On the Transfer of Damage Detectors Between Structures: An Experimental Case Study

L.A. Bull[a,*], P.A. Gardner[a], N. Dervilis[a], E. Papatheou[b], M. Haywood-Alexander[a], R.S. Mills[a], K. Worden[a]

[a] *Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK*
[b] *College of Engineering Mathematics and Physical Sciences, University of Exeter, Exeter EX4 4QF, UK*

Abstract

Incomplete data – which fail to represent environmental effects or damage – are a significant challenge for structural health monitoring (SHM). Population-based frameworks offer one solution by considering that information might be shared, in some sense, between similar structures. In this work, the data from a group of aircraft tailplanes are considered collectively, in a shared (more consistent) latent space. As a result, the measurements from one tailplane enable damage detection in another, utilising various pair-wise comparisons within the population.

Specifically, Transfer Component Analysis (TCA) is applied to match the normal condition data from different population members. The resulting nonlinear projection leads to a general representation for the normal condition across the population, which informs damage detection via measures of discordancy. The method is applied to a experimental dataset, based on vibration-based laser vibrometer measurements from three tailplanes. By considering the partial datasets together, consistent damage-sensitive features can be defined, leading to an 87% increase in the true positive rate, compared to conventional SHM.

Keywords: Population-based structural health monitoring; domain adaptation; transfer learning; novelty detection; one-class classification; damage detection

1. A Population-based Approach to SHM

Data-driven frameworks are critical within Structural Health Monitoring (SHM) [1], as well as related fields, including: condition monitoring [2], non-destructive testing [3, 4], and prognosis and health management [5]. In general terms, machine learning algorithms are used to detect patterns in previously-recorded data, to infer the condition of the system (operational, environmental, or health) given measurements into the future. Health and/or performance predictions are valuable, as they improve maintenance, decision making and efficiency [6].

The development of machine learning methods for health monitoring of engineering systems is prevalent in the literature [7, 8]; however, data-models typically require large amounts of *training data* to learn a variety of patterns, thus enabling good generalisation to future measurements. In SHM, however, the data available for training are often highly restricted and unrepresentative of future observations [9]. Consider a rail-network, for example: acquisition systems might record several years of data from operational lines; on

*Corresponding author
Email address: l.a.bull@sheffield.ac.uk (L.A. Bull)

the other hand, a newly-built line will have limited data. In this case, the absence of *a priori* measurements inhibits the use of conventional (supervised) data modelling for monitoring purposes.

One solution to incomplete training data adopts a Population-Based approach to SHM (PBSHM) [10–12]. By considering groups of *similar* systems (or models) together, information might be shared or transferred, in some sense, between members of that population. Returning to the rail-network example: measurements from one (or several) line(s) might prove useful to inform the data-modelling of another. The assumption here is that patterns in one dataset resemble patterns in another. Therefore, a critical step for PBSHM is establishing which systems are similar enough to share information – some similarity measures have been proposed in [11]. By transferring information between similar systems, the predictive performance of pattern recognition algorithms should significantly improve. Caution is required, however: if dissimilar systems are used to inform each other, the predictive performance can be degraded, leading to *negative transfer* [13].

In this work, the concept of knowledge transfer between groups of similar systems is demonstrated for SHM via an experimental case study. The data available for training are extended by considering the measurements from a population of structures collectively, in a shared, more consistent, latent space. The experiments show that, by considering the *partial* datasets together, the reliability of damage-detection across the population can increase.

The layout of the paper is as follows. Section 2 summarises transfer learning via domain adaptation, in the context of damage-detection, while Section 3 covers related work. Section 4 explains the aircraft tailplane experiments, as well as benchmark techniques, to motivate the need for transfer learning. Section 5 explains the TCA algorithm, and how it can be applied to inform damage detection. Section 6 presents the results and discussion, while Section 7 offers concluding remarks and future considerations.

2. PBSHM and Transfer: Domain Adaptation

Machine learning methods typically assume that the data between training and testing are drawn from the same feature space and underlying data distribution [13, 14]. For SHM, this implies no change in modelled distributions from inference to the monitoring of real-time signals. In practice, the assumption of stationarity [15, 16] can break down for various reasons: e.g. cold temperature effects can lead to a shift in the distribution for the normal condition data [17]; alternatively, one might look to use the data from one system to inform the model of another – as previously discussed.

Once the underlying distribution has changed, most statistical models must be completely re-learnt to ensure good generalisation [13]. Alternatively, *transfer learning* can accommodate for changes in the data distribution, task, or feature space.

At this stage, it is useful to define *domain* and *task* objects [18]:

- A **Domain** $\mathcal{D} = \{\mathcal{X}, p(\mathbf{X})\}$ is an object that consists of a feature space $\mathcal{X}$ and a marginal probability distribution $p(\mathbf{X})$ over a finite sample of feature data $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n} \in \mathcal{X}$.

- A **Task** $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$ is a combination of a label space $\mathcal{Y}$ and a predictive model/function $f(\cdot)$. The function $f : \mathcal{X} \to \mathcal{Y}$ is unknown and inferred from the training data, while the finite set of corresponding labels is $\mathbf{y} = \{y_i\}_{i=1}^{n} \in \mathcal{Y}$.

Considering vibration-based monitoring, frequency-domain observations ($\mathbf{x}_i$) from rotating machinery could represent the feature space $\{\mathbf{x}_i\}_{i=1}^{n} \in \mathcal{X}$, while the model $f(\cdot)$ might predict the condition of the

machine bearings – classified by labelling $y_i \in \mathcal{Y}$; for example: normal ($y_i = 0$), damaged ($y_i = 1$), or environmental effects ($y_i = 2$).

This work considers a transfer problem between two domains at a time, which is typical in the literature [13]. Firstly, the *source domain* $\mathcal{D}_s$ is a finite set of observations and corresponding labels, i.e. $\{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ where $\{\mathbf{x}_i^s \in \mathcal{X}_s,\ y_i^s \in \mathcal{Y}_s\}$. The second is referred to as the *target domain* $\mathcal{D}_t$ with observations and (potentially unknown) labels, i.e. $\left\{\left(\mathbf{x}_i^t, y_i^t\right)\right\}_{i=1}^{n_t}$ where $\{\mathbf{x}_i^t \in \mathcal{X}_t,\ y_i^t \in \mathcal{Y}_t\}$.

Following Pan and Yang [13], transfer learning can be defined generally,

> *Given a source domain $\mathcal{D}_s$ and task $\mathcal{T}_s$, as well as a target domain $\mathcal{D}_t$ and task $\mathcal{T}_t$, transfer learning aims to improve the target predictive function $f_t(\cdot)$ in $\mathcal{D}_t$, using the knowledge in $\mathcal{D}_s$ and $\mathcal{T}_s$, where $\mathcal{D}_s \neq \mathcal{D}_t$, or $\mathcal{T}_s \neq \mathcal{T}_t$.*

For the case study presented here, the source and target domains are associated with different structures, while the (common) task is to represent/approximate the normal condition data to improve damage detection. In other words, the task is a type of *one-class classification* [19], where $y_i = 0$ (normal) $\forall i$ in training. The goal is, therefore, to *match* the normal condition distributions between source and target systems, to improve the predictive performance (primarily for the target structure) though a *shared* model f.

These ideas align closely with those of *transductive* transfer learning [20]. Transductive transfer considers that the source and target tasks are the same ($\mathcal{T}_s = \mathcal{T}_t$) while the domains are different ($\mathcal{D}_s \neq \mathcal{D}_t$). Relating back to the experiments in this work, the model of the normal-condition data is the common task f, while the features and distributions from each structure represent source/target domains.

Specifically, domain adaptation is applied as a form of transductive transfer learning [20]; this is appropriate when a model $f(\cdot)$ will not generalise between the source and target domains due to differences between the distributions $p(\mathbf{x}_i^s)$ and $p(\mathbf{x}_i^t)$ [18]. Transfer Component Analysis (TCA) [14] is applied, which looks to *match* source and target distributions in a shared latent space ($\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^{n_s+n_t} \in \mathcal{Z}$) where the data properties can be preserved. In the latent space $\mathcal{Z}$, a shared model ($f = f_s = f_t$) can be learnt for novelty/damage detection. The population-based framework to match normal condition data between structures is visualised in Figure 1. Algorithm details are provided in Section 5.

3. Related Work

A 'shared' model to represent a population of systems is closely related to the concept of the *form* for PBHSM [10]. The *form* is typically represented by a model of some consistent feature recorded from members within a population. The model should capture the essential nature as well as variance of systems within the group. Future measurements can then be monitored through comparisons to the *form*. In previous work [10], a (Gaussian process) regression of the power curve was used to represent the form for a wind turbine population. The framework in [10] does not attempt to match the data distributions (or tasks) between members through transfer learning; instead, the population data are considered collectively in the original data-space, and the tasks are modelled separately.

The majority of literature concerning transfer learning for SHM focusses on image classification through Convolutional Neural Networks (CNNs), e.g. [21–24]. Typically, applications consider crack detection [21–23], and *fine tuning* the parameters of a CNN trained on the source domain to aid generalisation in the target domain. Generally, fine tuning does not aim to transfer label knowledge from source to target domains; instead, it focusses on re-purposing expensive-to-train 'deep' neural networks [18].

In addition to fine tuning, neural networks have been applied for domain adaptation within SHM, for example [25–27]. These techniques use a neural network to define the mapping from the domain-data into

