Proceedings Paper:
Gardner, P.A., Bull, L.A., Dervilis, N. et al. (2022) On the application of heterogeneous transfer learning to population-based structural health monitoring. In: Madarshahian, R. and Hemez, F., (eds.) Data Science in Engineering, Volume 9 : Proceedings of the 39th IMAC, A Conference and Exposition on Structural Dynamics 2021. 39th IMAC - A Conference and Exposition on Structural Dynamics 2021, 08-11 Feb 2021, Virtual conference. Conference Proceedings of the Society for Experimental Mechanics (CPSEMS). Springer International Publishing, pp. 87-98. ISBN: 9783030760038. ISSN: 2191-5644. EISSN: 2191-5652.

https://doi.org/10.1007/978-3-030-76004-5_11

On the application of heterogeneous transfer learning to population-based structural health monitoring

P. Gardner, L.A. Bull, N. Dervilis, & K. Worden
Dynamics Research Group
Department of Mechanical Engineering, University of Sheffield,
Mappin Street, Sheffield S1 3JD, UK

Abstract

Population-based structural health monitoring (PBSHM) is a branch of structural health monitoring (SHM) which seeks to leverage information from across a population of structures, with the aim of making robust data-based models that generalise across the population, allowing information to be exchanged and harnessed in constructing better inferences than considering an individual structure alone. PBSHM approaches overcome many of the challenges associated with conventional data-based SHM, such as limited labelled observations in training, classifiers failing to generalise when structural modifications or environmental variations occur etc. Transfer learning provides an important set of tools in performing PBSHM, with the technologies offering mechanisms for transferring label information between structures, and the ability to harness all the available information from all structures in the population, creating a single classification model that generalises across the complete population. This paper explores *heterogeneous transfer learning*, a branch of transfer learning where datasets have inconsistent feature spaces, i.e. the dimensions of datasets from one structure are different to those from another. In PBSHM, this scenario arises for several reasons; for example, the data acquisition processes on each structure may be different: e.g. the sample rates and durations were different for each structure, leading to transmissibilities with a different number of spectral lines. The paper compares two heterogeneous transfer learning approaches that are formed in a discriminative manner, namely kernelised Bayesian transfer learning and heterogeneous feature augmentation. The techniques are benchmarked against conventional approaches to data-based SHM, with the benefits of a heterogeneous transfer learning approach highlighted by a case study on a Gnat aircraft wing.

Keywords: Heterogeneous transfer learning; kernelised Bayesian transfer learning; heterogeneous feature augmentation; population-based structural health monitoring

1 Introduction

Data-based approaches to Structural Health Monitoring (SHM) have been demonstrated as successful in many scenarios [1, 2, 3, 4]. However, there are two main problems with data-based methods in general. Firstly, their construction conventionally assumes that the data distributions in training will be the same as in testing. This assumption is invalidated if changes to the system take place, such as structural repairs, which fundamentally change the way the system responds. Secondly, to perform diagnosis beyond anomaly detection, some labelled data are required, and often a set of labels that correspond to all the damage states of interest is unavailable in training due to economic and feasibility issues. By considering a population of structures, the number and variety of labelled data points can be increased, aiding inferences across the complete population, even when individuals have sparsely-labelled dataset. This approach is termed *Population-Based* Structural Health Monitoring (PBSHM) [7, 8, 9]. Transfer learning, a branch of machine learning, can be used to overcome both of these challenges, harmonising datasets from different structures even when data shift occurs, and allowing label datasets to be transferred between different members of a population. This paper seeks to compare transfer learning approaches for PBSHM, particularly in scenarios where the feature data from each member of the population are inconsistent and have different dimensions. For example, the data acquisition processes on each structure may be different: e.g. the sample rates and durations were different for each structure leading to transmissibilities with a different number of spectral lines. Transfer learning methods that handle inconsistent feature spaces are named *heterogeneous transfer learning* methods.

Although transfer learning methods have been implemented for health monitoring scenarios [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31], all of these approaches assume consistent feature spaces. A large number of these methods use a fine-tuning approach [17, 18, 19, 20, 21, 22, 23, 24], re-purposing pre-trained neural networks for new tasks, with the remainder using a form of domain adaptation [25, 26, 27, 28, 29, 30, 31], where the aim is to adapt the feature spaces so that information is transferred between datasets. However, as stated, none of these approaches handle inconsistent feature spaces. This paper therefore utilises two existing heterogeneous transfer learning algorithms, which learn a mapping to a shared latent space where a discriminative classifier is inferred, namely Kernelised Bayesian Transfer Learning (KBTL) [10] and Heterogeneous Feature Augmentation (HFA) [14, 15]. These approaches expand the datasets and population types that can be considered in PBSHM, making the approach more applicable.

The outline of the paper is as follows. Heterogeneous transfer learning is defined and two algorithms KBTL [10] and HFA [14, 15] are presented. A comparison of these approaches, benchmarked against conventional techniques, is shown for an SHM localisation problem on a dataset from a Gnat trainer aircraft. Finally conclusions are presented.

2 Heterogeneous transfer learning

Transfer learning is a category of machine learning that seeks to utilise information from various sources in improving a machine learning model. Formally, the definition requires two objects:

> A **domain** $\mathcal{D} = \{\mathcal{X}, p(X)\}$ consisting of a feature space $\mathcal{X}$ and a marginal probability distribution $p(X)$ over a finite sample of feature data $X = \{\boldsymbol{x}_i\}_{i=1}^{N} \in \mathcal{X}$ from $\mathcal{X}$.
>
> A **task** $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$ combing a label space $\mathcal{Y}$ and a predictive function $f : \mathcal{X} \to \mathcal{Y}$.

From these objects, transfer learning is the process of improving a target predictive function, for a target task, given knowledge from some a set of source domains and tasks [11, 12, 13]. Transfer learning can then be subdivided into homogeneous and heterogeneous transfer learning methods. The former category encompasses algorithms that require the feature spaces between source and target domains to be consistent and of the same dimensions, i.e. $\mathcal{X}_s = \mathcal{X}_t$ and $d_s = d_t$ where the subscripts $_s$ and $_t$ denote the source and target domains respectively for a single source and target domain. Heterogeneous transfer learning methods relax these restrictions, handling inconsistent feature spaces where each domain may have a different dimension, i.e. $\mathcal{X}_s \neq \mathcal{X}_t$ and $d_s \neq d_t$. In an engineering context, this assumption means that multiple different types of data can be used to transfer knowledge between structures. For example, imagine that a labelled source structure has a dataset consisting of ten natural frequencies ($d_s = 10$) and a target structure where the dataset are transmissibilities with 512 spectral lines ($d_t = 512$), these feature spaces are inconsistent and have different dimensions, meaning a heterogeneous transfer learning approach is required. Clearly, heterogeneous transfer learning methods are more general, but also more challenging approaches, with these algorithms needing to learn a dimensionality transformation as well as a mapping between datasets from source and target domains.

This paper compares two different heterogeneous transfer learning methods, namely kernelised Bayesian transfer learning and heterogeneous feature augmentation. Both techniques are constructed around a discriminative classifier, with the methods jointly inferring a mapping from the feature spaces onto a shared latent space where the classifier is applied. Details of the two algorithms are given in the following sections.

