

On the Application of Domain Adaptation for Aiding Supervised SHM Methods

PAUL GARDNER and KEITH WORDEN

ABSTRACT

The lack of available damage state data is a significant challenge within the field of Structural Health Monitoring (SHM). When data are obtainable for a given system of interest, a variety of machine learning approaches have been successful in addressing a range of supervised SHM problems. However, these methods assume that the training and testing data sets are drawn from the same distribution; as a consequence damage state data must be collected for each new structure and/or damage scenario considered, which is often infeasible and/or not economically viable. In these contexts it is useful to transfer knowledge obtained from known damage state data to different, but related contexts (or domains) of interest. By utilising transfer learning, knowledge obtained from different structures and/or damage scenarios can be used to improve learners in various target domains. Domain adaptation, a subcategory of transfer learning, is concerned with scenarios where the data distributions across source and target domains are different; and is demonstrated here to be applicable to SHM in a numerical and experimental case study.

INTRODUCTION

Machine learning approaches to Structural Health Monitoring (SHM) are successful when labelled damage state data are available [1]. However, in most scenarios these data are not obtainable, as they would be infeasible to collect. In addition, even when labelled damage state data are available, the trained classifier assumes that the training and testing sets are drawn from the same distribution; this means that for every new structure considered, damage state data must be collected for the classifier to work.

The work presented in this paper can be considered part of the broader category of *population-based* SHM [2], creating methods that generalise across a population of structures. Here the question posed is, can we map known damage labels for one struc-

Paul Gardner, Dynamics Research Group, University of Sheffield, Mappin Street, Sheffield, S1 3JD, U.K.. Email: p.gardner@sheffield.ac.uk

Keith Worden, Dynamics Research Group, University of Sheffield

ture onto a different structure — or even from a simulation onto an operational structure. This capability would allow a classifier learnt on one member of a population to be applied across the complete group.

Domain adaptation provides a solution to this problem by transforming the feature space, such that a single classifier trained on the transformed source data generalises to a given target structure, where labelled data are not available. Specifically, this paper utilises domain adaptation in the context of two structures, one in which labelled data are available and another where the data are unavailable. The structures considered are *homogeneous* i.e. nominally similar in topology, and the classification problem is two class i.e. undamaged and damaged.

This paper begins with an overview of domain adaptation outlining two methods: Transfer Component Analysis (TCA) and Joint Domain Adaption (JDA). Two case studies are presented: one numerical and the other experimental. Finally conclusions are made, whereby domain adaptation is shown as a viable method for performing population-based SHM on homogeneous structures.

DOMAIN ADAPTATION

Before defining transfer learning it is important to define two key quantities, a *domain* \mathcal{D} and a *task* \mathcal{T} . A domain refers to a feature space \mathcal{X} and a marginal probability distribution $\mathcal{D} = \{\mathcal{X}, p(X)\}$ where $X = \{\mathbf{x}_i\}_{i=1}^N \in \mathcal{X}$ i.e. a finite sample set from \mathcal{X} . A task for a given domain is defined by as $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$, where \mathcal{Y} is a label space and $f(\cdot)$ is a predictive function learnt from a training data set $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $y \in \mathcal{Y}$ (can also be seen as a conditional distribution $p(y | X)$).

Transfer learning considers scenarios where, given a source domain \mathcal{D}_s and task \mathcal{T}_s and target domain \mathcal{D}_t and task \mathcal{T}_t , can the learning of the target predictive function $f_t(\cdot)$ be improved using knowledge from \mathcal{D}_s and \mathcal{T}_s , where $\mathcal{D}_s \neq \mathcal{D}_t$ and $\mathcal{T}_s \neq \mathcal{T}_t$. Transfer learning methods are then based on whether $\mathcal{X}, p(X), \mathcal{Y}$ or $p(y | X)$ are consistent across the source and target [3].