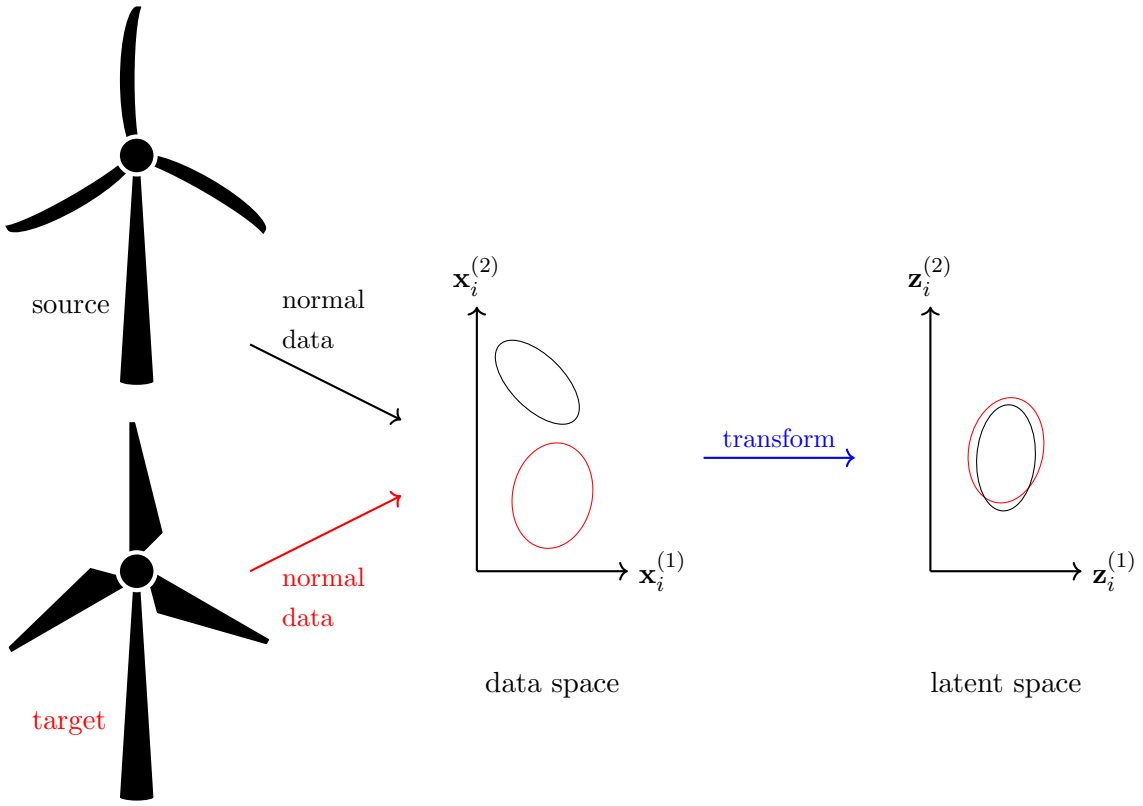


Figure 1: Visualisation of domain adaptation for knowledge transfer between source and target structures, distribution matching for the normal condition data. Ellipses represent clusters of data from the source (black) and target (red) structures. In the original space $\mathbf{x}_i \in \mathcal{X}$, the distributions for each structure (i.e. domain) are different; in the latent space $\mathbf{z}_i \in \mathcal{Z}$ (a transformation of $\mathbf{x}_i$) the distributions have been (approximately) matched.

a shared latent space, where a classification model is learnt. An interesting study is presented by Michau and Fink [28], demonstrating domain adaptation for condition monitoring, applied to a fleet of power plants. Unsupervised feature alignment is achieved using a variational auto-encoder; a new loss function is introduced to preserve the inter-point relationship between the inputs and latent features. Feature alignment is further enforced with an origin discriminator, trained in an adversarial manner [28]. The dataset for this application is large (tens of thousands of measurements), allowing for the inference of a relatively highly-parametrised network.

Besides neural networks, a paper by Chakraborty *et al.* [29] applies a probabilistic method for transfer learning, by defining an objective function such that information in the source and target domains is considered jointly. This application focusses on the important issue of *sensor coverage* (for a single structure) rather than transferring information between similar systems.

Domain adaptation has been discussed generally for SHM by Gardner *et al.* [18, 30]; this work considers methods for knowledge transfer, comparing tools to aid *supervised* damage classification. Case studies include a simulated source and target domain, as well as a simulated source and experimental target structure. In other words, two structures are considered per case study, at least one of which corresponds to a simulated system.

3.1. Novel contribution

Considering a population-based approach to SHM, transfer learning is applied here to an experimental case study, to aid *unsupervised* damage detection. The SHM data concern laser vibrometer measurements

from a population of three aircraft tailplanes (i.e. three pairs of structures). To extend the concept of the population *form* [10], transfer learning is applied to infer a shared model, representing the underlying distribution of the normal condition data for similar systems – as visualised in Figure 1. The model is utilised for damage detection through comparisons to future data.

Rather than considering transfer from a single source to a target structure, a population of three tailplanes are considered together, with (pair-wise) transfer between each member. All members correspond to an experimental rig, rather than simulated systems. Unlike various neural networks, the proposed methods are suited to small/incomplete data, which are found in many practical examples of SHM.

4. Experimental Data: Aircraft Tailplane Population

Three tailplanes from a Piper PA-28 aircraft are considered in the experiments. Two planes (labelled A and B) correspond to 'Arrow' variants of the PA-28, while the third corresponds to a 'Cherokee' variant (labelled C). Following separation from the aircraft, the elevators and wing-tips were removed from each tailplane. All wing sections have the same airfoil (NACA0012) and chord (0.76m) with different spans: tail C is 3.62m long, while A and B are 2.75m long.

Each plane was cut in half, to create three pairs of similar wing structures. These are referred to as: A1/A2, B1/B2, and C1/C2 – for port/starboard sides (1/2) respectively. The A plane was cut symmetrically, while the B plane was cut *asymmetrically* – the port side (B1) has a longer section at the original connection, while the starboard side (B2) is shorter – dimensions are provided in Figure 2. Lastly, the C plane was cut symmetrically, with damage artificially introduced to the starboard wing (C2) by removing a rectangular section from the skin – again, dimensions are provided in the schematic in Figure 2. The structures are also shown in Figure 3, where the damage to C2 is visible. It should be considered that the artificial damage imitates a fairly large, significant fault. The mass difference between the A planes was 1.89% (4.85kg and 4.76kg for A1/A2); for the B planes it was 9.39% (4.26kg and 4.66kg for B1/B2); for the C planes it was 0.56% (6.74kg and 6.70kg for C1/C2)[1].

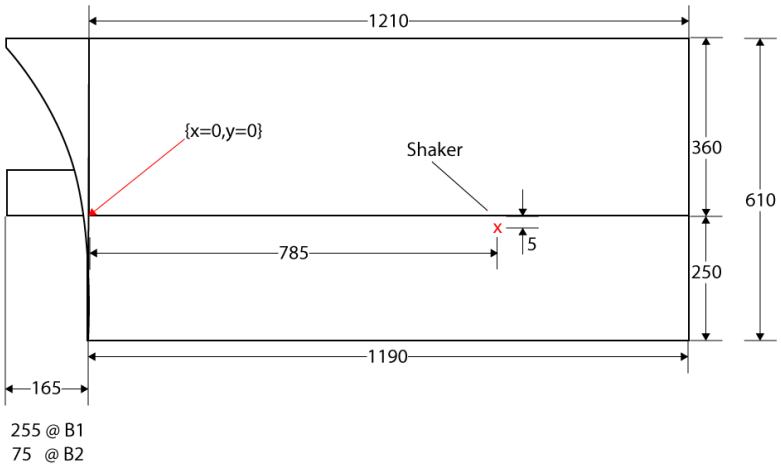
Following PBSHM terminology [10], the A and B tailplanes would be referred to as *weakly homogeneous*. In general terms, this implies the structures are nominally identical, with slight variations. That is, while each structure is the same model (Arrow), the port and starboard sides are mirrored; additionally, for B1/B2 the cut is asymmetric, leading to differences in mass and geometry[2]. When including the C planes, the group becomes *heterogeneous* [12]. While each structure is a tailplane wing, they correspond to two different models of aircraft (Cherokee/Arrow). Despite differences, however, the structures remain *similar*: the chord and aerofoil are the same design, and the A/B planes can be viewed as subsets of the C plane geometry (refer to Figures 2 and 3). Homogeneous and heterogeneous structures for PBSHM are more formally defined (through graphical representations) in [11].

4.1. Vibration tests

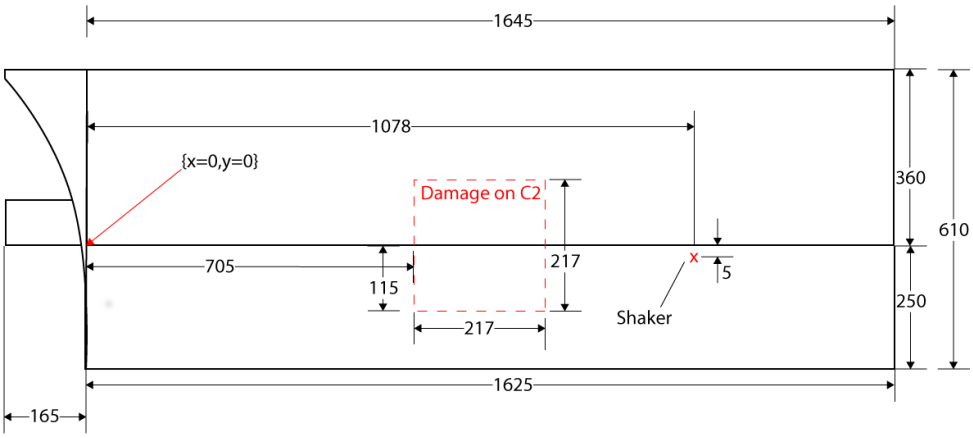
Each tailplane was excited with Gaussian white noise using an electromagnetic shaker. The tailplanes were hung free-free from a metal frame, and the shaker was attached to the rear, as demonstrated in Figure 4. The responses of the structures were recorded using a scanning laser Doppler vibrometer. Laser scanning vibrometry is beneficial, as it measures the response of a large surface in a single test, unaffected by the mass of sensing devices (measurements are contact free). The laser vibrometer was a tripod-mounted

[1] Percentages are w.r.t. the lightest wing.

[2] Manufacturing tolerances and imperfections (e.g. small dents or cracks) should also be considered.



(a) A/B planes, including asymmetric cut for B1/B2



(b) C plane, including damage in C2

Figure 2: Schematic, units are in mm. The reference point for scanning is $\{x = 0, y = 0\}$. The red cross indicates the forcing location, while the red dashed square shows the artificial damage in C2.

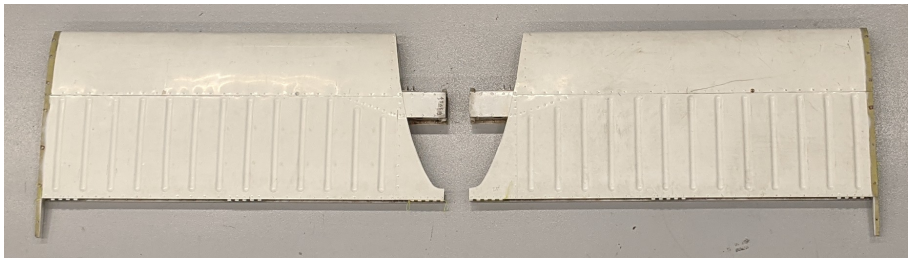
Polytec PSV-400 scanning head, with a PSV-A-420 geometry scan unit, controlled by a Polytec OFV-5000 controller and a JSV-500 junction box.

The frequency bandwidth during testing was 1kHz with a sample rate of 2.56kHz, leading to a resolution of 0.3125 Hz in the frequency domain (3200 spectral points). For tailplanes A and B, there were 180 scan points on the port and starboard wings (each). For the C plane, due to its larger span, there were 295 scanning points on the port and starboard wings (each). Scanning and forcing locations were consistent (as far as possible) between port and starboard pairs: the scanning grid for each pair is compared in Figure 5, while the forcing location is included in Figure 2. There are six averages per response point when storing frequency domain measurements – the Frequency Response Functions (FRFs, H1 estimator) and coherence functions are used in this work.