2.1 Kernelised Bayesian transfer learning

KBTL is a Bayesian discriminative classification model, proposed by Gönen and Margolin [10], that utilises T domains $\{\mathcal{D}_t\}_{t=1}^{T}$ with inconsistent feature spaces $\{\mathcal{X}_t\}_{t=1}^{T}$ (where there is no weighting towards a particular domain as a target domain). The method can be seen as performing two steps in a joint manner; firstly, a mapping is learnt that projects the data for each domain through a kernel and then onto a shared latent space via a linear mapping. The second step constructs an optimal discriminative classifier (in a one-vs-all sense) in the shared latent space that classifies all domains. More formally, the data from each domain $X_t = \{\boldsymbol{x}_{t,i} \in \mathcal{X}_t\}_{i=1}^{N_t} \in \mathbb{R}^{N_t \times d_t}$ (where N_t are the number of data points and d_t is the dimension of the feature set) are projected into a Reproducing Kernel Hilbert Space (RKHS) via a kernel $K_t = k_t(X_t, X_t) \in \mathbb{R}^{N_t \times N_t}\ \forall\ t \in 1\!:\!T$, before being mapped onto the shared latent space as $H_t = A_t^{\mathsf{T}} K_t \in \mathbb{R}^{R \times N_t},\ \forall\ t \in 1\!:\!T$. A joint discriminative classifier is subsequently inferred as $\boldsymbol{f}_{t,l} = H_t^{\mathsf{T}} \boldsymbol{w}_l + \mathbf{1} b_l\ \forall\ t \in 1\!:\!T$ and $\forall\ l \in 1\!:\!L$, where the parameters of the classifier $\{b_l \in \mathbb{R}^{1\times 1},\ \boldsymbol{w}_l \in \mathbb{R}^{R\times 1}\}$ are shared for all tasks. The function output can then be used to determine the label for each data point for each domain. It is

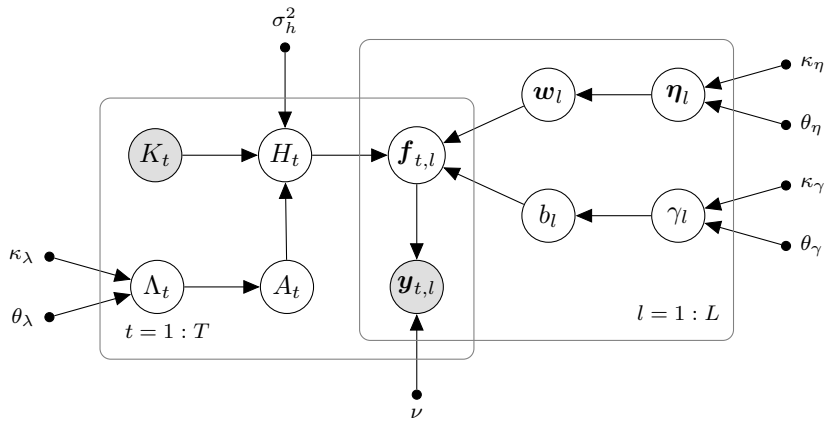


Figure 1: Multi-class classification: graphical model of KBTL. Recreated from [10].

noted that the classifier is formed in a similar manner to a Relevance Vector Machine (RVM) [35]. A graphical model of KBTL is presented in Figure 1, stating the conditional relationships between the observation parameters $\{K_t, Y_t\}_{t=1}^T$ (where $D = \{Y_t\}_{t=1}^T$), priors $\Xi = \{\{\Lambda_t\}_{t=1}^T, \{\boldsymbol{\eta}_l, \gamma_l\}_{l=1}^L\}$, hyperparameters[1] $\boldsymbol{\zeta} = \{\kappa_\lambda, \theta_\lambda, \kappa_\eta, \theta_\eta, \kappa_\gamma, \theta_\gamma, \sigma_h^2, \nu\}$, latent variables and model parameters $\Theta = \{\{b, \boldsymbol{w}\}_{l=1}^L, \{\{\boldsymbol{f}_{t,l}\}_{l=1}^L, A_t, H_t\}_{t=1}^T\}$.

The graphical model leads to the following modelling definitions for the projection part:

$$\Lambda_t[i,s] \sim \mathcal{G}(\Lambda_t[i,s] \mid \kappa_\lambda, \theta_\lambda) \tag{1a}$$

$$A_t[i,s] \mid \Lambda_t[i,s] \sim \mathcal{N}(A_t[i,s] \mid 0, (\Lambda_t[i,s])^{-1}) \tag{1b}$$

$$H_t[s,i] \mid A_t[:,s], K_t[:,i] \sim \mathcal{N}(H_t[s,i] \mid A_t[:,s]^\mathsf{T} K_t[:,i], \sigma_h^2) \tag{1c}$$

$\forall\, i \in 1\!:\!N_t$, $\forall\, s \in 1\!:\!R$, $\forall\, t \in 1\!:\!T$, where square brackets denote indices — two elements denote rows and columns of a matrix respectively, one element denotes a vector index, : denotes the set of index values. $\mathcal{G}(\cdot \mid \kappa, \theta)$ refers to a gamma distribution parametrised by shape κ and scale θ parameters, and $\mathcal{N}(\cdot \mid \boldsymbol{\mu}, \Sigma)$ refers to a Gaussian distribution parametrised by mean $\boldsymbol{\mu}$ and covariance Σ parameters.

Again, modelling assumptions for the classification part are:

$$\gamma_l \sim \mathcal{G}(\gamma_l \,|\, \kappa_\gamma, \theta_\gamma) \tag{2a}$$

$$b_l \mid \gamma_l \sim \mathcal{N}(b \mid 0, \gamma_l^{-1}) \tag{2b}$$

$$\boldsymbol{\eta}_l[s] \sim \mathcal{G}(\boldsymbol{\eta}_l[s]\,;\, \kappa_\eta, \theta_\eta) \tag{2c}$$

$$\boldsymbol{w}_l[s] \mid \boldsymbol{\eta}_l[s] \sim \mathcal{N}(\boldsymbol{w}_l[s] \mid 0, (\boldsymbol{\eta}_l[s])^{-1}) \tag{2d}$$

$$\boldsymbol{f}_{t,l}[i] \mid b_l, \boldsymbol{w}_l, H_t[:,i] \sim \mathcal{N}(\boldsymbol{f}_{t,l}[i] \mid \boldsymbol{w}_l^\mathsf{T} H_t[:,i] + b_l, 1) \tag{2e}$$

$$\boldsymbol{y}_{t,l}[i] \mid \boldsymbol{f}_{t,l}[i] \sim \delta(\boldsymbol{f}_{t,l}[i]\boldsymbol{y}_{t,l}[i] > \nu) \tag{2f}$$

$\forall\, i \in 1\!:\!N_t$, $\forall\, s \in 1\!:\!R$, $\forall\, l \in 1\!:\!L$, $\forall\, t \in 1\!:\!T$, where $\delta(\cdot)$ is a Dirac delta function. The model includes a non-negative margin parameter ν which creates a low-density region between the l^{th} class and the rest of the data (in a one-vs-all sense); this works in a similar manner to a margin in an Support Vector Machine (SVM).