Domain adaptation is a sub-category of transfer learning, where the assumption is that the feature space and label spaces are the same i.e. $\mathcal{X}_s = \mathcal{X}_t$ and $\mathcal{Y}_s = \mathcal{Y}_t$, but the marginal and (potentially) conditional densities of the finite sets are not equal, i.e. $p(X_s) \neq p(X_t)$ and/or $p(Y_s | X_s) \neq p(Y_t | X_t)$. The scenarios in which domain adaptation is applicable will occur when the classifier will not generalise across the source and target domains due to differences in the distributions. Approaches have therefore been developed to reduce these differences by minimising the distance (or divergence) between the distributions through some mapping $\phi(\cdot)$, such that $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s | \phi(X_s)) \approx p(Y_t | \phi(X_t))$.

Domain adaptation methods are therefore suited to *homogeneous* populations, i.e. nominally-identical structures, as the feature space and potential labels will be the same, but the distributions from the finite sample set can be different.

Transfer Component Analysis

TCA assumes $p(X_s) \neq p(X_t)$ but that $p(Y_s | X_s) = p(Y_t | X_t)$ i.e. the marginal distributions are very different. The method seeks to learn a nonlinear transformation

from the feature space to a Reproducing Kernel Hilbert Space (RKHS) i.e. $\phi : \mathcal{X} \rightarrow \mathcal{H}$, where $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s | \phi(X_s)) = p(Y_t | \phi(X_t))$ [4]. The approach utilises the (squared) Maximum Mean Discrepancy (MMD) distance as a criterion — the difference between two empirical means through a nonlinear mapping in a RKHS — where a kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$, forming Equation (1),

$$\text{Dist}(p(\tilde{X}_s), p(\tilde{X}_t)) = \text{tr}(KM) \quad (1)$$

where \tilde{X} are the transformed features, $K = \phi(X)^\top \phi(X) \in \mathbb{R}^{(N_s+N_t) \times (N_s+N_t)}$ given $X = X_s \cup X_t \in \mathbb{R}^{(N_s+N_t) \times D}$ where D is the dimension of the features, and M is the MMD matrix as defined by,

$$M(i, j) = \begin{cases} \frac{1}{N_s^2}, & x_i, x_j \in X_s \\ \frac{1}{N_t^2}, & x_i, x_j \in X_t \\ \frac{-1}{N_s N_t}, & \text{otherwise.} \end{cases} \quad (2)$$

Utilising the low-rank empirical kernel embedding $\tilde{K} = KWW^\top K$ [5], the distance can be rewritten as in Equation (3); where $W \in \mathbb{R}^{(N_s+N_t) \times k}$ are the weights which perform a reduction and transformation,

$$\text{Dist}(p(\tilde{X}_s), p(\tilde{X}_t)) = \text{tr}(W^\top KMKW) \quad (3)$$

The problem of finding the optimal weights can now be formed as an optimisation problem, where by minimising the weights W , the marginal distributions are brought together in the transformed space. The problem is performed under a square Frobenius norm regularisation constraint to control the complexity of W , and subject to kernel Principle Component Analysis (PCA) such that the trivial solution $W = 0$ is avoided as shown in,

$$\min_{W^\top KHKW = \mathbb{I}} = \text{tr}(W^\top KMKW) + \mu \text{tr}(W^\top W) \quad (4)$$

where μ is a regularisation trade-off parameter, $H = \mathbb{I} - 1/(N_s + N_t)\mathbf{1}\mathbf{1}^\top$ is a centring matrix, \mathbb{I} is an identify matrix and $\mathbf{1}$ a matrix of ones. Using a Lagrangian approach, the following optimisation problem can be solved for W as the eigenvectors corresponding to the k smallest eigenvalues in the problem,

$$(KMK + \mu\mathbb{I})W = KHKW\phi \quad (5)$$

Finally the k -dimensional transformed feature spaces is calculated by $Z = KW \in \mathbb{R}^{(N_s+N_t) \times k}$. Once obtained, a classifier can be trained in the transformed space using the source data and subsequently implemented on the target data.

