassifier, the physical data set shows equal results for the LogisticRegression, which is primarily not a classical classifier. Regarding the mixed data set, there are two important outcomes. First, all models perform equal or even better compared to the virtual and the physical data set. Second, the best performance is present for the LogisticRegression model. This draws the thesis that the the physical data set rather than the virtual data set is pioneering for the outcome of the mixed data set. A further result is that the recall of all models and all data sets is untrustworthy high. This is still comprehensible when comparing the only three undamaged cases with the much higher amount of damaged cases. Furthermore, this is a positive characteristic that provides high safety for the application. Nevertheless, instability is recognized, which is not visible in the results is presented in Table 5. Since the data sets contain only three cases of different loads, there are also only three undamaged cases for one of the two classes of the training data set. When splitting this data set there is a slight chance of filtering out the small class completely, which leads to an error. This effect should be covered automatically by the scikit-learn split function, but there seems to be a bug that data scientists should be aware of.

6.2 Regression: Damage position

Table 6 – Regression - Position of a damage: R2 score (R2), mean squared error (MSE), and mean absolute error (MAE) for the LinearRegression (LR), ElasticNet (EN), RandomForestRegressor (RFR), PolynomialFeatures (PF) from the virtual data set (VDS), physical data set (PDS), and the mixed data set (MDS).

		R2	MSE	MAE
VDS	LR	0.8888	1785.2633	31.9577
	EN	0.4583	8698.9515	80.2166
	RF	0.9590	659.1294	17.1609
	PF	0.9798	299.7171	12.4878
PDS	LR	0.9466	815.3106	24.8321
	EN	0.6677	5074.5676	54.2516
	RF	0.9675	497.0161	17.0822
	PF	0.9801	306.7420	13.5923
MDS	LR	-5.0991	94986.9032	261.4205
	EN	-1.2370	34839.0509	155.6380
	RF	-3.9311	76797.2057	244.8166
	PF	-5.0991	94986.9032	261.4205

The single regression task is about identifying the position of the damage on the cantilever along the x axis. The models under investigation are the LinearRegression, ElasticNet, RandomForestRegressor, and PolynomialFeatures. All models work with standard settings, except the PolynomialFeatures. For the PolynomialFeatures an ideal degree of 2 results from an evaluation of settings between $1 \leq$ neighbors ≤ 30. The baseline for this is the mixed data set since this is the overall goal. For both other data sets different degrees perform better. The evaluation metrics R2 score, mean squared error, and mean absolute error are compared to the different machine learning models. In turn, these different models are again compared to the machine learning model with the virtual data set, the physical data set, and the mixed data set.

Table 6 displaces the results for the evaluation of the regression task. There is a general ranking under the models that is equivalent to the virtual and physical data set. The ElasticNet model performs worst, while the PolynomialFeatures model shows the best results. For both data sets the model has an acceptable R2 score. Still, the mean absolute error is not satisfying for an accurate prediction. Compared to the overall length of the cantilever this results in a relative accuracy of $2\,\%$. In other words, this would equal a classification model with only 44 classes. Also, it is not promising to increase the data set size to decrease the mean absolute error. The reason for this is, that the physical data set performs equally with a much smaller data set size. Nevertheless, there is still room for improvement in both data sets. First, the degree of the PolynomialFeatures was fitted to the mixed data set model, but not the virtual and physical data set. Second, the data set could be increased with damage positions near or under a sensor location. This would be an option for the virtual data set only, since generating data in the physical domain for these positions would harm the sensors. Another relevant result is the performance of the physical data set, since the results are overall better, except for the mean squared error and the mean absolute error of the PolynomialFeatures model. For the mixed data set the investiga-

Table 7 – Multi-output regression - Position and width of a damage: R2 score (R2), mean squared error (MSE), and mean absolute error (MAE) of the damage position and the damage width for the LinearRegression (LR), RandomForestRegressor (RFR), ExtraTreesRegressor (ETR), KNeighborsRegressor (KNR), MultiOutputRegressor (MOR), and RegressorChain (RC) from the virtual data set (VDS), physical data set (PDS), and the mixed data set (MDS).