In order to efficiently infer the model parameters variational inference is used (via a mean-field approach)[2],

$$p(\{\Theta, \Xi\} \mid D) \approx q(\{\Theta, \Xi\}) = \prod_{t=1}^{T} [q(\Lambda_t) q(A_t) q(H_t)] \prod_{l=1}^{L} [q(\gamma_l) q(\boldsymbol{\eta}_l) q(b_l, \boldsymbol{w}_l)] \prod_{t=1}^{T} \prod_{l=1}^{L} q(\boldsymbol{f}_{t,l}). \tag{3}$$

This process leads to the following closed-form update equations ($\langle f(\cdot) \rangle$ denotes the posterior expectation $\mathbb{E}_{q(\cdot)}[f(\cdot)]$) for the dimensionality reduction part:

$$\kappa(\Lambda_t[i,s]) = \kappa_\lambda + 1/2 \tag{4a}$$

[1] For the sake of clarity, dependencies on any of the hyperparameters $\boldsymbol{\zeta}$ are dropped from the notation in this paper.

[2] For specific details the reader is referred to [10].

$$\theta(\Lambda_t[i,s]) = (1/\theta_\lambda + \langle A_t[i,s]^2\rangle/2)^{-1} \tag{4b}$$

$$\Sigma(A_t[:,s]) = (\mathrm{diag}(\langle \Lambda_t[:,s]\rangle) + K_t K_t^\mathsf{T}/\sigma_h^2)^{-1} \tag{5a}$$
$$\mu(A_t[:,s]) = \Sigma(A_t[:,s])(K_t\langle H_t[s,:]^\mathsf{T}\rangle/\sigma_h^2) \tag{5b}$$

$$\Sigma(H_t[:,i]) = \left(\mathbb{I}/\sigma_h^2 + \sum_{l=1}^{L}\langle \boldsymbol{w}_l \boldsymbol{w}_l^\mathsf{T}\rangle\right)^{-1} \tag{6a}$$

$$\mu(H_t[:,i]) = \Sigma(H_t[:,i])(\langle A_t^\mathsf{T}\rangle K_t[:,i]/\sigma_h^2 + \sum_{l=1}^{L}\langle \boldsymbol{f}_{t,l}[i]\rangle\langle \boldsymbol{w}_l\rangle - \langle b_l \boldsymbol{w}_l\rangle) \tag{6b}$$

meaning that the shared latent space is updated to reflect the performance of *all* L discriminative classifiers. The update equations for the classification part are,

$$\kappa(\gamma_l) = \kappa_\gamma + 1/2 \tag{7a}$$
$$\theta(\gamma_l) = (1/\theta_\gamma + \langle b_l^2\rangle/2)^{-1} \tag{7b}$$
$$\kappa(\boldsymbol{\eta}_l[s]) = \kappa_\eta + 1/2 \tag{7c}$$
$$\theta(\boldsymbol{\eta}_l[s]) = (1/\theta_\eta + \langle \boldsymbol{w}_l[s]^2\rangle/2)^{-1} \tag{7d}$$

$$\Sigma(b_l, \boldsymbol{w}_l) = \begin{bmatrix} \langle\gamma_l\rangle + \sum_{t=1}^{T} N_t & \sum_{t=1}^{T}\mathbf{1}^\mathsf{T}\langle H_t^\mathsf{T}\rangle \\ \sum_{t=1}^{T}\langle H_t\rangle\mathbf{1} & \mathrm{diag}(\langle\boldsymbol{\eta}_l\rangle) + \sum_{t=1}^{T}\langle H_t H_t^\mathsf{T}\rangle \end{bmatrix}^{-1} \tag{8a}$$

$$\mu(b_l, \boldsymbol{w}_l) = \Sigma(b_l, \boldsymbol{w}_l)\begin{bmatrix} \sum_{t=1}^{T}\mathbf{1}^\mathsf{T}\langle \boldsymbol{f}_{t,l}\rangle \\ \sum_{t=1}^{T}\langle H_t\rangle\langle \boldsymbol{f}_{t,l}\rangle \end{bmatrix} \tag{8b}$$

$$\Sigma(\boldsymbol{f}_{t,l}[i]) = 1 \tag{9a}$$
$$\mu(\boldsymbol{f}_{t,l}[i]) = \langle \boldsymbol{w}_l^\mathsf{T}\rangle\langle H_t[:,i]\rangle + \langle b_l\rangle \tag{9b}$$
$$\mathrm{a}(\boldsymbol{f}_{t,l}[i]) = \begin{cases} -\infty, & \text{if } \boldsymbol{y}_{t,l}[i] = -1 \\ \nu, & \text{if } \boldsymbol{y}_{t,l}[i] = +1 \end{cases} \tag{9c}$$
$$\mathrm{b}(\boldsymbol{f}_{t,l}[i]) = \begin{cases} -\nu, & \text{if } \boldsymbol{y}_{t,l}[i] = -1 \\ \infty, & \text{if } \boldsymbol{y}_{t,l}[i] = +1 \end{cases} \tag{9d}$$

Finally, the predictive equations are found by substituting the true posterior distributions with the variational approximation. The predictive equation for the latent space for each domain $\boldsymbol{h}_{t,*}$ for a new data point $\boldsymbol{x}_{t,*}$ (via its corresponding kernel $k(X_t, \boldsymbol{x}_{t,*}) = \boldsymbol{k}_{t,*}$) is,

$$p(\boldsymbol{h}_{t,*} \mid \boldsymbol{k}_{t,*}, D) = \prod_{s=1}^{R}\mathcal{N}(\boldsymbol{h}_{t,*}[s] \mid \ \mu(A_t[:,s])^\mathsf{T}\boldsymbol{k}_{t,*}, \ \ \sigma_h^2 + \boldsymbol{k}_{t,*}^\mathsf{T}\Sigma(A_t[:,s])\boldsymbol{k}_{t,*}) \tag{10}$$

where the predictive equation for the classifier is,

$$p(f_{t,l,*} \mid \boldsymbol{h}_{t,*}, D) = \mathcal{N}(\boldsymbol{f}_{t,l,*} \mid \ \mu(b_l, \boldsymbol{w}_l)^\mathsf{T}[1 \quad \boldsymbol{h}_{t,*}]^\mathsf{T}, \ \ 1 + [1 \quad \boldsymbol{h}_{t,*}]\Sigma(b_l, \boldsymbol{w}_l)[1 \quad \boldsymbol{h}_{t,*}]^\mathsf{T}). \tag{11}$$

By passing the predictive function through a truncated Gaussian cumulative density function, the probability of an observation belonging to class l is predicted as,

$$p(y_{t,l,*} = +1 \mid f_{t,l,*}, D) = z_{t,l,*}^{-1}\Phi\left(\frac{\mu(f_{t,l,*}) - \nu)}{\Sigma(f_{t,l,*})}\right) \tag{12}$$

where $z_{t,l,*}$ is the normalisation coefficient for a truncated Gaussian distribution, where [a b] (the truncation interval) are for the $y = +1$ case. This equation produces a $1 \times L$ vector $\boldsymbol{y}_{t,*}$, where a *Maximum A Posteriori* (MAP) estimate of the class label can be obtained by by finding the element l corresponding to $\max(p(\boldsymbol{y}_{t,*} = +1 \mid \boldsymbol{f}_{t,*}, D))$.