Each tailplane test (A, B, and C) is summarised below:

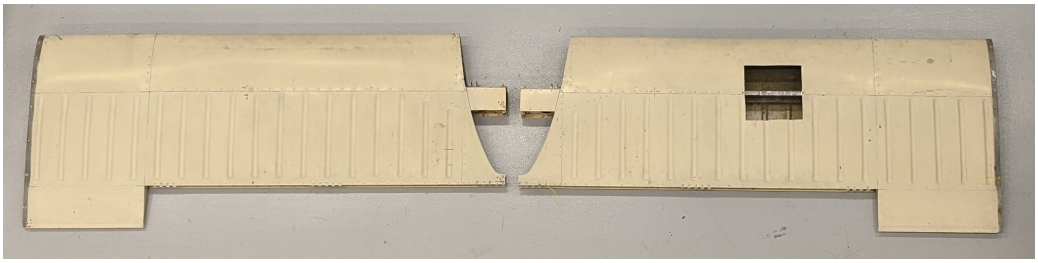
A plane PA-28 *Arrow*: symmetrically cut, 180 response points for port (A1) and starboard (A2) wings (each), 1.89% mass difference between port and starboard.

B plane PA-28 *Arrow*: asymmetrically cut, 180 response points for port (B1) and starboard (B2) wings (each), 9.39% mass difference between port and starboard.



(a) A1 (port)/A2 (starboard)

(b) B1 (port)/B2 (starboard)



(c) C1 (port)/C2 (starboard)

Figure 3: Tailplane structures, including artificial damage in wing C2 – dimensions are provided in Fig. 2.

C plane PA-28 *Cherokee*: symmetrically cut, 295 response points for port (C1) and starboard (C2) wings (each), 0.56% mass difference between port and starboard.

Figure 6 shows the summed FRF over the response locations for each tailplane. The response functions are self-normalised in the log-space through the Z-score (subtracting the sample mean and dividing by the standard deviation) using a subset of training data[3]. Clearly, the FRFs in Figure 6 differ significantly from structure to structure, even comparing port and starboard wings of the same plane. In agreement with the modal analysis by Papatheou *et al.* [31], however, some consistent patterns can be observed between port and starboard wings – as well as between A, B and C pairs. Intuitively, there are more similarities at lower frequencies of the response for A and B, while the discrepancy increases at higher frequencies. Even considering the C plane (including damage), a low-frequency mode for all structures appears at around 75Hz.

4.2. Problem set-up and feature selection

To define a knowledge transfer problem, each wing *pair* is associated with its own domain – A, B and C. In this way, the experiments are viewed as a three-domain problem: domains A and B contain normal

[3]Normalised features are better suited to Transfer Component Analysis, as well as the benchmark methods (Principal Component Analysis).

(a) test-rig and scanning laser vibrometer

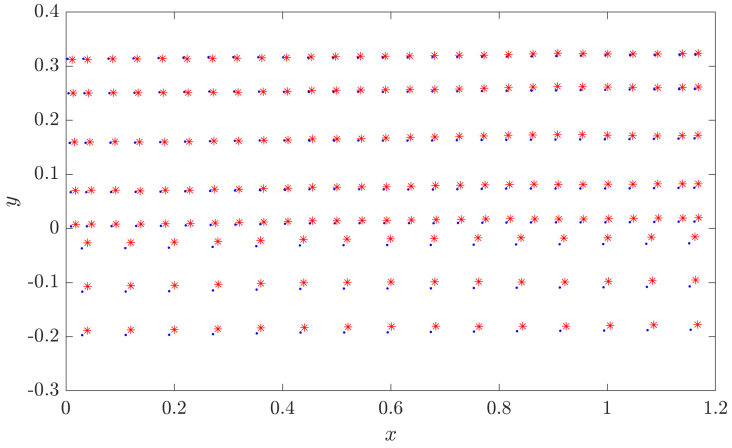
(b) electrodynamic shaker (fixed to rear)

Figure 4: Test set-up example – shown for PA-28 Arrow tailplane

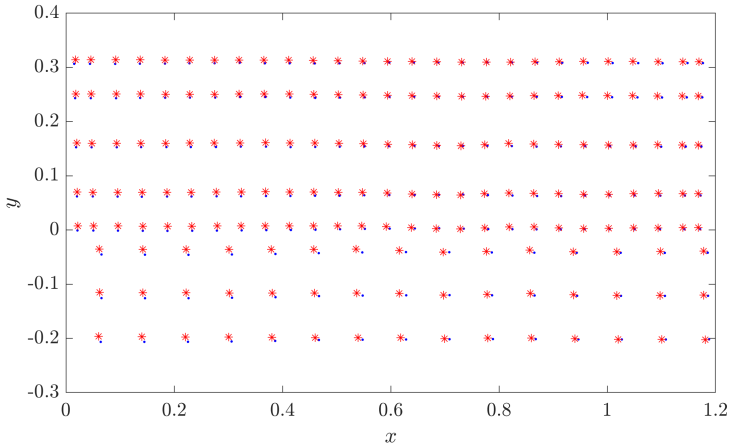
condition data, while domain C has both normal and damage data. It is assumed, therefore, that the *normal* data from port and starboard sides are sampled from the same underlying distribution.

Considering this approximation, it is necessary to select features that are (relatively) consistent between the port and starboard wings (of A and B) for transfer learning to make sense[4]. Based on engineering/physics-based knowledge of the FRF, Figure 6 is inspected visually, considering wings A and B only. According to the framework in [32], the window between 56.56 – 81.56Hz can be selected as a consistent feature, highlighted with dotted lines. This region of the response covers a common mode for the normal condition FRFs. The consistency of this feature is logical, as the response below 50Hz is increasingly influenced by rigid body dynamics, leading to discrepancies in the response for the B tailplanes; here, the mass imbalance (due to the asymmetric cut) can be viewed as an additional lumped mass to the B2 system. On the other hand, higher modes become increasingly sensitive to small imperfections in the skin, ribs, or stringers of the tailplane, thus, the response for the normal data becomes increasingly inconsistent.

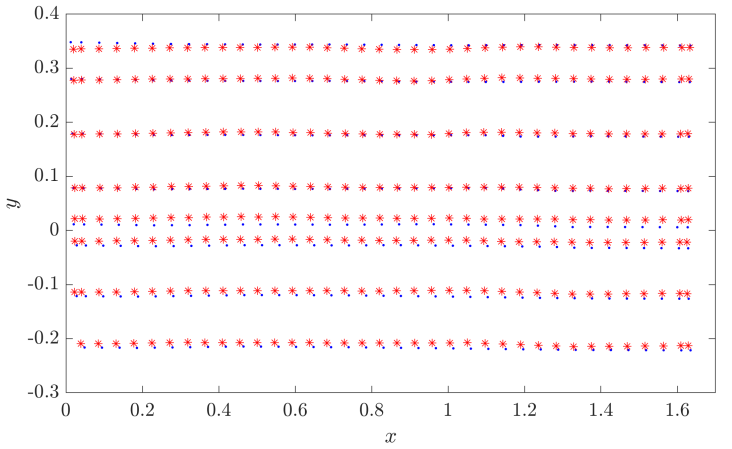
[4]For the C plane, the features do not have to be consistent, as the starboard wing (C2) is damaged.



(a) A scanning grid – 180 points.



(b) B scanning grid – 180 points.



(c) C scanning grid – 295 points.

Figure 5: Scanning grid corresponding to response locations: blue dots indicate the port wing (A1/B1/C1), red stars indicate starboard wing (A2/B2/C2). The reference point $\{x = 0, y = 0\}$ is shown on the wing schematic in Fig. 2.

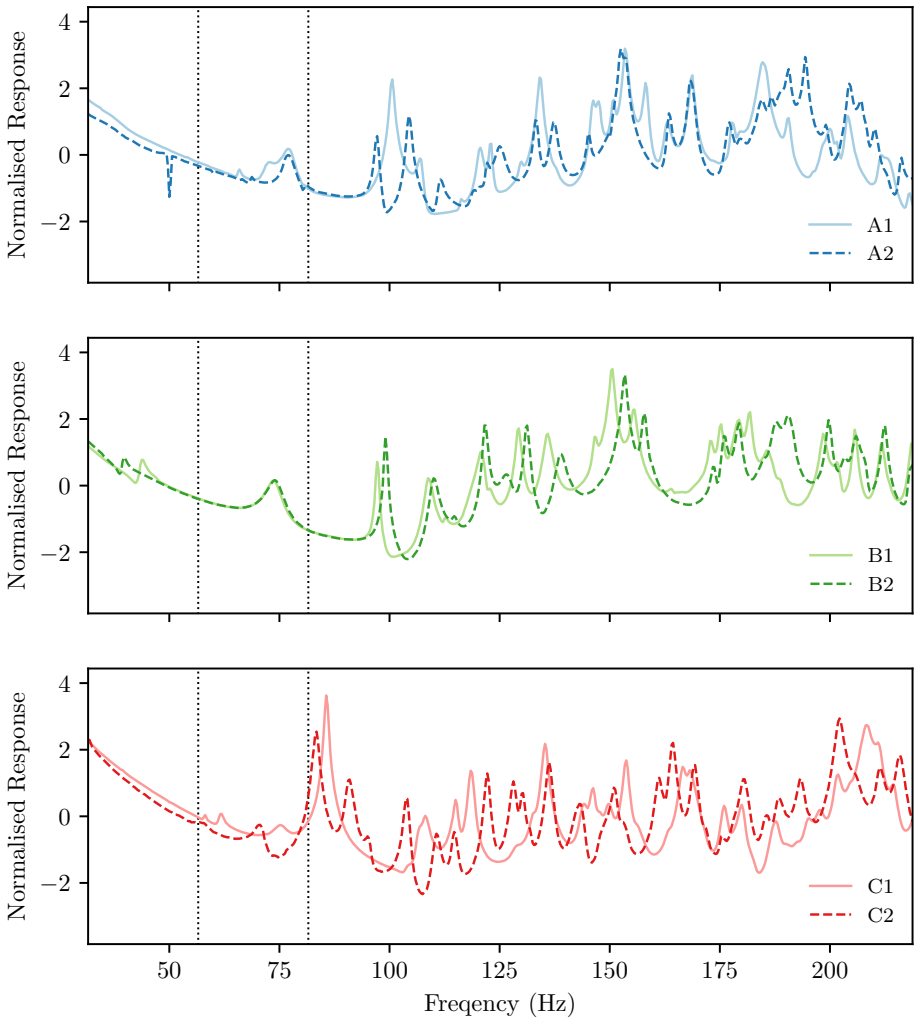


Figure 6: Frequency response functions for tailplane pairs, normalised in the log-space. Dotted lines highlight the range of consistent features selected for transfer learning.

At this stage, the authors emphasise that feature selection remains critical in transfer learning, as with other forms of machine learning. If the features prove too dissimilar, or represent different information, the risk of *negative transfer* can increase dramatically, even for very similar systems [12, 13]. In the worst-case scenario, predictions can be inferior to conventional domain-wise methods. In consequence, the available features should influence the choice in model and associated parameters to aid successful knowledge transfer, particularly in an unsupervised damage detection setting.