		Label	R2	MSE	MAE
VDS	LR	Position	0.8901	1562.6833	31.0280
		Width	0.0758	0.9216	0.9120
	RFR	Position	0.9501	710.3228	18.1043
		Width	-0.6363	1.6316	1.1163
	ETR	Position	0.9537	658.1575	16.3872
		Width	-0.9639	1.9583	1.1512
	KNR	Position	0.9598	571.8235	15.8567
		Width	-0.0698	1.0668	0.9339
	MOR	Position	0.9456	773.0409	13.9146
		Width	-0.0429	1.0400	0.6716
	RC	Position	0.9456	773.0409	13.9146
		Width	-0.3538	1.3500	0.9001
PDS	LR	Position	0.9791	365.2563	13.9495
		Width	-0.0261	1.0226	0.9047
	RFR	Position	0.9737	460.2150	16.2195
		Width	-0.0757	1.0720	0.9506
	ETR	Position	0.9923	134.2369	10.1623
		Width	-0.2842	1.2797	1.0176
	KNR	Position	0.8915	1897.1567	33.4951
		Width	-0.3883	1.3835	1.1059
	MOR	Position	-0.0026	17538.1056	109.0731
		Width	0.0220	0.9746	0.9847
	RC	Position	-0.0026	17538.1056	109.0731
		Width	-0.0099	1.0064	1.0024
MDS	LR	Position	-5.0991	94986.9032	261.4205
		Width	-0.4983	1.4962	1.0432
	RFR	Position	-3.9437	76993.5066	244.3924
		Width	-0.0988	1.0973	1.0096
	ETR	Position	-3.7477	73940.2559	238.7111
		Width	-0.1233	1.1218	1.0024
	KNR	Position	-3.7936	74654.8991	242.4634
		Width	-0.0895	1.0880	1.0078
	MOR	Position	-4.0280	78305.4432	238.6889
		Width	-7.0134	8.0023	2.2701
	RC	Position	-4.0280	78305.4432	238.6889
		Width	-0.5413	1.5391	1.0676

tion failed. The failure is indicated by the R2 score, which is negative for every model. This means that the model does not follow the trend of the data [47]. Furthermore, the mean squared errors are way too high. An analysis of the virtual and physical data shows that the characteristics diverge. This can be seen when comparing the strain over the damage positions. From these results, two main outcomes are derived.

First, further investigation of the test scenario is mandatory to exclude failures for the physical or virtual model. Regarding the virtual model, there is the possibility of natural non-linearity that is not covered by the linear solver and thus leads to a wrong virtual data set. Moreover, the sharp edges of the artificial damage can raise additional non-linearity the solver can not handle. Last but not least, the missing virtual data near the clamp possibly underlines the problems (see section 5.). For the physical model, there are even more possibilities that lead to defects or inaccuracies in the data set. E.g. these are the strain gauges, the incremental measurement system, and the artificial damage. The strain gauges can have issues with the application, wiring as well as deviations in location and rotation. Most likely, there are problems with artificial damage. It possibly compresses or bends under the load of the cantilever. This also includes vertical or horizontal displacements resulting from the linear guides. Furthermore, the adjustment of the exact height including some prestressing force is challenging. Errors from this stage can result in deviations in the expected bending line of the cantilever. Moreover, there are slight differences between the physical and the virtual model, like applied cables, sensors, protections, and fixations on the physical cantilever. These are not covered in the virtual model.

Second, there is still a chance to improve the predictions towards a stable train on synthetic - test on real approach. On the one hand, this is the main reason where domain adaptation comes in. Within the domain adaptation discipline, several methods can be tested to improve the predictions. This can also include the development of a model that predicts the correlation between both data sets. On the other hand, there is still the possibility to use neural networks that bring additional possibilities for solving this problem.

6.3 Multi-Output Regression: Damage position and damage width

The multi-output regression task is about identifying the position and the width of the damage at the same time. The models under investigation are the LinearRegression, RandomForestRegressor, ExtraTreesRegressor, KNeighborsRegresso, MultiOutputRegressor, and MultiOutputRegressor. All models work with standard settings, except the MultiOutputRegressor and MultiOutputRegressor. Here the support vector regression is used with a radial basis function kernel, a kernel coefficient of 0.1, a regularization parameter of 100, and an epsilon(-tube) of 0.1. These are the baseline settings for all three data sets. For this, there has been no hyperparameter study, yet. The evaluation metrics R2 score, mean squared error, and mean absolute error are compared to the different machine learning models. In addition, the evaluation metrics further distinguish between both labels. In turn, these different models are again compared to the machine learning model with the virtual data set, the physical data set, and the mixed data set.