2.2 Heterogeneous Feature Augmentation

Heterogeneous Feature Augmentation (HFA), first proposed by Duan *et al.*, seeks to learn a mapping from the data domains to an augmented (and shared) latent space in which a discriminative classifier, using a support vector machine (SVM) formulation, is inferred and used to jointly classify the data [14]. The approach is constructed assuming a single source and target domain, with respective feature data $X_s \in \mathbb{R}^{d_s \times N_s}$ and $X_t \in \mathbb{R}^{d_s \times N_s}$ (meaning $\mathcal{X}_s \neq \mathcal{X}_t$). The main idea in HFA is that feature replication can be used to augment the original feature space and allow transfer between inconsistent feature spaces, replicating the source and target domains to form a larger shared space. Additionally, in order to adapt the original source and target feature spaces such that knowledge can be transferred, part of the shared augmented space consists of a common subspace ($\mathbb{R}^{d_c}$), where the projection matrices P and Q are used to map source and target samples $\boldsymbol{x}_s$ and $\boldsymbol{x}_t$ onto this common subspace. The mappings that transform the source and target datasets onto the shared augmented space are defined as $\psi_s(\boldsymbol{x}_s) = \begin{bmatrix} P\phi_s(\boldsymbol{x}_s)^\mathsf{T} & \phi_s(\boldsymbol{x}_s)^\mathsf{T} & \mathbf{0}_{d_t}^\mathsf{T} \end{bmatrix}^\mathsf{T}$ and $\psi_t(\boldsymbol{x}_t) = \begin{bmatrix} Q\phi(\boldsymbol{x}_t)^\mathsf{T} & \mathbf{0}_{d_s}^\mathsf{T} & \phi_t(\boldsymbol{x}_t)^\mathsf{T} \end{bmatrix}^\mathsf{T}$ respectively, where $\phi_s(\cdot)$ and $\phi_t(\cdot)$ are some nonlinear mapping such that the kernels on the source and target domain are defined as $K_s = \Phi_s^\mathsf{T}\Phi_s$ and $K_t = \Phi_t^\mathsf{T}\Phi_t$. These augmented feature spaces are then adapted into an SVM formulation outlined below.

The weight vector for the SVM formulation is defined as $\boldsymbol{w} = [\boldsymbol{w}_c^\mathsf{T}\ \boldsymbol{w}_s^\mathsf{T}\ \boldsymbol{w}_t^\mathsf{T}]^\mathsf{T}$, combining weights for the common, source and target subspaces (which make up the augmented shared space). The structural risk for the SVM is therefore,

$$\min_{P,Q} \min_{\boldsymbol{w},b,\xi_s[i],\xi_t[i]} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\left(\sum_{i=1}^{N_s}\xi_s[i] + \sum_{i=1}^{N_t}\xi_t[i]\right) \tag{13a}$$

$$\text{s.t. } y_s[i](\boldsymbol{w}^\mathsf{T}\psi_s(\boldsymbol{x}_s[i]) + b) \geq 1 - \xi_s[i],\ \xi_s[i] \geq 0 \tag{13b}$$

$$y_t[i](\boldsymbol{w}^\mathsf{T}\psi_t(\boldsymbol{x}_t[i]) + b) \geq 1 - \xi_t[i],\ \xi_t[i] \geq 0 \tag{13c}$$

$$\|P\|_F^2 \leq \lambda_p,\ \ \|Q\|_F^2 \leq \lambda_q, \tag{13d}$$

where $C > 0$ trades off model complexity and empirical losses on both domains, with λ_p and λ_q controlling the complexity of P and Q. The dual form of the inner optimisation problem is formed by introducing the dual variables $\boldsymbol{\alpha}_s$ and $\boldsymbol{\alpha}_t$ for the constraints in equations (13b) and (13c). The Karush-Kuhn-Tucker (KKT) conditions are found by setting the derivatives of the Lagrangian of equation (13a) with respect to $\{\boldsymbol{w}, b, \xi_s[i], \xi_t[i]\}$ to zeros, leading to $\boldsymbol{w} = \sum_{i=1}^{N_s}\alpha_s[i]y_s[i]\psi_s(\boldsymbol{x}_s[i]) + \sum_{i=1}^{N_t}\alpha_t[i]y_t[i]\psi_t(\boldsymbol{x}_t[i])$, $\sum_{i=1}^{N_s}\alpha_s[i]y_s[i] + \sum_{i=1}^{N_t}\alpha_t[i]y_t[i] = 0$ and $0 \leq \alpha_s[i], \alpha_t[i] \leq C$. Furthermore, rather than explicitly solving for P and Q, the approach utilises a latent representation via $\tilde{H}$, a nonlinear transformation matrix that satisfies $H = \Phi K^{-1/2}\tilde{H}K^{-1/2}\Phi^\mathsf{T}$ where $H = [P\ Q]^\mathsf{T}[P\ Q]$ and,

$$K = \begin{bmatrix} K_s & O_{N_s \times N_t} \\ O_{N_t \times N_s} & K_t \end{bmatrix}$$

where the $O_{N\times M}$ notation defines an $N \times M$ matrix of zeros. The KKT conditions and the introduction of the nonlinear transformation metric lead to the following dual problem,

$$\min_{\tilde{H}\succeq 0} \max_{\boldsymbol{\alpha}} \mathbf{1}^\mathsf{T}\boldsymbol{\alpha} - \frac{1}{2}(\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T}K_{\tilde{H}}(\boldsymbol{\alpha}\circ\boldsymbol{y}) \tag{14a}$$

$$\text{s.t. } \boldsymbol{y}^\mathsf{T}\boldsymbol{\alpha} = 0,\ \ 0 \leq \boldsymbol{\alpha} \leq C\mathbf{1} \tag{14b}$$

$$\text{tr}\left(\tilde{H}\right) \leq \lambda \tag{14c}$$

where $K_{\tilde{H}} = \Phi^\mathsf{T}(H + \mathbb{I})\Phi = K^{1/2}(\tilde{H} + \mathbb{I})K^{1/2}$, $\lambda = \lambda_p + \lambda_q$, $\boldsymbol{\alpha} = \begin{bmatrix}\boldsymbol{\alpha}_s^\mathsf{T} & \boldsymbol{\alpha}_t^\mathsf{T}\end{bmatrix}^\mathsf{T}$ and $\boldsymbol{y} = \begin{bmatrix}\boldsymbol{y}_s^\mathsf{T} & \boldsymbol{y}_t^\mathsf{T}\end{bmatrix}^\mathsf{T} \in \{+1 - 1\}$.

In order to guarantee convergence to the global solution Li *et al.* then introduce a convex form of equation (14a) [15], replacing the trace regularisation term in equation (14c) with a regularisation term in the objective function leading to,

$$\min_{\tilde{H}\succeq 0} \max_{\boldsymbol{\alpha}} \mathbf{1}^\mathsf{T}\boldsymbol{\alpha} - \frac{1}{2}(\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T}K_{\tilde{H}}(\boldsymbol{\alpha}\circ\boldsymbol{y}) + \eta\text{tr}\left(\tilde{H}\right) \tag{15a}$$

$$\text{s.t. } \boldsymbol{y}^\mathsf{T}\boldsymbol{\alpha} = 0,\ \ 0 \leq \boldsymbol{\alpha} \leq C\mathbf{1} \tag{15b}$$