Joint Domain Adaptation

JDA assumes $p(Y_s, X_s) \neq p(Y_t, X_t)$ i.e. the joint distributions are different. The method seeks to learn a nonlinear transform from the feature space to a RKHS i.e. $\phi : \mathcal{X} \rightarrow \mathcal{H}$, where $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s | \phi(X_s)) \approx p(Y_t | \phi(X_t))$ at the same time [6].

However the conditional in the target domain $p(Y_t | X_t)$ cannot be modelled directly as there are no target labelled data. To overcome this problem JDA utilises a pseudo-labelling approach, whereby a classifier trained on the source data is applied to the target data in order to provide estimates of the labels \hat{Y}_t . In addition, the posterior probabilities $p(Y | X)$ are complicated to obtain, meaning that JDA utilises the class conditional distributions $p(X_s | Y_s)$ and $p(X_t | Y_t)$. By using the true source labels and pseudo target labels JDA matches the conditional distribution for each class $p(X_s | Y_s = c)$ and $p(X_t | Y_t = c)$ where $c \in \{1, \dots, C\}$ in the label set \mathcal{Y} .

The MMD between these class conditional distributions (using the empirical kernel embedding) can be formed as in,

$$\text{Dist}(p(\tilde{X}_s), p(\tilde{X}_t)) + \text{Dist}(p(Y_s | \tilde{X}_s), p(Y_t | \tilde{X}_t)) \approx \text{tr}(W^\top K M_c K W) \quad (6)$$

It is noted that if $c = 0$ then this formulation becomes TCA and therefore if $c \in \{0, 1, \dots, C\}$ both the marginal and class conditionals distances (and hence an approximation of the joint) are minimised. The MMD matrix now becomes,

$$M_c(i, j) = \begin{cases} \frac{1}{N_s^{(c)} N_s^{(c)}}, & x_i, x_j \in \mathcal{D}_s^{(c)} \\ \frac{1}{N_t^{(c)} N_t^{(c)}}, & x_i, x_j \in \mathcal{D}_t^{(c)} \\ \frac{-1}{N_s^{(c)} N_t^{(c)}}, & \begin{cases} x_i \in \mathcal{D}_s^{(c)} \\ x_j \in \mathcal{D}_t^{(c)} \end{cases} \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

where $\mathcal{D}_s^{(c)} = \{\mathbf{x}_i : \mathbf{x}_i \in \mathcal{D}_s \wedge y(\mathbf{x}_i) = c\}$ are the instances that belong in class c given the true source label $y(\mathbf{x}_i)$ of \mathbf{x}_i and $N_s^{(c)} = |\mathcal{D}_s^{(c)}|$; and $\mathcal{D}_t^{(c)} = \{\mathbf{x}_i : \mathbf{x}_i \in \mathcal{D}_t \wedge \hat{y}(\mathbf{x}_i) = c\}$ are the instances that belong in class c given the pseudo target label $\hat{y}(\mathbf{x}_i)$ of \mathbf{x}_i and $N_t^{(c)} = |\mathcal{D}_t^{(c)}|$. Following the same formulation as TCA the optimisation problem (subject to the regularisation constraint and kernel PCA) again becomes an eigenvalue problem where the optimal W is obtained from the eigenvectors corresponding to the k smallest eigenvalues in,

$$\left(K \sum_{c=0}^C M_c K + \mu \mathbb{I} \right) W = K H K W \phi \quad (8)$$

Due to the pseudo-labelling of the target features problem, [6] recommends running several iterations of the optimisation to find the optimal W . Again the k -dimensional transformed feature space is calculated by $Z = KW \in \mathbb{R}^{(N_s+N_t) \times k}$, and a classifier trained on the transformed source data can be applied to the transformed target data.

NUMERICAL CASE STUDY

A numerical case study is presented in order to demonstrate each of the two domain adaptation algorithms in the context of SHM. The case study involves two homogeneous structures, specifically three degree-of-freedom shear structures as presented in Figure 1.