To define the transfer learning dataset, observations are re-sampled from uncertain representations of the experimental FRFs. According to [33], the coherence function $\gamma_p(\omega)$ can be used to estimate the uncertainty in the magnitude of the response at frequency ω. Specifically, assuming Gaussian statistics at each response point p, the standard deviation in the FRF magnitude is [34],

$$\sigma(|H_p(\omega)|) = \frac{\sqrt{1-\gamma_p^2(\omega)}}{|\gamma_p(\omega)|\sqrt{2N}}\,|H_p(\omega)| \tag{1}$$

where $N = 6$, the number of repeats for frequency-domain measurements. The associated uncertainty can be propagated to the *summed* FRF via a Monte-Carlo method [35] – this involves iterating the following steps: (i) a sample is taken from the uncertain FRF at each response point $|H_p(\omega)|$, (ii) the samples are summed $\sum_p |H_p|$ and then stored. Steps (i) and (ii) are repeated for a large number of iterations (10,0000), then the sample mean and standard deviation are calculated from the batch of summed FRFs. Figure 7 shows the resulting mean and deviation for the selected feature, between 56.56 and 81.56Hz, for all tailplanes.

Lastly, observations are sampled from the uncertain feature shown in Figure 7 for domains A, B and C. The multi-domain data are defined as follows:

A plane domain $\mathcal{D}_A$, data $\{\mathbf{x}_i^A\}_{i=1}^{n_A}$, 200 observations of normal condition data (100 each from A1/A2).

B plane domain $\mathcal{D}_B$, data $\{\mathbf{x}_i^B\}_{i=1}^{n_B}$, 200 observations of normal condition data (100 each from B1/B2).

C plane domain $\mathcal{D}_C$ with data $\{\mathbf{x}_i^C\}_{i=1}^{n_C}$, 100 observations of normal condition data (C1) and 100 observations of damage data (C2).

Importantly, the training data are defined as a random subset of 50% of the samples from the port-side FRFs *only*: A1/B1/C1 (i.e. 25% of the total data in each domain). Therefore, only data corresponding to the normal condition are available for training. The testing data include the remaining 50% of samples from the port wing, and the starboard examples. Thus, testing will verify the approximation of a single domain per tailplane pair, as well as standard generalisation.

The problem here is one of *exclusive* outlier analysis [36]: only normal data are available for training, and the task $\mathfrak{T}$ is to model the normal data data through f. Predictions then determine if future data are samples from the same underlying distribution (inlying), or not (outlying).

4.3. Benchmark: principal component analysis for novelty detection

Conventional damage detection is applied via Principal Component Analysis (PCA) and the Mahalanobis distance [36], whereby each domain is considered *separately*. These results are useful to motivate the need for transfer learning.

Firstly, PCA is applied to each domain, to project the data into a two-dimensional subspace. Note, two-dimensional subspaces are used throughout for consistency, and to aid visualisation. Briefly, PCA is a linear projection $\mathcal{X} \to \mathcal{Z}$, where $\{\mathbf{x}_i\} \in \mathcal{X} \in \mathbb{R}^M$ and $\{\mathbf{z}_i\} \in \mathcal{Z} \in \mathbb{R}^D$. In the $\mathcal{Z}$ subspace, the d^{th} projected dimension is a weighted sum $\mathbf{w}_d$ of the original features,

$$z_{id} = \mathbf{w}_d^\top \mathbf{x}_i \tag{2}$$

A total of D projection vectors $\{\mathbf{w}_1, \ldots, \mathbf{w}_D\}$ (M-dimensional) are defined, such that $d \in \{1, \ldots, D\}$. Thus, D is the dimensionality of the desired subspace (in this case $D = 2$), while M is the dimensionality of the original feature ($M = 70$, as there are 70 frequency bins in the response between 56.56 and 81.56Hz).

PCA uses the variance in the projected space $\mathbf{z}_i \in \mathcal{Z}, \mathbb{R}^D$ as a criterion for the optimal weight-vector for each dimension $\{\mathbf{w}_1, \ldots, \mathbf{w}_D\}$. Conveniently, the optimisation can be solved by defining a Lagrangian (an intuitive proof is available in [35]) and taking the (partial) derivative with respect to $\mathbf{w}_d$. In this form, the optimal weights are the solution to a standard eigenvector/value problem,

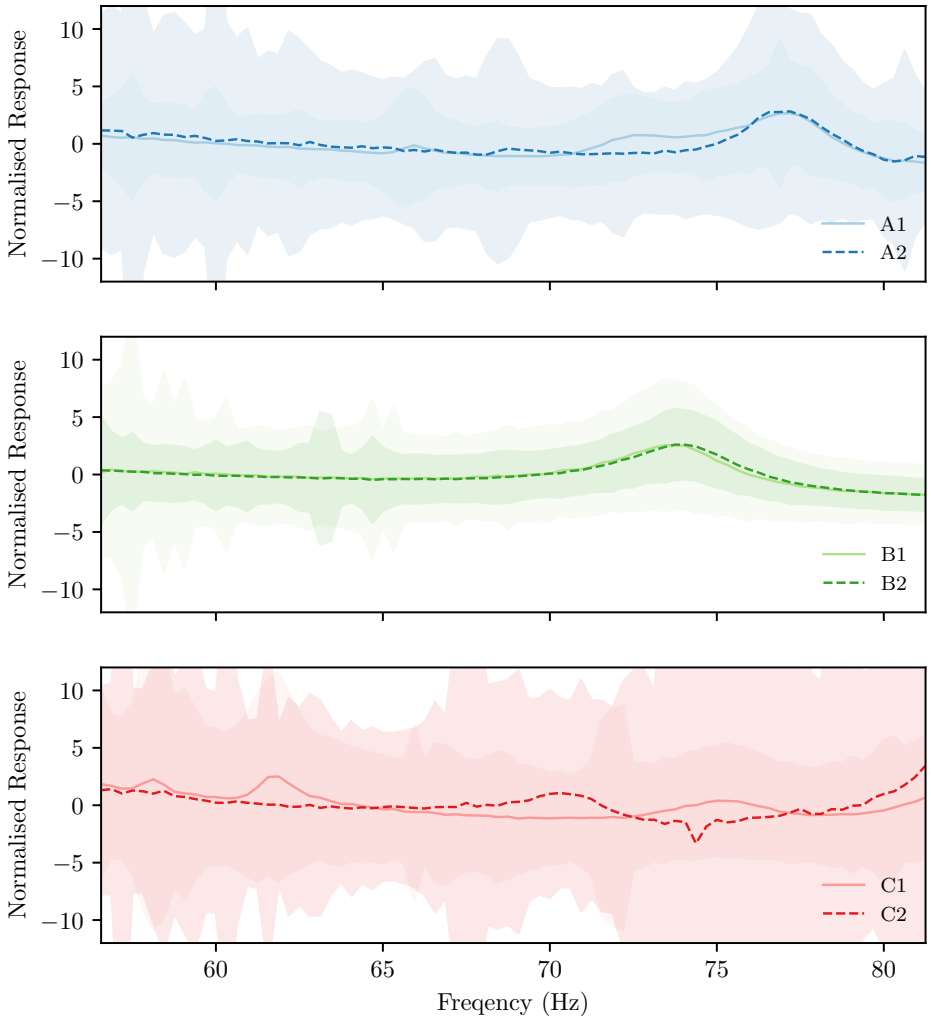
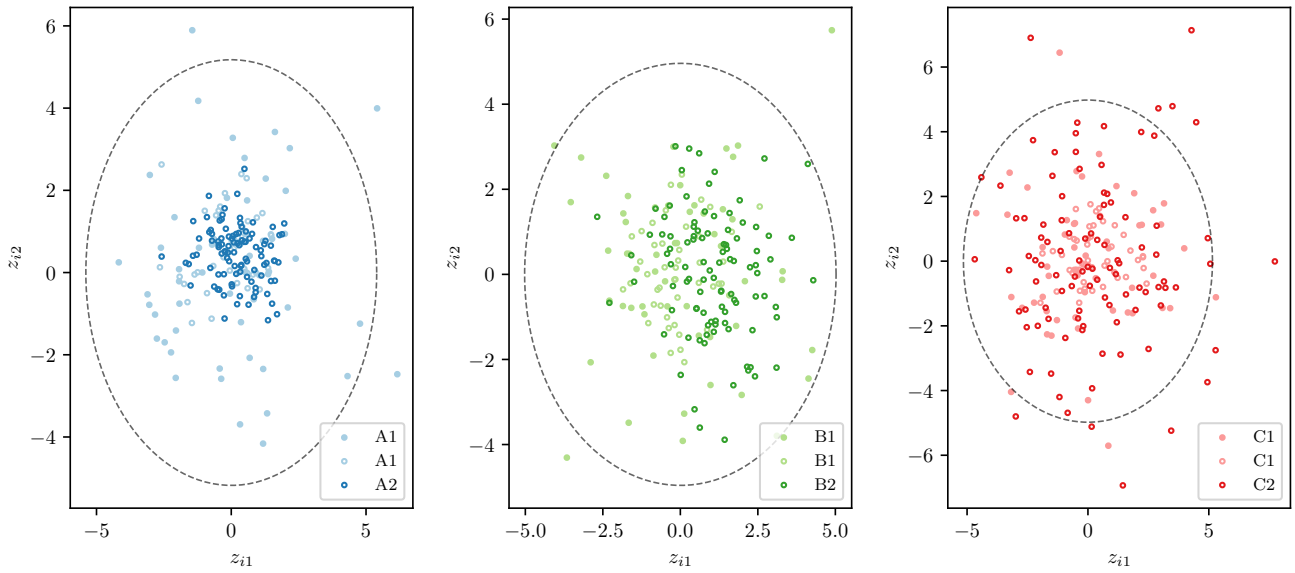


Figure 7: Feature between 56.56 – 81.56Hz, for all tailplanes. Shaded region shows one standard deviation around the mean.



(a) $\mathcal{D}_A$ (b) $\mathcal{D}_B$ (c) $\mathcal{D}_C$

Figure 8: Domain-wise PCA. Filled markers • represent training data, hollow markers ∘ represent test data. The dashed line represents the 95% confidence ellipse for f.

$$\mathbf{S}\mathbf{w}_d = \lambda_d \mathbf{w}_d \tag{3}$$

where $\mathbf{S}$ is the sample covariance of the original data, $\mathbf{S} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top$ (since the data are *normalised*). The eigenvector with the highest eigenvalue corresponds to the projected dimension with maximum variance $\mathbf{w}_1$, while the second highest eigenvalue corresponds to the projection with the second highest variance $\mathbf{w}_2$, and so on [35].