where η is a trade-off parameter. Subsequently, to avoid the non-trivial semi-definite programming problem, $\tilde{H}$ is decomposed into a linear combination of positive semi-definite (PSD) matrices, in this case rank-one normalised PSD matrices $\mathcal{M} = \{M_r|_{r=1}^{\infty}\}$, such that $\tilde{H} = H_\theta = \sum_{r=1}^{\infty} \theta_r M_r$ where $M_r = \boldsymbol{h}_r \boldsymbol{h}_r^\mathsf{T}$ and $\boldsymbol{h}_r^\mathsf{T} \boldsymbol{h}_r = 1$, with $\boldsymbol{\theta}$ being a vector of linear combination coefficients. By substituting in the linear combinations, the regularisation term $\mathrm{tr}\left(\tilde{H}\right) = \mathrm{tr}(\sum_{r=1}^{\infty} \theta_r M_r) = \mathbf{1}^\mathsf{T}\boldsymbol{\theta}$ and the optimisation problem can now be formed in terms of $\boldsymbol{\theta}$ instead of $\tilde{H}$ as,

$$\min_{\boldsymbol{\theta}\in\mathcal{D}_\theta} \max_{\boldsymbol{\alpha}\in\mathcal{A}} \mathbf{1}^\mathsf{T}\boldsymbol{\alpha} - \frac{1}{2}(\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T} \sum_{r=1}^{\infty} \theta_r K_r (\boldsymbol{\alpha}\circ\boldsymbol{y}) \tag{16}$$

where $K_r = K^{1/2}(\lambda M_r + \mathbb{I})K^{1/2}$, $\mathcal{D}_\theta = \{\boldsymbol{\theta}|\mathbf{1}^\mathsf{T}\boldsymbol{\theta} \leq 1, \boldsymbol{\theta} \geq 0\}$ and $\mathcal{A} = \{\boldsymbol{\alpha}|\boldsymbol{y}^\mathsf{T}\boldsymbol{\alpha} = 0, \mathbf{0} \leq \boldsymbol{\alpha} \leq C\mathbf{1}\}$, which is an infinite kernel learning problem with each base kernel as K_r (where [15] states the proof that optimising for $\boldsymbol{\theta}$ is the same as finding the optimal H_θ).

To simplify the problem in equation 16, $\tilde{H}$ can be represented by a finite and small number of base kernels, constructed using a cutting-plane algorithm. By forming a new dual problem where the dual variable τ is introduced for $\boldsymbol{\theta}$ the problem becomes,

$$\max_{\tau,\boldsymbol{\alpha}\in\mathcal{A}} \mathbf{1}^\mathsf{T}\boldsymbol{\alpha} - \tau \tag{17a}$$

$$\text{s.t. } \frac{1}{2}(\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T} K_r (\boldsymbol{\alpha}\circ\boldsymbol{y}) \leq \tau, \ \ \forall r \tag{17b}$$

meaning that there are now an infinite number of constraints. The approach then allows the structural risk to be approximated by iteratively adding a kernel where the corresponding constraint is violated according to the current solution; the kernel associated with this constraint is called an *active kernel*. The most active kernel is found by maximising the left hand side of equation (17b),

$$\max_{M\in\mathcal{M}} \frac{1}{2}(\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T} K_M (\boldsymbol{\alpha}\circ\boldsymbol{y}) \tag{18}$$

where $K_m = K^{1/2}(\lambda M + \mathbb{I})K^{1/2}$. This maximisation has a closed-form solution where $M = \boldsymbol{h}\boldsymbol{h}^\mathsf{T}$ and $\boldsymbol{h} = K^{1/2}(\boldsymbol{\alpha}\circ\boldsymbol{y})/\|K^{1/2}(\boldsymbol{\alpha}\circ\boldsymbol{y})\|$. The optimisation is therefore conducted by initialising the set of rank-one PSD matrices with $M_1 = \boldsymbol{h}_1\boldsymbol{h}_1^\mathsf{T}$ where $\boldsymbol{h}_1$ is a unit vector. Using this current $\mathcal{M}$, the min-max problem in equation (16) is solved to find the optimal $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$. These values are subsequently used to find the most active kernel using equation (18), and this is added to the current set of $\mathcal{M}$. The process is iterated, optimising for $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$ then adding the most active kernel to $\mathcal{M}$ until convergence.

Once converged, the predictive equations using the optimal $\boldsymbol{\alpha}$ and $\tilde{H}(=\lambda\sum_r \theta_r M_r)$ for a new test sample $\boldsymbol{x}$ are,

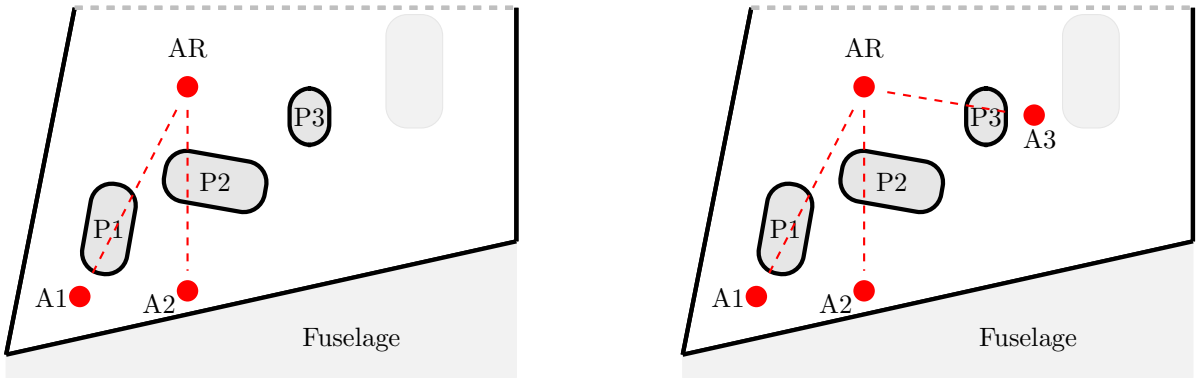
$$f(\boldsymbol{x}) = (\boldsymbol{\alpha}\circ\boldsymbol{y})^\mathsf{T} K^{1/2}(\tilde{H} + \mathbb{I}) \begin{bmatrix} 0_{N_s \times N_t} \\ K_t^{-1/2} \end{bmatrix} \boldsymbol{k}_t + b \tag{19}$$

where $\boldsymbol{k}_t = k(X_t, \boldsymbol{x})$.