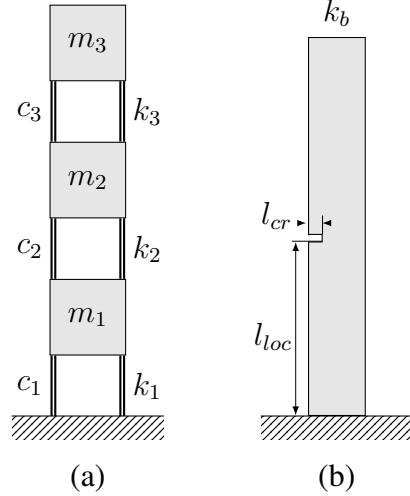


Figure 1: Schematic of the three degree-of-freedom shear structure: panel (a) is of the full system and panel (b) depicts the cantilever beam component where $\{k_i\}_{i=1}^3 = 4k_b$.

The SHM scenario is a two-class detection problem, i.e. for each system a single undamaged and damaged class exist. It is assumed that labels are known for the source structure and unknown for the target structure. Domain adaptation is implemented in order to build a classifier using the labelled information in the source domain that holds for the target domain. In order to demonstrate the effectiveness of domain adaptation, a k-Nearest Neighbour (k-NN) classifier is used; as if the learnt mapping is accurate, the source and target distributions should lie on top of each other and therefore be close in Euclidean distance.

The two shear structures are modelled as lumped-mass systems. The three masses $\{m_i\}_{i=1}^3$ are calculated from a given rectangular volume $v = l_m w_m t_m$ where the density ρ at each mass is considered different and l_m , w_m , t_m are the length, width and thickness of the volume. Similarly, the three stiffness coefficients $\{k_i\}_{i=1}^3$ are composed from the tip stiffness of four cantilever beams in parallel i.e. $4k_b = 4(3EI/l_b^3)$, where E is the elastic modulus, I the second moment of area and l the beam length. Each stiffness coefficient is constructed from a different E . Additionally damping coefficients $\{c_i\}_{i=1}^3$ are given for each structure but are not derived from a physical model.

In the damage scenario, an open crack of length l_{cr} is introduced to one of the four beams between the fixed support and the first mass; this results in $k_1 = 3k_b + k_d$, where k_d is the tip stiffness of a cantilever beam with a crack length l_{cr} at location l_{loc} along the length of the beam. The stiffness reduction due to an open crack in a cantilever beam is modelled as proposed by Christides and Barr in [7]. This model assumes a function of elastic modulus and second moment of area across the length of the beam x given by,

$$EI(x) = \frac{EI_0}{1 + C \exp(-2\alpha|x - l_{loc}|/t_b)} \quad (9)$$

where I_0 the second moment of the undamaged beam, t_b the thickness of the beam and α a coefficient experimental defined by Christides and Barr as 0.667. The constant $C = (I_0 - I_c)/I_c$ is a function of the undamaged I_0 and damaged second moments of area I_c , which for a rectangular beam are $I_0 = (w_b t_b^3)/12$ and $I_c = w_b(t_b - l_{cr})^3/12$. The damaged tip stiffness k_d is obtained from $k_d = -F/y_{tip}$ where F is a given force and

TABLE I: PROPERTIES OF THE SOURCE AND TARGET STRUCTURES.

Property	Unit	Source	Target
Beam geometry, $\{l_b, w_b, t_b\}$	mm	$\{160, 25, 6\}$	$\{180, 40, 5\}$
Mass geometry, $\{l_m, w_m, t_m\}$	mm	$\{300, 250, 25\}$	$\{280, 230, 40\}$
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	$\{4.0, 80\}$	$\{3.8, 60\}$
Elastic modulus, E	GPa	$\mathcal{N}(71, 10)$	$\mathcal{N}(210, 10)$
Density, ρ	kg/m ³	$\mathcal{N}(2700, 500)$	$\mathcal{N}(7800, 1000)$
Damping coefficient, c	Ns/m	$\mathcal{G}(50, 0.1)$	$\mathcal{G}(8, 0.8)$

y_{tip} is the tip deflection from numerically integrating the Euler-Bernoulli bending beam equation in,

$$\frac{\partial^2 y}{\partial x^2} = -\frac{M(x)}{EI(x)} \quad (10)$$

The properties of the source and target structures are displayed in Table I. The source and target structures are aluminium and steel respectively with similar geometry proportions. The crack properties are set such that the location and extent of damage are different in terms of percentage crack length and relative location between the source and target domains. The elastic modulus, density and damping coefficients are set as probability distributions such that $\{m_i, c_i, k_i\}_{i=1}^3$ are obtained from a random draw from these distributions.