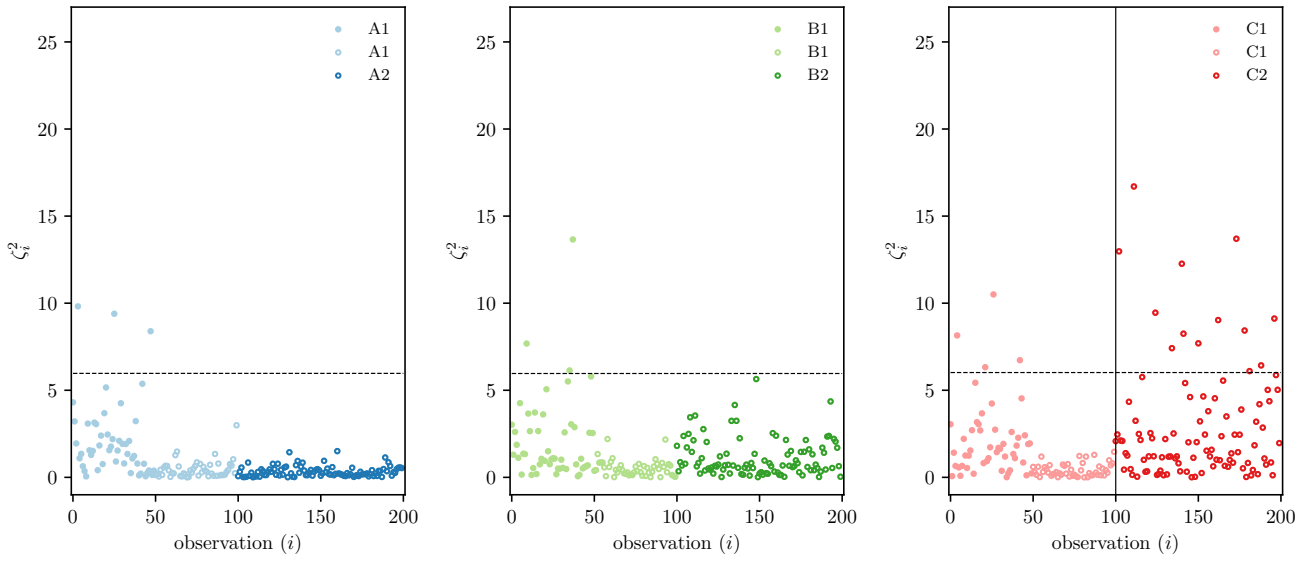
Only the training data should be used to find $\mathbf{w}_d$ for each domain. The weights $\{\mathbf{w}_1, \mathbf{w}_2\}$ can then be used to project the complete data into a two-dimensional subspace. Here, the sample mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ can be calculated for $\{\mathbf{z}_i\}_{i=1}^n$ to approximate the underlying distribution for the (normal) training data, assuming they are Gaussian distributed $f = p(\mathbf{z}_i) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ [36].

The PCA subspace for the tailplane domains (A, B and C), as well as the Gaussian model for normal data f, are shown in Figure 8. All the test data appear to be sampled from the same underlying distribution for A and B – this is intuitive, considering the aims of feature selection, as the test data correspond to the normal condition only. However, the distributions also remain similar in domain C, despite including *outliers* (the damaged wing, C2). In consequence, the PCA space appears insensitive to the novelty (damage) of interest.

To quantify *similarity*, the i^{th} datum in the two-dimensional subspace ($\mathbf{z}_i \in \mathcal{Z}$) can be compared to the normal condition (f) through the Mahalanobis-Squared-Distance (MSD). This measure of discordancy is regularly used for outlier analysis and damage detection [1, 36, 37],

$$\zeta_i^2 = (\mathbf{z}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{z}_i - \boldsymbol{\mu}) \tag{4}$$

In words, ζ_i^2 is the distance of $\mathbf{z}_i$ from the centre of the training data, *normalised* by the variance. Generally, an observation is considered outlying if ζ_i^2 passes a threshold or critical value; here, like the data, the threshold is determined via a Monte Carlo method [1, 36, 37] to define the 95% critical value.



(a) $\mathcal{D}_A$ (b) $\mathcal{D}_B$ (c) $\mathcal{D}_C$

Figure 9: Domain-wise Mahalanobis distance. Filled markers • represent training data, hollow markers ∘ represent test data. The solid vertical line separates the damage data (C2), the dashed line shows the critical value for damage detection.

Figure 9 shows the MSD for the data in each domain; these distances agree with observations made from the PCA space. The test data (∘) for A and B are inlying, as they correspond to normal condition FRFs. For domain C, however, the damage data (C2) incorrectly appear as inlying. To reiterate, the feature space defined by PCA is insensitive to damage, and the *outliers* are being *masked* by the normal condition [37].

Before transfer learning, Kernel PCA (KPCA) is applied to extend conventional domain-wise damage detection[5]. Via a kernel substitution, KPCA defines a nonlinear transformation of the original feature to maximise variance. Recalling equation (3) (assuming zero-mean for the projected data) the covariance matrix $\mathbf{S} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top$ is replaced with,

$$\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n}\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^\top \tag{5}$$

where $\phi(\mathbf{x}_i)$ is a nonlinear transform into an n-dimensional feature space. Standard PCA is applied in the new feature space, which implicitly defines a nonlinear projection in the original space, $\mathbf{x}_i \in \mathcal{X}$ [38]. Conveniently, the eigenvalue problem $\mathbf{C}\mathbf{w}_d = \lambda_d\mathbf{w}_d$ can be solved without having to work explicitly in the nonlinear feature space through the *kernel trick*. A nice proof is available in [38], which leads to the following expression for a nonlinear projection (in $\mathcal{X}$),

$$z_{id} = \sum_{j=1}^{n} a_{dj}k(\mathbf{x}_i, \mathbf{x}_j) \tag{6}$$

where $\mathbf{a}_d$ is the d^{th} largest eigenvector for the problem $\mathbf{K}\mathbf{a}_d = \lambda_d n\mathbf{a}_d$. The kernel function is $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top\phi(\mathbf{x}_j)$, and $\mathbf{K}$ is the kernel matrix, with elements $\mathbf{K}_{[i,j]} = k(\mathbf{x}_i, \mathbf{x}_j)\ \forall\ i, j \in \{1, \ldots, n\}$.

[5]Additionally, the KPCA benchmark will be more directly comparable to transfer component analysis.

Any valid choice of kernel $k(\cdot)$ can be used; for consistency, the Radial-Basis Function (RBF) is used throughout this work,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left\{\frac{|| \mathbf{x}_i - \mathbf{x}_j ||^2}{2l^2}\right\} \tag{7}$$

The parameter l defines the *length-scale*. Generally speaking, the kernel encodes the degree of *coupling* between two points in the feature space. It is widely known that defining kernel hyperparameters in an unsupervised (damage detection) setting proves to be challenging and application specific [39, 40]. The user can employ techniques such as the median heuristic [40] or reconstruction metrics [39] to help select this parameter. For consistency, the length scale is set to $l = 16$ throughout – this value was found to define an acceptable reconstruction error for KPCA, while remaining close to the median heuristic for the tailplane population data.

Again, the subspace and MSDs produced by KPCA lead to similar issues of masking, shown in Appendix A, such that the MSDs for damage data fail to pass the detection threshold. Clearly, *single domain* projections that maximise variance (PCA/KPCA) produce feature spaces that are insensitive to damage: an alternative projection is needed to aid separation of the damage data in $\mathcal{D}_C$. Transfer learning via domain adaptation offers one solution, such that normal data from all domains are considered *together* in a shared subspace. By extending the size of the effective training set, the criterion for projection can be adapted, and transfer learning should enable damage detection in $\mathcal{D}_C$, while remaining general in domains $\mathcal{D}_A$ and $\mathcal{D}_B$.

5. Transfer Component Analysis for Novelty Detection

Transfer Component Analysis (Pan *et al.* [14]) is used to match the distributions of the normal data from each tailplane in a shared latent space – i.e. Figure 1. The general assumption is that the marginal distributions between the source and target domains are different, such that $p(\mathbf{X}_s) \neq p(\mathbf{X}_t)$. Therefore, TCA looks to find a common latent representation for $\mathbf{X}_s = \{\mathbf{x}_i^s\}_{i=1}^{n_s} \in \mathcal{X}$ and $\mathbf{X}_t = \{\mathbf{x}_i^t\}_{i=1}^{n_t} \in \mathcal{X}$ which preserves the data configuration after transformation [14]. Specifically, a nonlinear transform of the feature space is learnt (similar to KPCA) $\psi : \mathcal{X} \to \mathcal{Z}$, where the marginals are approximately *matched*,

$$p\left(\psi\left(\mathbf{X}_s\right)\right) \approx p\left(\psi\left(\mathbf{X}_t\right)\right) \tag{8}$$

Additionally, the conditionals are assumed consistent, i.e. $p\left(\mathbf{Y}_s|\psi\left(\mathbf{X}_s\right)\right) = p\left(\mathbf{Y}_t|\psi\left(\mathbf{X}_t\right)\right)$. For damage detection in this context, these assumptions imply that: while the *training* data are distributed differently in each domain, they all correspond to *normal* structures, which can be matched in the latent space[6].

In brief terms, the transform ψ minimises the distance between the source and target data by defining a nonlinear mapping into a Reproducing Kernel Hilbert Space $\mathcal{H}$ (details/proofs can be found in [14]). By virtue of the kernel trick, the Maximum Mean Discrepancy (MMD) can then be used as the criterion to minimise the difference between the empirical means for the source and target data in the latent space [14],

$$\text{Dist}\left[p\left(\psi\left(\mathbf{X}_s\right)\right),\ p\left(\psi\left(\mathbf{X}_t\right)\right)\right] = \left|\left|\frac{1}{n_s}\sum_{i=1}^{n_s}\psi(\mathbf{x}_i^s) - \frac{1}{n_t}\sum_{i=1}^{n_t}\psi(\mathbf{x}_i^t)\right|\right|_{\mathcal{H}}^2 = \text{tr}\left(\mathbf{KL}\right) \tag{9}$$

here, $\mathbf{K} \in \mathbb{R}^{(n_s+n_t)\times(n_s+n_t)}$ is the kernel matrix for both the source and target data, $\mathbf{X} = \mathbf{X}_s \cup \mathbf{X}_t \in \mathbb{R}^{(n_s+n_t)\times M}$ (including cross domain distances) and $\mathbf{L}$ is the MMD matrix, such that,

[6] As this is exclusive outlier analysis, the damage data appear in testing.

$$\mathbf{L}_{i,j} = \begin{cases} \frac{1}{n_s^2} & \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}_s \\ \frac{1}{n_t^2} & \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}_t \\ \frac{-1}{n_s n_t} & \text{otherwise} \end{cases} \tag{10}$$