3 Gnat aircraft dataset

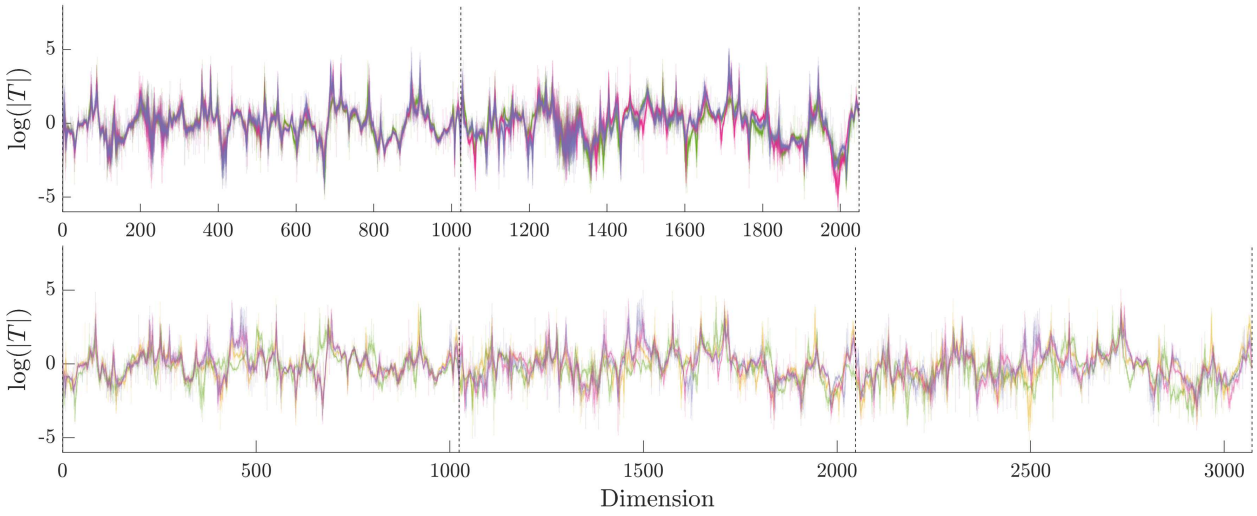
The Gnat aircraft dataset is an experimental dataset that has been well-studied within the field of SHM [36, 37, 38, 39, 40, 41]. Although the dataset is from one structure, namely an aircraft wing from a Gnat trainer aircraft [37, 38], the dataset does form two distinct data domains due to data shift that occurs from changes caused by reattaching inspection panels (even when the torque of the fasteners is monitored [38]). A scenario is imagined where the sensor configurations change between the two domains, meaning a different number of sensors were available in each domain. This change in available sensors, shown in Figure 2, creates feature spaces for each domain that are inconsistent (where transmissibility paths are used as the features) and therefore heterogeneous transfer learning is required to transfer information between the two domains.

The Gnat dataset consists of measurements from the starboard wing *in-situ*, undergoing a band-limited Gaussian noise input, where the response was measured at several locations using uni-axial accelerometers, forming transmissibility paths. Transmissibilities with 1024 spectral lines between 1024-2048Hz form the feature set. Damage was introduced via sequentially removing the inspection panels shown in Figure 2 (as it was not possible to damage the structure), i.e. no panels are removed, $Y = 0$, P1 is removed, $Y = 1$, P1 and P2 are removed, $Y = 2$, and P1, P2 and

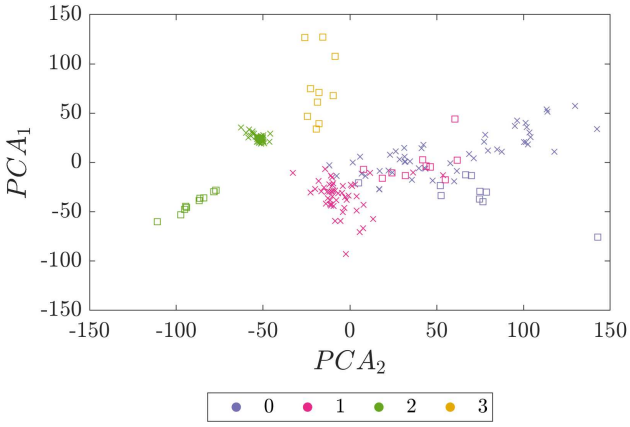


Domain One Domain Two

Figure 2: Schematics sensor configurations for Domain One and Domain Two for the Gnat aircraft wing (not to scale).



(a)



(b)

Figure 3: Visualisation of Domains One and Two training feature data. The Panel (a) displays the stacked log transmissibilities for Domain One (top) and Domain Two (bottom) where a vertical line indicates the start of a new transmissibility. The Panel (b) compares the first two principal components of Domain One (×) and Domain Two (□).

P3 are removed, $Y = 3$. For each class, 100 measurements were collected before all the panels were reattached (with fastener torque being controlled [38]) and the process repeated in order to collect a second data domain. For the sake of brevity etc., the interested reader is referred to [38] for more details on the dataset.

As stated, the dataset forms two data domains (both with 400 observations in total), where Domain One consists of two transmissibilities ($T1 = A1/AR$ and $T2 = A2/AR$), and three transmissibilities ($T1$ $T2$ and $T3 = A3/AR$) are available for Domain Two. This leads to two inconsistent feature spaces as $\mathcal{X}_1 = \{T1, T2\} \in \mathbb{R}^{N_1 \times 2048}$ and $\mathcal{X}_2 = \{T1, T2, T3\} \in \mathbb{R}^{N_2 \times 3072}$. In this case study, the training dataset for Domain One only consist of 50 observations for class 0, 1, and 2 (i.e. $N_1^{train} = 150$) with no training data for the removal of panel 3. This reflects a scenario where the monitoring system was not designed initially to target locating damage to panel 3. In contrast, the training dataset for Domain Two has 10 observations for all four classes (i.e. $N_2^{train} = 40$). A visualisation of the training dataset is presented in Figure 3, with the transmissibilities stacked for each observation. Figure 3 also shows the first two principal components for each domain, highlighting the data shift between domains and the need for heterogeneous transfer learning.

3.1 Comparison of Heterogeneous Transfer Learning Methods

KBTL and HFA were applied to the Gnat dataset in order to compare their performance. HFA has been applied in both combinations, i.e. consider Domain One the source and Domain Two the target and Domain Two as the source and Domain One as the target. The techniques were also benchmarked against two single-domain approaches

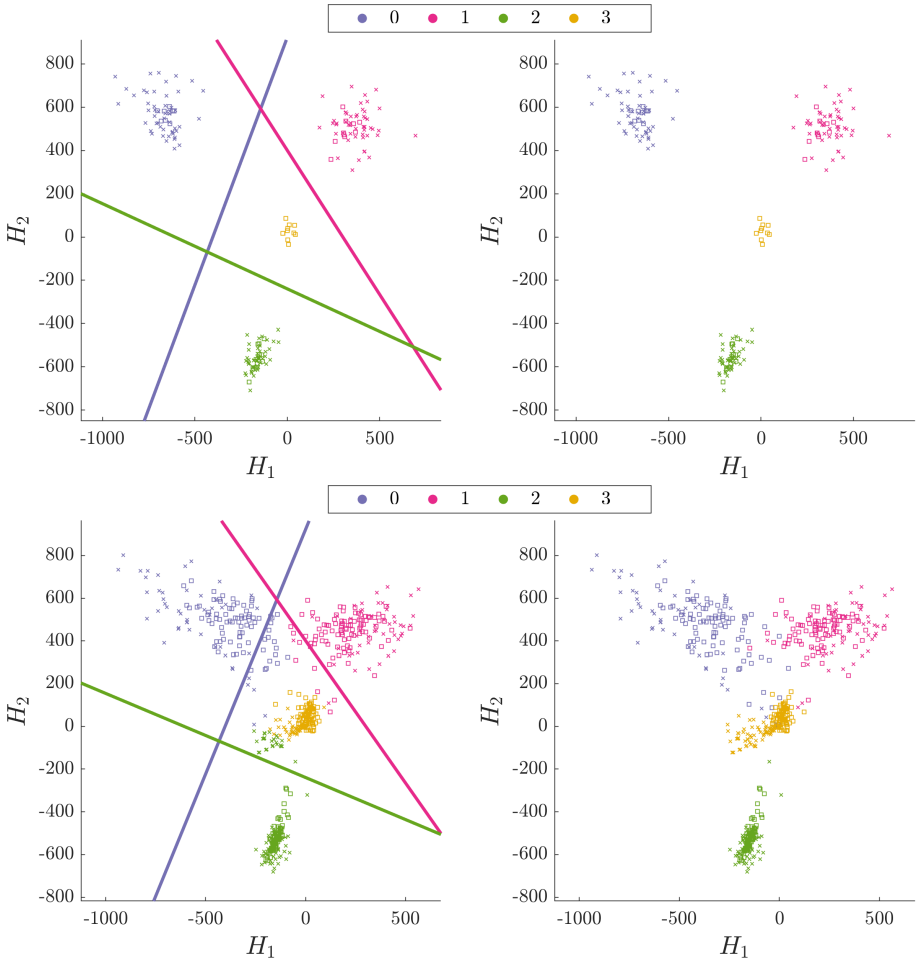


Figure 4: Visualisation of KBTL latent space and classifier for training (top) and testing (bottom) data. The left panels depict the one-vs-all discriminative classifiers (mean (-) and three standard deviations (- -)) for each class (indicated by the colour) as well as the predicted label MAP estimates. The right panels present the true labels. Each domain is denoted by a different symbol for Domains One (×) and Two (□).