Table 7 shows the results for the evaluation of the multi-output regression task. In general, there is a high variance between the different machine learning models within the different data sets. Furthermore, the variance between the different evaluation metrics is partly high. When comparing the virtual and the physical data set only one model guarantees equality, namely the LinearRegression model. While the model has a slightly better R2 score for the virtual data set, the mean squared error and the mean absolute error are much smaller for the physical data set. This is unusual, due to the smaller data set size. Possible problems tend not to be on the physical data set but on the virtual data set. An indicator for this is the mean squared error and the mean absolute error, which are unreasonably high in comparison to the other models of the virtual data set. With a focus on the R2 scores, there seems to be a strongly contrary behavior for the labels. While the quality of the R2 score increases for the position, it decreases for the width. This characteristic is more present for the virtual than for the physical data set. There is no clear answer to the question of the best machine learning model for the virtual data set since each model has different advantages and disadvantages. If one needs to choose a specific machine learning model the recommendation would be the MultiOutputRegresso model, because the mean absolute error for the position and width are the lowest here. Nevertheless, this model does not show the best results for the mean squared error and especially the R2 score. Similar conclusions can be drawn for the physical data set but for different machine learning models. The ExtraTreesRegressor performs best regarding the position of the damage when looking at the mean absolute error. In contrast, it does not the best prediction regarding the width of the damage. However, this is less decisive since the variance of the widths is much smaller than the variance of the positions. Conspicuous are the models MultiOutputRegressor and RegressorChain. While the results for the damage width are equal among the different models and data sets, the damage position fails in all three evaluation metrics. A curious result is, that the mean absolute error of the damage position is partly lower compared to the regression task from subsection 6.2. Regarding the mixed data set the models show equal behavior like for the regression task (see subsection 6.2). Thus, the resulting recommendations for action are the same as for the single regression task, too. Finally, it is to point out that the multi-output regression seems to have additional instabilities among all data sets.

7. Conclusion and outlook

The present paper introduces a methodology for the development of a train on synthetic - test on real application. This methodology is both, a structure for this specific project, but also a valid method for equal applications. Train on synthetic - test on real is a sub-category of transfer learning or more precisely transductive transfer learning. It is also known as sim to real.

The approach is used for the strain-based damage detection of an aircraft wing. For this application example a simplified test scenario is developed, to eliminate material and geometrical complexity. This test scenario considers a bending beam with the possibility of applying an anomaly to the can-

tilever. As an anomaly an artificial reversible crack-like damage is introduced, which is a support trestle with a rectangular contact surface.
Following the test scenario splits into a physical and a virtual model. The physical model covers a test stand with a clamped cantilever with applied strain gauges, and the artificial damage with an incremental measurement. The virtual model consists of structural finite element model simulations. Both models generate a corresponding data set. This data set includes the applied load on the cantilever and the sensor values from the strain gauges as features as well as the damage position and damage width as labels.
Besides a data pre-analysis, the data sets are used to train and test different machine learning models. For this, the training takes place in three different stages, each covering a different discipline. First, the existence of a damage is predicted with classification models. Second, the damage position is predicted with regression models. Finally, the damage position and the damage width are predicted with multi-output regression models. All these models are pre-trained with the physical and virtual data set inside the corresponding domain. Afterwards, the virtual data set acts as a training data set, while the physical data set acts as a testing data set, which results in a mixed data set.
The results for the classification task show a good performance, despite the small amount of undamaged cases in both data sets. Nevertheless, caution is advised, since some of the models run into an error, because of a data splitting that excludes the weak represented class. Two remarkable outcomes of this investigation are first, the high overall recall and second, the performance increase of the mixed data set compared to each of the virtual and the physical data set.
For the regression task, the virtual and the physical data sets show acceptable results. Regardless of the much smaller physical data set size, the results are comparable for two of the four investigated models. For the two other models, there are deviations in the behavior of the different data sets. The model with the mixed data set does not follow the data trend. Thus, the train on synthetic - test on real approach does not show success for the prediction of the damage position.
Similar behavior is recognized for the multi-output regression. The models perform partly even better in comparison to the (single) regression models when analyzing the mean absolute error of the virtual and physical data set. As for the regression models, the prediction of the damage position and the damage width show no success for the mixed data set, too.
There are multiple possible reasons for the unfulfilled train on synthetic - test on real approach for the given test scenario. During the data pre-analysis deviations between the virtual and physical data set has been recognised. These deviations are the most relevant reason for the instability of the machine learning models. The physical data set consists of some discontinuities for the comparison of the strain and the crack position. Yet, it is not clear whether the problem exists in the virtual or physical domain.