This numerical example uses the three bending natural frequencies and damping ratios as features for the classification problem i.e. $X_s \in \mathbb{R}^{N_s \times 6}$ and $X_t \in \mathbb{R}^{N_t \times 6}$; these are shown in Figure 2. 250 repeats of the two structures were obtained i.e. $N_s = N_t = 250$, where each repeat was calculated from a random draw from the material properties and damping coefficients. Figure 2 demonstrates the difference between the two domains, not only in the absolute values of natural frequencies and damping ratios, but also in the size and scaling of the undamaged and damaged distributions. In addition the source domain can be seen to have clearer separation between the two classes than the target domain.

To demonstrate and motivate the need for domain adaptation in this example, a k-NN classifier (using Euclidean distance and $k = 1$) was trained on the labelled source domain data and tested on the unlabelled target domain data (where the features were normalised). As is obvious from Figure 2, the classifier fails to correctly label the target domain data, reflected in a 50% accuracy.

Domain Adaptation

The two domain adaptation methods were applied to the numerical data. In order to determine the number of components and regularisation parameters, k -fold cross-validation was implemented with $k = 10$. In addition, a linear kernel was utilised for the two approaches.

Cross-validation identified five components for TCA with the results being indifferent to μ (the results displayed here are for $\mu = 0.001$). Figure 3 presents a visualisation of the learnt mapping, where it can be seen that the source and target domains have been mapped onto each other. Although TCA only minimises the marginal distribution in the transformed space, Figure 3 shows the class conditions are well maintained, suggesting

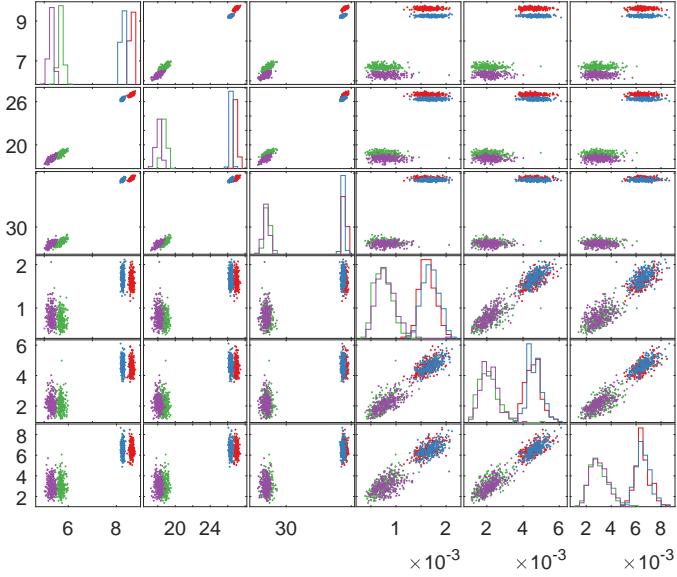


Figure 2: The source domain undamaged state (red) and damaged state (blue) compared to the target domain undamaged state (green) and damaged state (purple). Each quadrant is a two-dimensional comparison between feature combinations; with axes being the three natural frequencies followed by the damping ratios i.e. the bottom left quadrant is the first natural frequency against the third damping ratio.