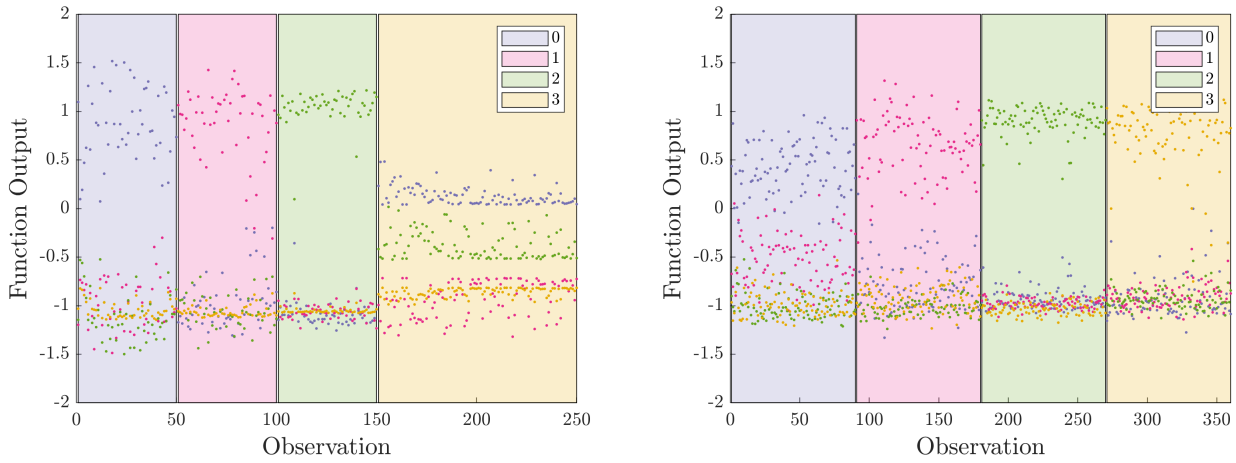


Figure 5: Visualisation of HFA one-vs-all function predictions on corresponding target domain. Left panel shows predictions on Domain One when Domain Two is the source domain, and the right panel shows predictions on Domain Two when Domain One is the source domain.

representing conventional machine learning methods with no knowledge transfer, namely Single-Domain KBTL (SD-KBTL) (KBTL trained and tested on a single domain), and an RVM, trained using the algorithm in [35] with a Bernoulli likelihood (used in a one-vs-all approach). In each approach a radial-basis kernel was implemented where the scale hyperparameter was defined for each domain using the median heuristics approach [42].

The KBTL hyperparameters were set to $(\kappa_\lambda, \theta_\lambda) = (\kappa_\gamma, \theta_\gamma) = (\kappa_\eta, \theta_\eta) = (1 \times 10^{-3}, 1 \times 10^{-3})$ due to the small number of observations in each domain and the latent space standard deviation was $\sigma_h = 1$ reflecting a degree of uncertainty in mapping, with a small margin of $\nu = 0.1$ used to try and aid separability. For visualisation purposes $R = 2$ was used. The trade-off parameters for HFA were set as $C = 1$ and $\lambda = 100$, with the approach repeated in a one-vs-all manner for each class label.

As the KBTL latent subspace was two-dimensional, the latent space and classifier can be visualised. Figure 4 displays the expected posterior latent space and classifier function, with the label MAP estimates on the left panels and the true labels on the right panels. The figure also shows the generalisation of the approach with the training and testing data in the top and bottom panels respectively. KBTL has placed class $Y = 3$ at the origin of the latent space as a result of limited information provided by Domain Two.

The high-dimensional augmented feature space in HFA means that only the function predictions at each observation are displayed in Figure 5. The results show that in the scenario where Domain One is the source domain (right panel), the fact that $Y = 3$ occurs in the target domain means that the classifier has inferred a generalised form that applies to the testing target data. However, the inverse is true when Domain Two is used as the source domain, where the

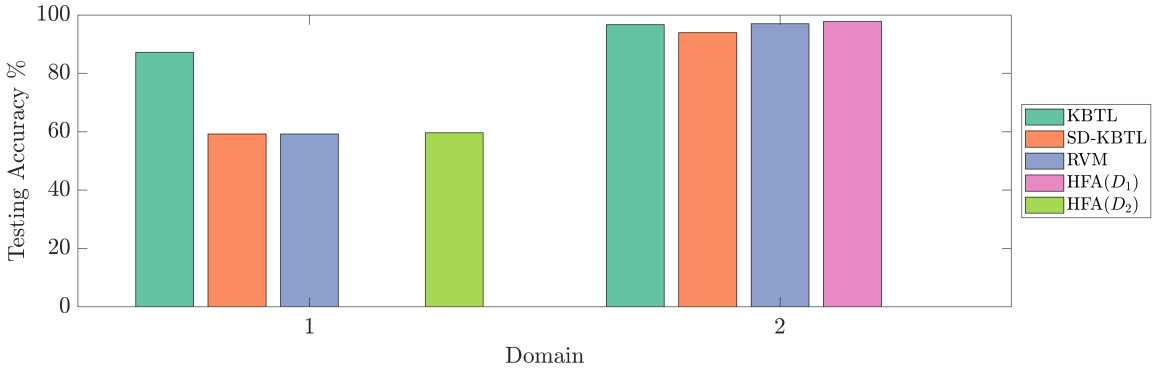


Figure 6: Comparison of testing accuracies. HFA(D_i) refers to the results where the i^{th} domain is considered the source domain. Shaded regions denote the true label.

lack of $Y = 3$ in the target training set has caused confusion.

Finally the testing accuracies are compared in Figure 6. These results show that KBTL has more successfully managed to transfer knowledge between Domains Two and One than all other approaches, with HFA only marginally improving performance of conventional single-domain approaches. However, in the well-labelled target scenario, where knowledge is transferred from Domain One to Domain Two, the heterogeneous transfer learning approaches and conventional approaches show comparable results, with HFA achieving the highest classification accuracy. This demonstrates the challenges with KBTL. By trying to learn one generalised latent space and classifier for all domains, KBTL classification performance might not be improve on a well-labelled domain in order to improve classification on other sparsely labelled domains.

4 Discussion and conclusions

Transferring label knowledge between different datasets is part of a population-based approach to SHM. This paper has investigated the use of two heterogeneous transfer learning methods, KBTL and HFA, that allow for knowledge to be transferred between feature spaces that are inconsistent and have different dimensions.