A matrix $\mathbf{W} \in \mathbb{R}^{(n_s+n_t)\times D}$ can be inserted to define a low-rank kernel embedding $\tilde{\mathbf{K}} = \mathbf{K}\mathbf{W}\mathbf{W}^\top\mathbf{K}$, which transforms the feature vectors into the D-dimensional latent space, where $D < n_t + n_s$ (in this case, $D = 2$, as with PCA). The embedding of the i^{th} point in the latent space becomes,

$$\mathbf{z}_i = [\mathbf{W}^\top\mathbf{K}]_{[:,i]}, \qquad \mathbf{z}_i \in \mathbb{R}^D$$

the matrix subscript $[:, i]$ refers to the i^{th} column. Considering the kernel embedding into the latent subspace, the distance can be re-written [14],

$$\text{Dist}\Big[p\left(\psi\left(\mathbf{X}_s\right)\right),\ p\left(\psi\left(\mathbf{X}_t\right)\right)\Big] = \text{tr}\left((\mathbf{K}\mathbf{W}\mathbf{W}^\top\mathbf{K})\mathbf{L}\right) \tag{11}$$

$$= \text{tr}\left(\mathbf{W}^\top\mathbf{K}\mathbf{L}\mathbf{K}\mathbf{W}\right) \tag{12}$$

For a given kernel, the optimal weights can be determined by minimising (12); in this way, the source and target data are brought together in the subspace:

$$\begin{aligned} \min_{\mathbf{W}} &= \text{tr}\left(\mathbf{W}^\top\mathbf{K}\mathbf{L}\mathbf{K}\mathbf{W}\right) + \alpha\ \text{tr}\left(\mathbf{W}^\top\mathbf{W}\right) \\ \text{s.t.} \quad & \mathbf{W}^\top\mathbf{K}\mathbf{H}\mathbf{K}\mathbf{W} = \mathbf{I}_D \end{aligned} \tag{13}$$

The term $\alpha\ \text{tr}\left(\mathbf{W}^\top\mathbf{W}\right)$ regularises the complexity of the projection, such that α is a trade-off parameter. The constraint $\mathbf{W}^\top\mathbf{K}\mathbf{H}\mathbf{K}\mathbf{W} = \mathbf{I}_D$ is added to prevent the trivial solution $\mathbf{W} = \mathbf{0}$, whereby all the data are transformed onto the origin. $\mathbf{I}_D$ is the identity matrix, dimension D. Finally, $\mathbf{H}$ is a centring matrix, where $\mathbf{H} = \mathbf{I}_{ns+nt} + \frac{1}{n_s+n_t}\mathbf{1}$, and $\mathbf{1}$ is a $(n_s + n_t) \times (n_s + n_t)$ matrix of ones.

As with PCA, the optimal transform can be defined as another eigenvalue problem, by defining a Lagrangian from (13) and taking the partial derivative w.r.t. $\mathbf{W}$ [14],

$$\Big[\left(\mathbf{K}\mathbf{L}\mathbf{K} + \alpha\mathbf{I}\right)^{-1}\mathbf{K}\mathbf{H}\mathbf{K}\Big]\,\mathbf{w}_d = \lambda\mathbf{w}_d \tag{14}$$

where $\mathbf{W} = \{\mathbf{w}_d\}_{d=1}^{D}$ correspond to the D leading eigenvalues.

6. Results and Discussion

As discussed, $D = 2$ for the TCA subspace in the tailplane applications; the RBF kernel is used (7) with length $l = 16$ (to be consistent with PCA/KPCA). In line with other work [14], the regularisation trade-off parameter is set to $\alpha = 0.1$; in fact, for these data, any value $\alpha > 1 \times 10^{-10}$ was found to avoid rank deficiency of the denominator for the generalised eigendecomposition (14) (results were insensitive to changes above this value).

Three, pair-wise transfer experiments are investigated:

- A $\leftrightarrow$ C: using the normal data from $\mathcal{D}_A$ and $\mathcal{D}_C$ to improve damage detection in $\mathcal{D}_C$
- A $\leftrightarrow$ B: using the normal data from $\mathcal{D}_A$ and $\mathcal{D}_B$ to ensure generalisation in both $\mathcal{D}_A$ and $\mathcal{D}_B$

- B $\leftrightarrow$ C: using the normal data from $\mathcal{D}_B$ and $\mathcal{D}_C$ to improve damage detection in $\mathcal{D}_C$

Considering exclusive outlier analysis, the distinction between the source and target domains becomes less relevant, as both domains are labelled/assumed *normal* (i.e. $y_i^s = y_i^t = 1 \; \forall i$ in training). As a result, the proposed experiments cover all possible source/target combinations. That said, in this application, there is a practical focus to improve damage detection in domain C.

Interestingly, as each domain is implicitly labelled during exclusive outlier analysis, unsupervised TCA becomes (somewhat) analogous to a *multi-task learning*[7]; for details, the reader is referred to [13].

6.1. Knowledge transfer: A $\leftrightarrow$ C

TCA successfully defines damage-sensitive representation of the data in the latent space $\mathcal{Z}$, by distribution matching for $\mathcal{D}_A$ and $\mathcal{D}_C$. Visually, Figure 10a shows that the damage data (C2) have been projected out from the normal condition cluster f. All of the MSD values in Figure 10b that correspond to damage are correctly flagged as outlying, with zero false negatives (below the threshold).

The results are quantified and compared to the PCA benchmarks in Figure 10c. For each domain, the relevant metric(s) is (are) displayed. For domain A, the True Negative Rate (TNR) is provided only (as all the data are inlying),

$$\text{TNR} = \frac{\text{correctly predicted inliers}}{\text{number inliers in the test set}}$$

For domain C, the True Positive Rate (TPR) is also provided,

$$\text{TPR} = \frac{\text{correctly predicted outliers}}{\text{number of outliers in the test set}}$$

The TPR for domain C is important, as TPR ≈ 1 indicates sensitivity to damage. On the other hand, a TNR ≈ 1 is necessary to indicate good generalisation to test examples of the normal condition.

Therefore, Figure 10c quantifies the success of transfer learning, which enables damage detection in domain C through distribution matching of the normal data from $\mathcal{D}_A$. Specifically, the model remains general, with the TNR at 1, while the TPR improves significantly: from 0.13 and 0.00 for PCA and KPCA (respectively), to TPR = 1 through TCA. In other words, domain adaptation corresponds to an 87% TPR increase compared to conventional (domain-wise) damage detection.

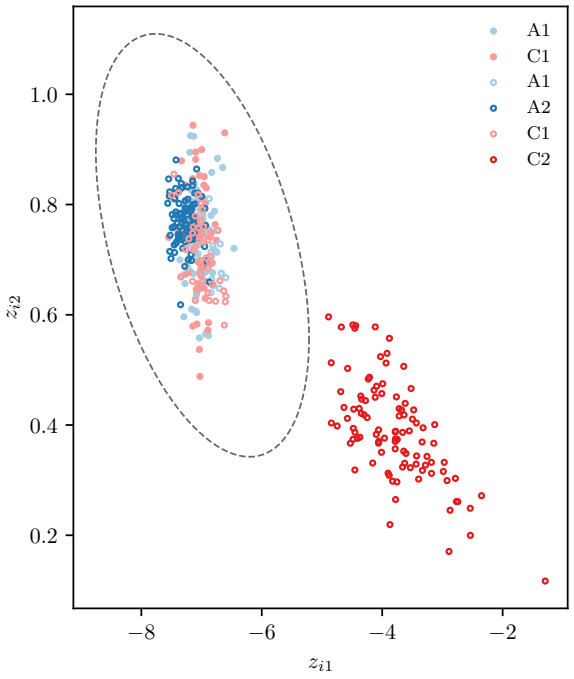
When assuming Gaussian statistics for f, with a 95% threshold, it is acknowledged that the TNR should be approximately 0.95 (corresponding to a 5% false positive rate). The low number of false positives is (unsurprisingly) due to the fact that data appear non-Gaussian in the $\mathcal{Z}$-space; the simplified representation proves suitable, however, to detect outlying observations in this application, as in previous work [1, 7, 36, 37].

6.2. Knowledge transfer: A $\leftrightarrow$ B

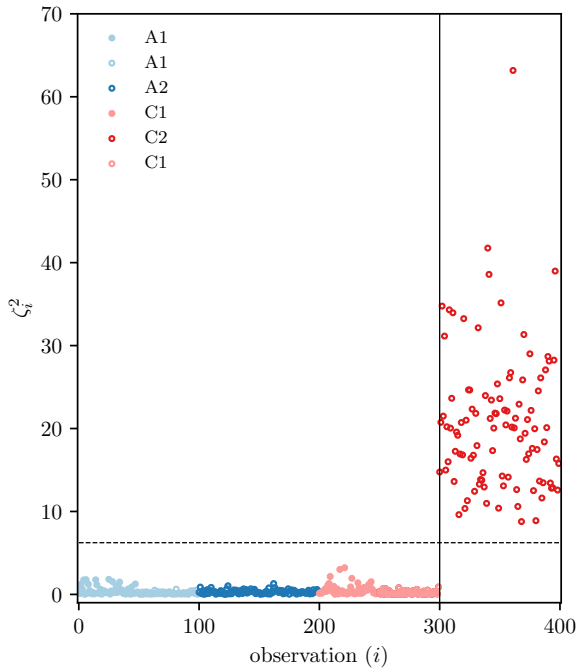
When distribution matching between $\mathcal{D}_A$ and $\mathcal{D}_B$, the model of the normal condition data remains general between the port and starboard sides. Good generalisation between A and B verifies the approximation of a single domain for each tailplane pair. Generalisation can be observed visually in Figure 11a, as the data from all structures appear similar and approximately from the same underlying distribution.

Some drift can be observed for the data sampled from B2, which leads to a false positive in the MSDs (normal data crossing the threshold) in Figure 11b. While this situation is not ideal, it is unsurprising

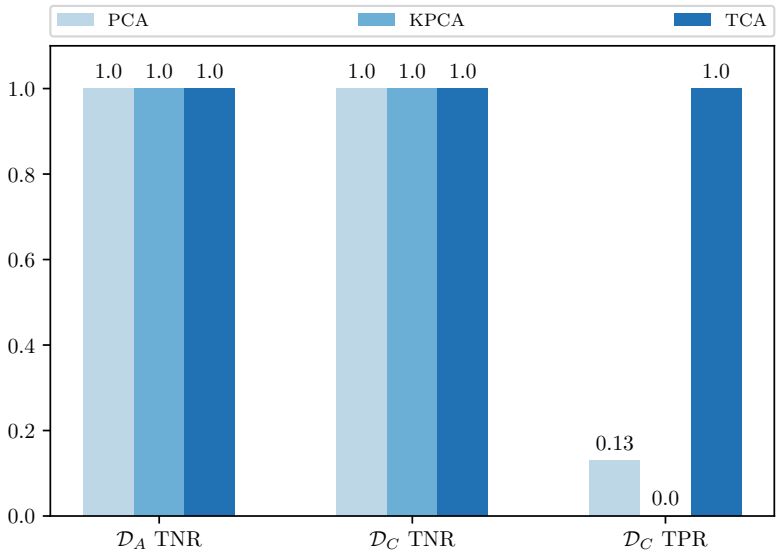
[7] Where a common task f is improved for multiple (labelled) domains, as opposed to transferring label information from a source to target domain [13].



(a) $\mathcal{Z}$: filled markers • represent training data, hollow markers ∘ represent test data. The dashed line shows the confidence ellipse for damage detection.