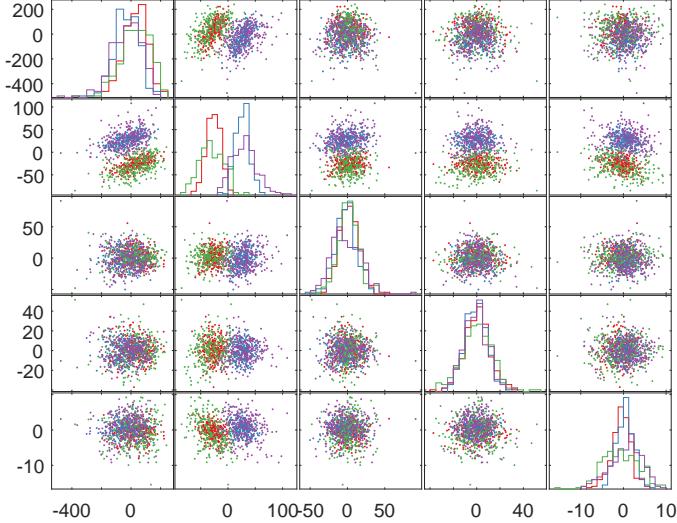


Figure 3: TCA components: the source domain undamaged state (red) and damaged state (blue) compared to the target domain undamaged state (green) and damaged state (purple). Each quadrant is a two-dimensional comparison between component combinations.

that the assumption $p(Y_s | \phi(X_s)) = p(Y_t | \phi(X_t))$ holds.

JDA was implemented using 10 iterations and a k-NN classifier for determining the pseudo labels (in order to maintain consistency with the other approaches). Three components and $\mu = 1 \times 10^{-5}$ were identified from cross-validation as presented in Figure 4. It can be seen that, as with TCA, the source and target domains have been mapped onto each other.

Finally, classification results for the target domain, when trained using the source domain, are presented for the two-class problem in Table II. Here two performance metrics are used, namely Accuracy and F1 score. The first, Accuracy is defined as,

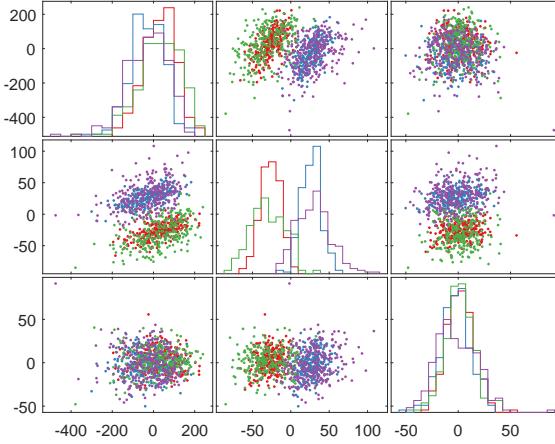


Figure 4: JDA components: the source domain undamaged state (red) and damaged state (blue) compared to the target domain undamaged state (green) and damaged state (purple). Each quadrant is a two-dimensional comparison between component combinations.

TABLE II: CLASSIFICATION RESULTS — NUMERICAL CASE STUDY.

Method	k-NN	TCA	JDA
Accuracy	50.0%	96.4%	96.0%
F1 score	0.667	0.965	0.960

$$\text{Accuracy} = \frac{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_{test} \vee f(\mathbf{x}) = y(\mathbf{x})|}{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_{test}|}. \quad (11)$$

The second, the F1 score, is formed as in,

$$F_1 = \frac{2PR}{P + R} \quad (12)$$

where P and R are the Precision and Recall, which are defined in terms of true positives (TP), false positives (FP) and false negatives (FN),

$$P = \frac{TP}{TP + FP} \quad (13)$$

$$R = \frac{TP}{TP + FN} \quad (14)$$

The case study demonstrates the benefits of applying domain adaptation with the two approaches outperforming a k-NN classifier trained on the untransformed data. For this case study, TCA outperforms JDA; this is likely due to the pseudo-label generation in JDA making it prone to overfitting and therefore cross-validation identifying fewer components than TCA; meaning there is less information to separate out the class conditional distributions.

EXPERIMENTAL CASE STUDY

A second case study is presented whereby the source domain is a numerical simulation and the target domain experimental data. The scenario involves two homogeneous

structures, three degree-of-freedom shear structures where the experimental structure is shown in Figure 5. As with the numerical case study, the SHM problem is two-class detection, and a k-NN classifier is utilised for the same reason. This case study provides a demonstration of the potential for numerical simulations to provide damage state labels for operational structures.