Nevertheless, this outcome offers a chance for future work from different perspectives.
First, identifying the inaccuracy of either the virtual or the physical model. This includes further investigations on both domains. Regarding the virtual model, a switch to a non-linear solver is an option to calculate a possible non-linear bending line. In addition, this could solve the missing data near the clamping, which itself could improve the performance of the models. For the physical model, unrecognized inaccuracy can arise from the different measurements. Also, it could be that there is a lack of accuracy for the artificial damage between both domains.
Second, there are still many different options to increase the performance of the prediction in the field of machine learning. Since this was an initial investigation, no non-trivial domain adaptation methods have been applied, yet. Furthermore, neural networks can be an option. These have extended possibilities to find more complex correlations between the virtual and the physical data set.
Finally, as a result of the literature review the field of train on synthetic - test on real is a rising scientific area with the potential for further research. This is especially the case for applications examples with a background in structural health monitoring. This covers systematical literature reviews and further experiments with other test scenarios.

Acknowledgements

The authors would like to acknowledge the contributions of their colleagues within the Institute of Maintenance, Repair and Overhaul of the German Aerospace Center (DLR). The contributions cover discussions on machine learning, structural mechanics, software development, and measurement system.

Contact Author Email Address

Philipp Conen M.Sc.
philipp.conen@dlr.de | +49 40 2489641-134
Research Associate at the German Aerospace Center (DLR) in the Institute of Maintenance, Repair and Overhaul specializing in strain-based structural health monitoring, condition monitoring, machine learning, domain adaptation, and simulation as well as in UAS MRO platforms.

Copyright Statement

References

[1] United Nations. The paris agreement. https://unfccc.int/process-and-meetings/the-paris-agreement. Accessed: 2023-12-21.

[2] Anna Christmann and Oliver Luksic. Klimaneutrale Luftfahrt - Gemeinsames Papier der Bundesregierung. Technical Report DOE-SLC-6903-1, German Federal Government, Berlin, Germany, June 2022.

[3] European Union Aviation Safety Agency (EASA). European aviation environmental report. https://www.easa.europa.eu/eco/eaer/topics/introduction/industry-goals. Accessed: 2023-12-21.

[4] Air Transport Action Group. Sustainable development goals and aviation. https://aviationbenefits.org/un-sustainable-development-goals/sustainable-development-goals-and-aviation/. Accessed: 2023-12-21.

[5] United Nations Department of Economic and Social Affairs. Sustainable development goals. https://sdgs.un.org/goals. Accessed: 2023-12-21.

[6] Frederico Afonso, Martin Sohst, Carlos M.A. Diogo, Simão S. Rodrigues, Ana Ferreira, Inês Ribeiro, Ricardo Marques, Francisco F.C. Rego, Abdolrasoul Sohouli, Joana Portugal-Pereira, Hugo Policarpo, Bruno Soares, Bruna Ferreira, Edgar C. Fernandes, Fernando Lau, and Afzal Suleman. Strategies towards a more sustainable aviation: A systematic review. *Progress in Aerospace Sciences*, 137:100878, 2023.

[7] Mikhail A. Ershov, Vsevolod D. Savelenko, Nikita O. Burov, Uliana A. Makhova, Daria Y. Mukhina, David R. Aleksanyan, Vladimir M. Kapustin, Marina M. Lobashova, Alexander V. Sereda, Tamer M.M. Abdellatief, Ahmad Baroutaji, and Mohammad Ali Abdelkareem. An incorporating innovation and new interactive technology into obtaining sustainable aviation fuels. *Energy*, 280:128156, 2023.