(b) MSD: filled markers • represent training data, hollow markers ∘ represent test data. The solid vertical line separates the damage data (C2), the dashed line shows the critical value for damage detection.



(c) Metrics of interest for each domain: true negative rate (TNR), true positive rate (TPR).

Figure 10: Transfer learning for novelty detection between tailplane A and tailplane C.

physically, due to the mass difference between the B1/B2 structures. In consequence, a single outlier represents good generalisation for the normal data across domains A and B, such that the TNR remains at 99%[8].

6.3. Knowledge transfer: $B \leftrightarrow C$

The final experiment matches the data from B with the normal data from C. As with transfer between A and C (Figure 10), the data from the B can also be used to define damage-sensitive features in the shared latent space. Again, the features are shown visually in the $\mathcal{Z}$-space (in Figure 12a) and through the MSD (in Figure 12b) where the damage data (C2) are distinct from the remaining normal data (B1/B2/C1). In agreement with transfer from A to B, there is drift in the normal data from $\mathcal{D}_B$, leading to a false positive for B2 (i.e. TNR = 0.99). This is a small trade-off, however, considering the mass differences in B1/B2 and the increase in sensitivity to damage for domain C (again, 87% compared to conventional methods) with no false negatives (TPR = 1).

7. Concluding remarks

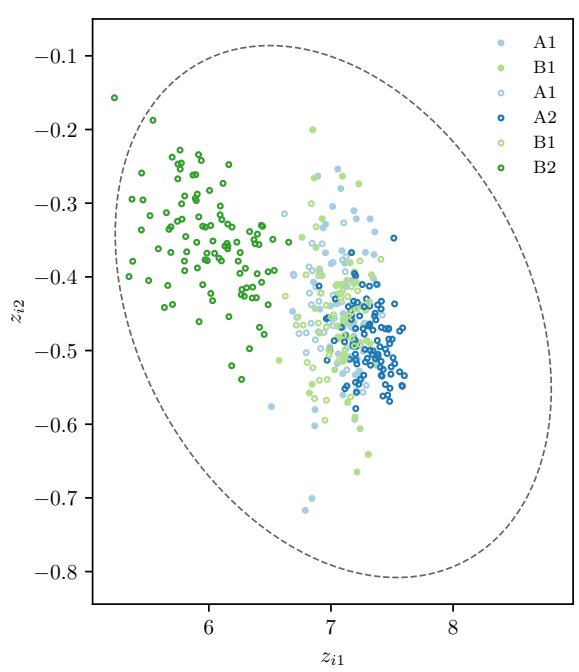
A population-based approach to SHM has been utilised to *share* information between a group of similar (but not identical) structures. Specifically, transfer learning, via domain adaptation, has been utilised for damage detection through an experimental case study concerning aircraft tailplanes. By considering the collective data from a population of three tailplane pairs, knowledge transfer enables the definition of damage-sensitive features, where conventional methods fail (domain-wise PCA/KPCA). Specifically, Transfer Component Analysis (TCA) is utilised to *match* the normal condition data from different structures through a nonlinear transformation onto a shared latent space, which defines a damage-sensitive representation. When distribution matching, TCA relies on a statistical distance between observations from different structures; thus, there must be enough data to embed the underlying distributions through an appropriate kernel. Utilising different pair-wise comparisons across the population, transfer learning leads to an 87% increase in the sensitivity to damage (true positive rate).

As with all forms of machine learning, feature and model selection are application specific; these steps prove critical to the success of transfer learning for unsupervised damage detection. Future work should investigate ways in which feature/model selection can be automated and enhanced within unsupervised transfer learning for novelty detection.

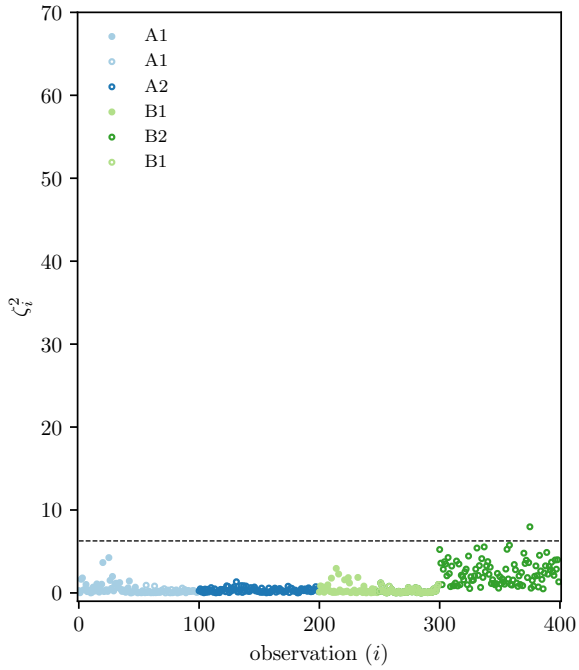
Acknowledgements

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) through Grant references EP/R003645/1, EP/R004900/1, EP/R006768/1, EP/N010884/1.

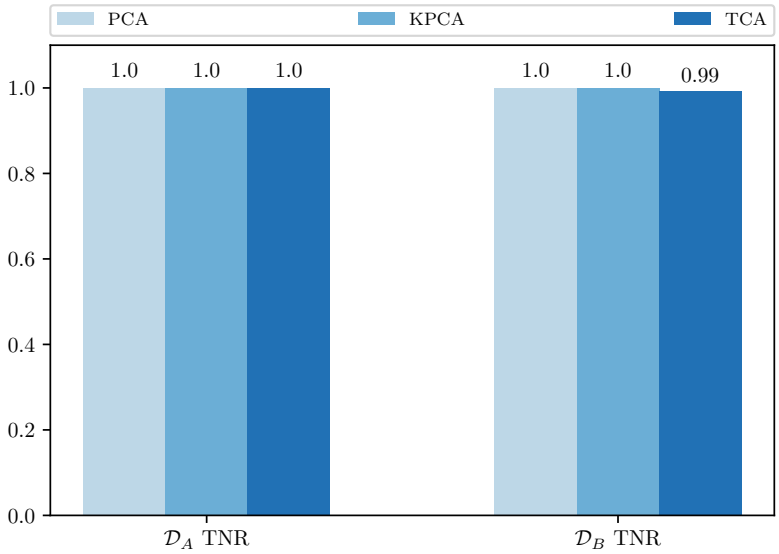
[8]The TPR is not provided in Figure 11c for domains A/B, as they contain inlying (negative) data only.



(a) $\mathcal{Z}$: filled markers • represent training data, hollow markers ∘ represent test data. The dashed line shows the confidence ellipse for damage detection.

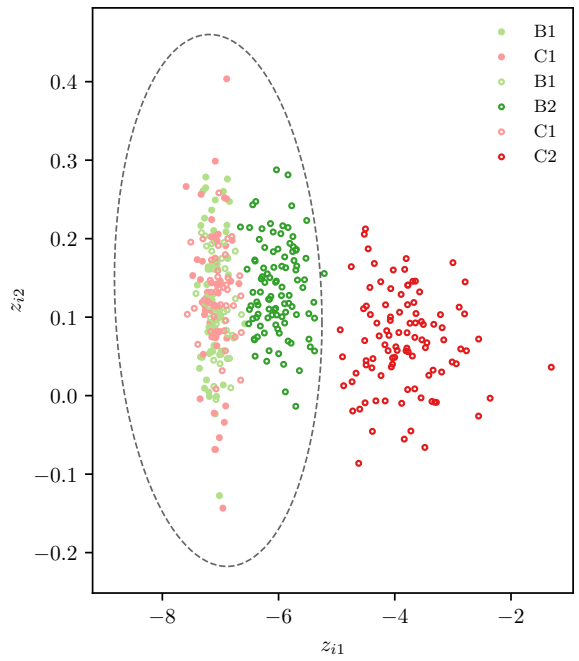


(b) MSD: filled markers • represent training data, hollow markers ∘ represent test data. The dashed line shows the critical value for damage detection.

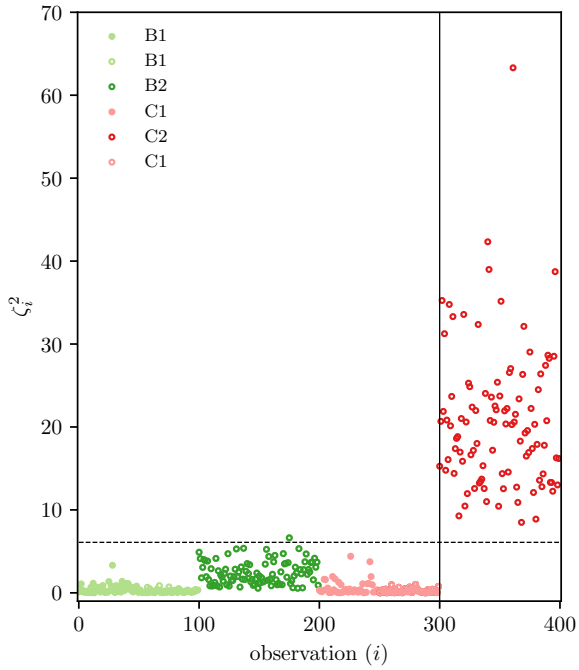


(c) True negative rate (TNR) for each domain.

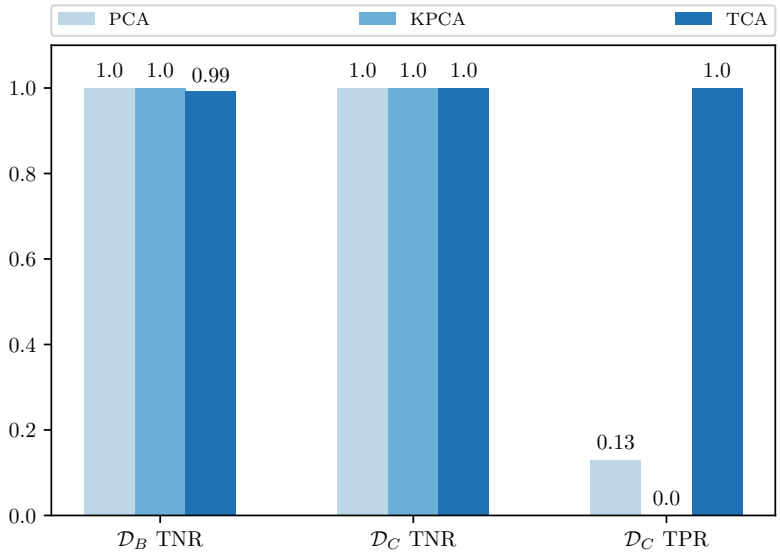
Figure 11: Transfer learning for novelty detection between tailplane A and tailplane B.



(a) $\mathcal{Z}$: filled markers • represent training data, hollow markers ∘ represent test data. The dashed line shows the confidence ellipse for damage detection.



(b) MSD: filled markers • represent training data, hollow markers ∘ represent test data. The solid vertical line separates the damage data (C2), the dashed line shows the critical value for damage detection.