The algorithms were applied to datasets from a Gnat training aircraft where, over both domains, KBTL outperformed HFA. The ability to construct one classification model over all domains, rather than just transferring from a single source to target, meant that KBTL was better able to generalise to all domains. Furthermore, both heterogeneous transfer learning methods outperform conventional approaches to machine learning. The results show that it is possible to map between inconsistent datasets and overcome data shift, meaning a wider range of PBSHM problems can be addressed. Future research will seek to investigate the limitations of these approaches as feature inconsistency increases in a PBSHM context.

Acknowledgements

The authors would like to acknowledge the support of the UK Engineering and Physical Sciences Research Council via grants EP/R006768/1, EP/R003645/1 and EP/R004900/1.

References

[1] K. Worden and J. M. Dulieu-Barton, "An overview of intelligent fault detection in systems and structures," *Structural Health Monitoring*, vol. 3, no. 1, pp. 85–98, 2004.

[2] K. Worden and G. Manson, "The application of machine learning to structural health monitoring," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 515–537, 2007.

[3] E. Figueiredo, G. Park, C. R. Farrar, K. Worden, and J. Figueiras, "Machine learning algorithms for damage detection under operational and environmental variability," *Structural Health Monitoring*, vol. 10, no. 6, pp. 559–572, 2011.

[4] C. R. Farrar and K. Worden, *Structural Health Monitoring: a Machine Learning Perspective*. Chichester, UK: John Wiley & Sons, Ltd, 2012.

[5] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. MIT press, 2006.

[6] L. A. Bull, K. Worden, and N. Dervilis, "Towards semi-supervised and probabilistic classification in structural health monitoring," *Mechanical Systems and Signal Processing*, vol. 140, p. 106653, 2020.

[7] L. A. Bull, P. Gardner, J. Gosliga, N. Dervilis, E. Papatheou, A. E. Maguire, C. Campos, T. J. Rogers, E. J. Cross, and K. Worden, "Foundations of population-based structural health monitoring, Part I: Homogeneous populations and forms," *Mechanical Systems and Signal Processing*, vol. 148, p. 107141, 2021.

[8] J. Gosliga, P. Gardner, L. A. Bull, N. Dervilis, and K. Worden, "Foundations of population-based structural health monitoring, Part II: Heterogeneous populations and structures as graphs, networks, and communities," *Mechanical Systems and Signal Processing*, vol. 148, p. 107144, 2021.

[9] P. Gardner, L. A. Bull, J. Gosliga, N. Dervilis, and K. Worden, "Foundations of population-based structural health monitoring, Part III: Heterogeneous populations, transfer and mapping," *Mechanical Systems and Signal Processing*, vol. 149, p. 107142, 2021.

[10] M. Gönen and A. A. Margolin, "Kernelized Bayesian transfer learning," in *Proceedings of the National Conference on Artificial Intelligence*, 2014.

[11] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, pp. 1345–1359, 2010.

[12] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, p. 29, 2017.

[13] O. Day and T. M. Khoshgoftaar, "A survey on heterogeneous transfer learning," *Journal of Big Data*, vol. 4, p. 29, 2017.

[14] L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for heterogeneous domain adaptation," in *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, 2012.

[15] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 1134–1148, 2014.

[16] A. Rytter, "Vibrational based inspection of civil engineering structures," Ph.D. dissertation, Aalborg University, Denmark, 1993.

[17] 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*, vol. 6, pp. 26 241–26 253, 2018.

[18] 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*, vol. 186, pp. 1031–1045, 2018.

[19] A. Vetrivel, M. Gerke, N. Kerle, F. Nex, and G. Vosselman, "Disaster damage detection through synergistic use of deep learning and 3D point cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, pp. 45–59, 2018.

[20] Y. Gao and K. M. Mosalam, "Deep transfer learning for image-based structural damage recognition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 748–768, 2018.

[21] C. Feng, H. Zhang, S. Wang, Y. Li, H. Wang, and F. Yan, "Structural damage detection using deep convolutional neural network and transfer learning," *KSCE Journal of Civil Engineering*, no. 23, pp. 4493–4502, 2019.

[22] K. Jang, N. Kim, and Y. An, "Deep learning-based autonomous concrete crack evaluation through hybrid image scanning," *Structural Health Monitoring*, p. 147592171882171, 2019.

[23] N. Dhieb, H. Ghazzai, H. Besbes, and Y. Massoud, "A very deep transfer learning model for vehicle damage detection and localization," in *2019 31st International Conference on Microelectronics (ICM)*, 2019, pp. 158–161.

[24] M. Azimi, A. D. Eslamlou, and G. Pekcan, "Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review," *Sensors*, vol. 20, no. 10, 2020.

[25] 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, p. 798147.

[26] J. Ye, T. Kobayashi, H. Tsuda, and M. Murakawa, "Robust hammering echo analysis for concrete assessment with transfer learning," in *Proceedings of the the 11th International Workshop on Structural Health Monitoring*, 2017, pp. 943–949.

[27] 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*, vol. 17, no. 2, 2017.

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

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

[30] P. Gardner and K. Worden, "On the application of domain adaptation for aiding supervised SHM methods," in *Proceedings of the 12th International Workshop on Structural Health Monitoring*, Stanford, USA, 2019, pp. 3347–3357.

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

[32] H. Wan and Y. Ni, "Bayesian multi-task learning methodology for reconstruction of structural health monitoring data," *Structural Health Monitoring*, vol. 18, pp. 1282–1309, 2019.

[33] Y. Huang, J. L. Beck, and H. Li, "Multitask sparse Bayesian learning with applications in structural health monitoring," *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 9, pp. 732–754, 2019.

[34] Y. Zhang and Q. Yang, "An overview of multi-task learning," *National Science Review*, vol. 5, pp. 30–43, 2018.

[35] M. E. Tipping, "The relevance vector machine," in *Advances in Neural Information Processing Systems*. MIT Press, 2000, pp. 652–658.

[36] K. Worden, G. Manson, and D. Allman, "Experimental validation of a structural health monitoring methodology: Part I. Novelty detection on a laboratory structure," *Journal of Sound and Vibration*, vol. 259, no. 2, pp. 323–343, 2003.

[37] 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*, vol. 259, no. 2, pp. 345–363, 2003.

[38] ——, "Experimental validation of a structural health monitoring methodology: Part III. Damage location on an aircraft wing," *Journal of Sound and Vibration*, vol. 259, no. 2, pp. 365–385, 2003.

[39] K. Worden, G. Manson, G. Hilson, and S. G. Pierce, "Genetic optimisation of a neural damage locator," *Journal of Sound and Vibration*, vol. 309, no. 3, pp. 529–544, 2008.

[40] L. A. Bull, K. Worden, R. Fuentes, G. Manson, E. J. 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*, vol. 453, pp. 126–150, 2019.

[41] G. Tsialiamanis, D. J. Wagg, P. Gardner, N. Dervilis, and K. Worden, "On partitioning of an SHM problem and parallels with transfer learning," in *Proceedings of IMAC XXXVIII International Conference on Modal Analysis*, Houston, USA, 2020.

[42] A. Gretton, B. Sriperumbudur, D. Sejdinovic, H. Strathmann, and M. Pontil, "Optimal kernel choice for large-scale two-sample tests," in *Neural Information Processing Systems*, 2012, pp. 1205–1213.