Figure 5: Experimental setup — three storey shear structure.

TABLE III: PROPERTIES OF THE SOURCE STRUCTURE.

Property	Unit	Source
Beam geometry, $\{l_b, w_b, t_b\}$	mm	{177.8, 25.4, 6.3}
Mass geometry, $\{l_m, w_m, t_m\}$	mm	{304.8, 254.0, 25.4}
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	{17.5, 88.9}
Elastic modulus, E	GPa	$\mathcal{N}(71, 1 \times 10^{-9})$
Density, ρ	kg/m ³	$\mathcal{N}(2700, 50)$
Damping coefficient, c	Ns/m	$\mathcal{G}(9, 0.5)$

The numerical simulation for the source domain is generated in the same manner as the previous case study i.e. a lumped-mass model where the damage is introduced by an open crack affecting the beam tip stiffness via Equation (9). A key difference in this case study is that the open crack is introduced through the width of the beam (rather than the thickness) meaning that $I_c = (w_b - l_{cr})t_b^3/12$. The properties of the numerical simulation are shown in Table III, where the structure and crack geometries are defined to be equivalent to the experimental structure and the material properties are set as typical aluminium properties, as the experimental structure is constructed from aluminium 6082. Clearly this analytical model is an oversimplification of the experimental structure given that it only includes the stiffness due to bending excluding the shear stiffness and excludes full geometries and joints etc. 100 repeats were obtained from the numerical simulation.

The target domain data were collected via modal testing, where the set up is shown in Figure 5. The structure was excited with broadband white noise via an electrodynamic shaker and the acceleration response measured at each of the three floors such that the first three bending natural frequencies were obtained. The structure was tested in the undamaged condition before a saw cut of 17.5mm was introduced to the front right beam in Figure 5, located between the ground and first floor. Five repeats were obtained for each of the two damage classes.

Figure 6 displays the feature sets for the source and target domain. It can be seen that the natural frequencies are underestimated by the numerical model by a factor of two due to the oversimplification; this presents a more challenging problem for the domain adaptation techniques, and also demonstrates the potential of this approach in utilising models that have not been validated and may contain model form errors, but capture to some extent the changes in the features due to damage. A k-NN classifier applied to these (normalised) features produces an accuracy of 50% when trained on the source data and applied to the target domain.

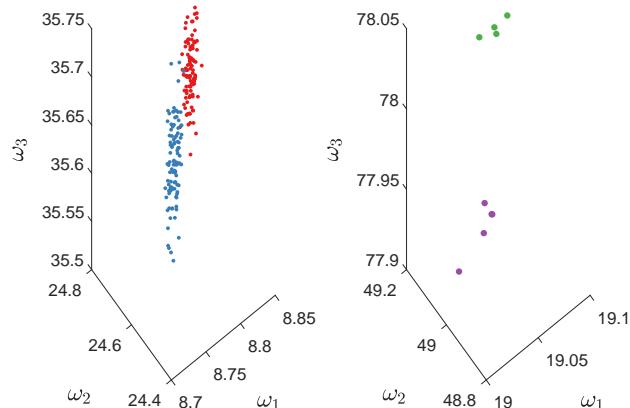


Figure 6: The source domain undamaged state (red) and damaged state (blue) (left) compared to the target domain undamaged state (green) and damaged state (purple) (right).

Domain Adaptation

Application of the two domain adaptation techniques to the second case study is presented. A linear kernel was implemented again with a k-NN classifier (utilised for pseudo-labelling in JDA). In this case study, cross-validation is not pursued due to the small number of experimental data points and the small feature dimension.

Both TCA and JDA optimise to the same weights, showing that for this case study $p(Y_s | \phi(X_s)) = p(Y_t | \phi(X_t))$ (given two components, $\mu = 0.1$ and 15 iteration for JDA). Figure 7 presents the transformed components from both TCA and JDA, where it can be seen that the two methods have successfully transformed the feature space such that all but one data point are classified correctly using a k-NN classifier. The classification accuracy and F1 scores reflect, in Table IV, the success of the approach; 90% and 0.909 respectively for both TCA and JDA (a single misclassified point).