(c) Metrics of interest for each domain: true negative rate (TNR), true positive rate (TPR).

Figure 12: Transfer learning for novelty detection between tailplane B and tailplane C.

References

[1] C. Farrar and K. Worden. *Structural Health Monitoring: a Machine Learning Perspective*. John Wiley and Sons, 2012.

[2] B. Rao. *Handbook of Condition Monitoring*. Elsevier, 1996.

[3] J. Blitz and G. Simpson. *Ultrasonic Methods of Non-Destructive Testing*, volume 2. Springer Science & Business Media, 1995.

[4] ISO 13372 / BS ISO 17359. Condition Monitoring and Diagnostics of Machines. Standard, International Organization for Standardization, 2012.

[5] E. R. Brown, N. N. McCollom, E.-E. Moore, and A. Hess. Prognostics and health management a data-driven approach to supporting the F-35 lightning II. In *2007 IEEE Aerospace Conference*, pages 1–12. IEEE, 2007.

[6] T. Tinga and R. Loendersloot. Aligning PHM, SHM and CBM by understanding the physical system failure behaviour. In *European Conference on the Prognostics and Health Management Society*, 2014.

[7] H. Sohn, C. R. Farrar, F. M. Hemez, D. D. Shunk, D. W. Stinemates, B. R. Nadler, and J. J. Czarnecki. A review of structural health monitoring literature: 1996–2001. *Los Alamos National Laboratory, USA*, 1, 2003.

[8] P. D. Samuel and D. J. Pines. A review of vibration-based techniques for helicopter transmission diagnostics. *Journal of Sound and Vibration*, 282(1-2):475–508, 2005.

[9] L. Bull, T. Rogers, C. Wickramarachchi, E. Cross, K. Worden, and N. Dervilis. Probabilistic active learning: An online framework for structural health monitoring. *Mechanical Systems and Signal Processing*, 134:106294, 2019.

[10] L. Bull, P. Gardner, J. Gosliga, T. Rogers, N. Dervilis, E. Cross, E. Papatheou, A. Maguire, C. Campos, and K. Worden. Foundations of population-based SHM, part I: Homogeneous populations and forms. *Mechanical Systems and Signal Processing*, 148:107141, 2020.

[11] J. Gosliga, P. Gardner, L. Bull, N. Dervilis, and K. Worden. Foundations of population-based SHM, part II: Heterogeneous populations–graphs, networks, and communities. *Mechanical Systems and Signal Processing*, 148:107144, 2020.

[12] P. Gardner, L. Bull, J. Gosliga, N. Dervilis, and K. Worden. Foundations of population-based SHM, part III: Heterogeneous populations–mapping and transfer. *Mechanical Systems and Signal Processing*, 149:107142, 2020.

[13] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

[14] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010.

[15] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46(4):1–37, 2014.

[16] A. Tsymbal. The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 106(2):58, 2004.

[17] B. Peeters and G. De Roeck. One-year monitoring of the Z24-bridge: environmental effects versus damage events. *Earthquake Engineering & Structural Dynamics*, 30(2):149–171, 2001.

[18] P. Gardner, X. Liu, and K. Worden. On the application of domain adaptation in structural health monitoring. *Mechanical Systems and Signal Processing*, 138:106550, 2020.

[19] M. M. Moya, M. W. Koch, and L. D. Hostetler. One-class classifier networks for target recognition applications. In *Proceedings World Congress on Neural Networks*, pages 359–367, 1993.

[20] A. Arnold, R. Nallapati, and W. W. Cohen. A comparative study of methods for transductive transfer learning. In *Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007)*, pages 77–82. IEEE, 2007.

[21] S. Dorafshan, R. J. Thomas, and M. Maguire. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Construction and Building Materials*, 186: 1031–1045, 2018.

[22] Y. Gao and K. M. Mosalam. Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33(9):748–768, 2018.

[23] K. Jang, N. Kim, and Y.-K. An. Deep learning–based autonomous concrete crack evaluation through hybrid image scanning. *Structural Health Monitoring*, 18(5-6):1722–1737, 2019.

[24] P. Cao, S. Zhang, and J. Tang. Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning. *IEEE Access*, 6:26241–26253, 2018.

[25] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang. A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. *Sensors*, 17(2):425, 2017.

[26] X. Li, W. Zhang, Q. Ding, and J.-Q. Sun. Multi-layer domain adaptation method for rolling bearing fault diagnosis. *Signal Processing*, 157:180–197, 2019.

[27] Q. Wang, G. Michau, and O. Fink. Domain adaptive transfer learning for fault diagnosis. In *2019 Prognostics and System Health Management Conference (PHM-Paris)*, pages 279–285. IEEE, 2019.

[28] G. Michau and O. Fink. Domain adaptation for one-class classification: monitoring the health of critical systems under limited information. *International Journal of Prognostics and Health Management*, 10: 11, 2019.

[29] D. Chakraborty, N. Kovvali, B. Chakraborty, A. Papandreou-Suppappola, and A. Chattopadhyay. Structural damage detection with insufficient data using transfer learning techniques. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2011*, volume 7981, page 798147. International Society for Optics and Photonics, 2011.

[30] P. Gardner, R. Fuentes, N. Dervilis, C. Mineo, S. Pierce, E. Cross, and K. Worden. Machine learning at the interface of structural health monitoring and non-destructive evaluation. *Philosophical Transactions of the Royal Society A*, 378(2182):20190581, 2020.

[31] E. Papatheou, T. Rahman, R. Barthorpe, J. Park, and K. Worden. An experimental investigation of feature complexity and diversity in nominally similar test structures. *Proceedings of ISMA2014, Leuven, Belgium*, 2014.

[32] G. Manson, K. Worden, and D. Allman. Experimental validation of a structural health monitoring methodology: Part II. novelty detection on a Gnat aircraft. *Journal of Sound and Vibration*, 259(2): 345–363, 2003.

[33] J. S. Bendat and A. G. Piersol. *Engineering Applications of Correlation and Spectral Analysis*. 1980.

[34] K. Worden. Confidence bounds for frequency response functions from time series models. *Mechanical Systems and Signal Processing*, 12(4):559–569, 1998.

[35] S. Rogers and M. Girolami. *A First Course in Machine Learning*. CRC Press, 2016.

[36] K. Worden, G. Manson, and N. R. Fieller. Damage detection using outlier analysis. *Journal of Sound and Vibration*, 229(3):647–667, 2000.

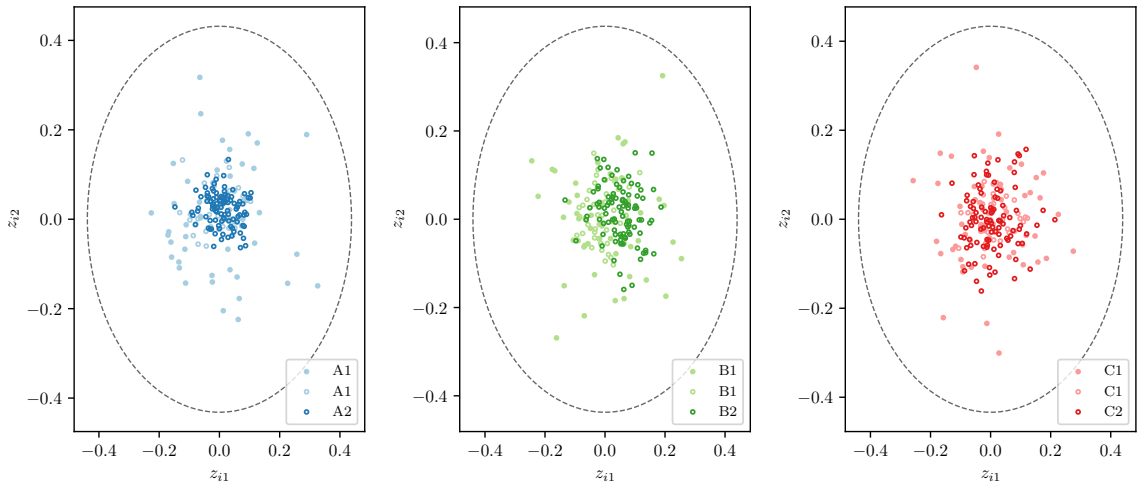
[37] L. Bull, K. Worden, R. Fuentes, G. Manson, E. Cross, and N. Dervilis. Outlier ensembles: A robust method for damage detection and unsupervised feature extraction from high-dimensional data. *Journal of Sound and Vibration*, 453:126–150, 2019.

[38] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[39] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch. Kernel PCA and de-noising in feature spaces. *Advances in Neural Information Processing Systems*, 11:536–542, 1998.

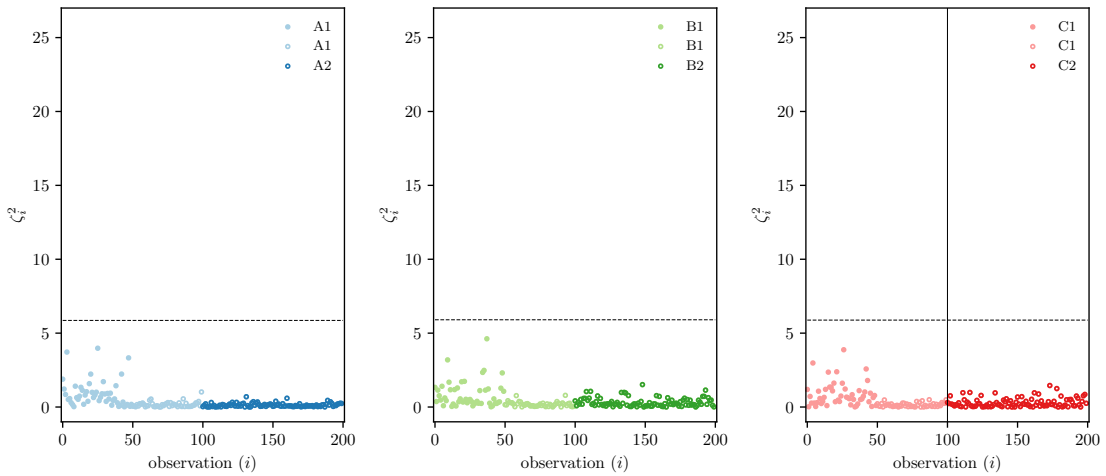
[40] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Appendix A. Kernel PCA – Masked Outliers Domain C



(a) $\mathcal{D}_A$ (b) $\mathcal{D}_B$ (c) $\mathcal{D}_C$

Figure A.13: Domain-wise KPCA. Filled markers • represent training data, hollow markers ∘ represent test data. The dashed line shows the 95% confidence ellipse for f.



(a) $\mathcal{D}_A$ (b) $\mathcal{D}_B$ (c) $\mathcal{D}_C$

Figure A.14: Domain-wise Mahalanobis distance for KPCA. Filled markers • represent training data, hollow markers ∘ represent test data. The solid vertical line separates the damage data (C2), the dashed line shows the critical value for damage detection.