[8] Talal Yusaf, Louis Fernandes, Abd Rahim Abu Talib, Yazan S. M. Altarazi, Waleed Alrefae, Kumaran Kadirgama, Devarajan Ramasamy, Aruna Jayasuriya, Gordon Brown, Rizalman Mamat, Hayder Al Dhahad, F. Benedict, and Mohamd Laimon. Sustainable aviation - hydrogen is the future. *Sustainability*, 14(1), 2022.

[9] Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younes Bennani. *Advances in domain adaptation theory*. Elsevier, 2019.

[10] Mohammad Navid Fekri, Ananda Mohon Ghosh, and Katarina Grolinger. Generating energy data for machine learning with recurrent generative adversarial networks. *Energies*, 13(1):130, 2019.

[11] Tariq Alkhalifah, Hanchen Wang, and Oleg Ovcharenko. Mlreal: Bridging the gap between training on synthetic data and real data applications in machine learning. *Artificial Intelligence in Geosciences*, 3:101–114, 2022.

[12] Wouter M Kouw and Marco Loog. An introduction to domain adaptation and transfer learning. *arXiv preprint arXiv:1812.11806*, 2018.

[13] Shiliang Sun, Honglei Shi, and Yuanbin Wu. A survey of multi-source domain adaptation. *Information Fusion*, 24:84–92, 2015.

[14] Xingxing Weng, Yuchun Huang, Yanan Li, He Yang, and Shaohuai Yu. Unsupervised domain adaptation for crack detection. *Automation in Construction*, 153, 2023.

[15] Daniel Asefa Beyene, Dai Quoc Tran, Michael Bekele Maru, Taeheon Kim, Solmoi Park, and Seunghee Park. Unsupervised domain adaptation-based crack segmentation using transformer network. *Journal of Building Engineering*, 80, 2023.

[16] Junxian Shen, Di Song, Tianchi Ma, and Feiyun Xu. Blade crack detection based on domain adaptation and autoencoder of multidimensional vibro-acoustic feature fusion. *Structural Health Monitoring*, 22(5):3498 – 3513, 2023.

[17] Kangcheng Liu and Ben M. Chen. Industrial uav-based unsupervised domain adaptive crack recognitions: From database towards real-site infrastructural inspections. *IEEE Transactions on Industrial Electronics*, 70(9):9410 – 9420, 2023.

[18] Chunmian Lin, Daxin Tian, Xuting Duan, Jianshan Zhou, Dezong Zhao, and Dongpu Cao. Da-rdd: Toward domain adaptive road damage detection across different countries. *IEEE Transactions on Intelligent Transportation Systems*, 24(3):3091 – 3103, 2023.

[19] Ling Xiang, Xingyu Zhang, Yue Zhang, Aijun Hu, and Hankun Bing. A novel method for rotor fault diagnosis based on deep transfer learning with simulated samples. *Measurement: Journal of the International Measurement Confederation*, 207, 2023.

[20] Yanwen Wu, Mingjian Hong, Ao Li, Sheng Huang, Huijun Liu, and Yongxin Ge. Self-supervised adversarial learning for domain adaptation of pavement distress classification. *IEEE Transactions on Intelligent Transportation Systems*, page 1–12, 2023.

[21] Kangcheng Liu. Robust industrial uav/ugv-based unsupervised domain adaptive crack recognitions with depth and edge awareness: From system and database constructions to real-site inspections. page 5361 – 5370, 2022.

[22] Zhening Huang, Weiwei Chen, Abir Al-Tabbaa, and Ioannis Brilakis. Nha12d: A new pavement crack dataset and a comparison study of crack detection algorithms. page 285 – 292, 2022.

[23] Huijun Liu, Chunhua Yang, Ao Li, Sheng Huang, Xin Feng, Zhimin Ruan, and Yongxin Ge. Deep domain adaptation for pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems*, 24(2):1669 – 1681, 2023.

[24] Mitchell J. Hallee, Rebecca K. Napolitano, Wesley F. Reinhart, and Branko Glisic. Crack detection in images of masonry using cnns. *MDPI - Sensors*, 21(14), 2021.

[25] Mohamad Alipour and Devin K. Harris. Increasing the robustness of material-specific deep learning models for crack detection across different materials. *Engineering Structures*, 206, 2020.

[26] K. Prajna and C.K. Mukhopadhyay. Fractional fourier transform based adaptive filtering techniques for acoustic emission signal enhancement. *Journal of Nondestructive Evaluation*, 39(1), 2020.