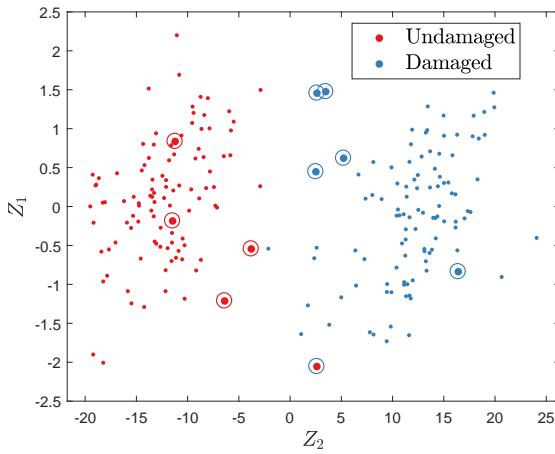


Figure 7: TCA and JDA components: true source (.), true target (●) and predicted target labels (○).

TABLE IV: CLASSIFICATION RESULTS
— EXPERIMENTAL CASE STUDY.

Method	k-NN
Accuracy	50.0%
F1 score	0.667

Method	TCA	JDA
Accuracy	90.0%	90.0%
F1 score	0.909	0.909

CONCLUSIONS

Two domain adaptation methods, TCA and JDA, have been demonstrated to be ef-

fective in performing population-based SHM across two homogeneous structures in a two-class classification problem, where labelled data is known for one structure and unknown for the other. These results have been demonstrated in both a numerical and experimental case study, with F1 scores greater than 0.9 for both studies compared to 0.67 when domain adaptation is not applied. These methods therefore provide a solution to the problem of the lack of available data.

Further research is required into applying transfer learning to heterogeneous structures where the feature space and label space may be the same i.e. the first five natural frequencies of a structure labelled by five crack locations, but the structures are very different i.e. a bridge and an aeroplane. A key theme will be in determining the similarities and distances between two structures in order to assess whether transfer learning is possible.

Acknowledgements

The authors would like to acknowledge the support of the UK Engineering and Physical Sciences Research Council via grants EP/R006768/1 and EP/R003645/1.

REFERENCES

1. Farrar, C. R. and K. Worden. 2012. *Structural Health Monitoring: A Machine Learning Perspective*, ISBN 9781119994336, doi:10.1002/9781118443118.
2. Papatheou, E., N. Dervilis, E. A. Maguire, I. Antoniadou, and K. Worden. 2015. “Population-based SHM: a case study on an offshore wind farm,” in *Proceedings of 10th International Workshop on Structural Health Monitoring*.
3. Pan, S. J. and Q. Yang. 2010. “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, ISSN 10414347, doi:10.1109/TKDE.2009.191.
4. Pan, S. J., I. W. Tsang, J. T. Kwok, and Q. Yang. 2011. “Domain Adaptation via Transfer Component Analysis,” *IEEE Transactions on Neural Networks*, 22(2):199–210, ISSN 1045-9227, doi: 10.1109/TNN.2010.2091281.
5. Schölkopf, B., A. Smola, and K. R. Müller. 1998. “Nonlinear Component Analysis as a Kernel Eigenvalue Problem,” *Neural Computation*, ISSN 08997667, doi:10.1162/089976698300017467.
6. Long, M., J. Wang, G. Ding, J. Sun, and P. S. Yu. 2013. “Transfer Feature Learning with Joint Distribution Adaptation,” in *2013 IEEE International Conference on Computer Vision*, IEEE, ISBN 978-1-4799-2840-8, pp. 2200–2207, doi:10.1109/ICCV.2013.274.
7. Christides, S. and A. D. Barr. 1984. “One-dimensional theory of cracked Bernoulli-Euler beams,” *International Journal of Mechanical Sciences*, ISSN 00207403, doi:10.1016/0020-7403(84)90017-1.