[27] István A. Veres and Dieter M. Profunser. Time-resolved modal decomposition based on the linear prediction method - applied for crack detection. *Journal of the Acoustical Society of America*, 118(5):3021 – 3030, 2005.

[28] Li Wang, Guoqiang Liu, Chao Zhang, Yu Yang, and Jinhao Qiu. Fem simulation-based adversarial domain adaptation for fatigue crack detection using lamb wave. *MDPI - Sensors*, 23(4), 2023.

[29] Xiaodong Wang, Feng Liu, and Dongdong Zhao. Deep transfer fault diagnosis using digital twin and generativeadversarial network. page 186 – 193, 2021.

[30] Jong Moon Ha and Olga Fink. Domain knowledge-informed synthetic fault sample generation with health data map for cross-domain planetary gearbox fault diagnosis. *Mechanical Systems and Signal Processing*, 202, 2023.

[31] Mohammadreza Ghorvei, Mohammadreza Kavianpour, Mohammad TH Beheshti, and Amin Ramezani. Synthetic to real framework based on convolutional multi-head attention and hybrid domain alignment. 2022.

[32] Will Koehrsen. Neural network embeddings explained. *Towards Data Science*, 13, 2018.

[33] Julián Alberto Palladino, Diego Fernandez Slezak, and Enzo Ferrante. Unsupervised domain adaptation via cyclegan for white matter hyperintensity segmentation in multicenter mr images. In *16th International Symposium on Medical Information Processing and Analysis*, volume 11583, page 1158302. SPIE, 2020.

[34] Cédric Villani et al. *Optimal transport: old and new*. Springer Nature, 2009.

[35] Gemmel Metalle. Legierungsbeschreibung EN AW-6060. `https://www.gemmel-metalle.de/`

downloads/Legierungsbeschreibung_AlMgSi0,5_F22.pdf. Accessed: 2023-12-21.

[36] Sinadrives. Deflection calculation. https://sinadrives.com/en/deflection-calculation/, 2022. Accessed: 2024-06-15.

[37] Christian Spura. *Technische Mechanik 2. Elastostatik: Nach fest kommt ab*. Springer Fachmedien Wiesbaden, 2019.

[38] Rollon GmbH. Linear line catalogue. https://www.rollon.com/deu/de/familie/linearfuehrungen/profilschienenfuehrungen-mit-kugelumlaufsystem-mono-rail/. Accessed: 2023-12-21.

[39] HBM Test and Measurement. *Dehnungsmessstreifen*. Hottinger Baldwin Messtechnik GmbH, Darmstadt, Germany, 2023.

[40] Hottinger Baldwin Messtechnik GmbH. Quantumx mx1615b - data sheet. https://www.hbm.com/fileadmin/mediapool/hbmdoc/technical/B03899.pdf. Accessed: 2023-12-21.

[41] ELGO. Technische informationen: Elgo magnetbänder - inkrementelle magnetbänder für lineare messsysteme. https://www.elgo.de/fileadmin/user_upload/pdf/manual/accessory/MB20-XX-10-1-R-X-MA-D.pdf, 2013. Accessed: 2024-06-15.

[42] ELGO. Betriebsanleitung: Serie lmix2 - magnetisches inkremental-längenmesssystem mit 0,025 mm auflösung. https://www.elgo.de/fileadmin/user_upload/pdf/manual/sensors/LMIX2-000-MA-D.pdf, 2018. Accessed: 2024-06-15.

[43] ELGO. Betriebsanleitung: Serie z50-000 - 1-achs positionsanzeige 24 vdc. https://www.elgo.de/fileadmin/user_upload/pdf/manual/indicators/Z50-000-MA-D.pdf, 201. Accessed: 2024-06-15.

[44] Hottinger Baldwin Messtechnik GmbH. Configurable pre-wired strain gauges. https://b2bstore.hbm.com/. Accessed: 2023-12-21.

[45] Peter Steinke. *Finite-Elemente-Methode: Rechnergestützte Einführung*. Springer Vieweg, 2015.

[46] Encord. Training, validation, test split for machine learning datasets. https://encord.com/blog/train-val-test-split/, 2023. Accessed: 2024-06-15.

[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.