Hierarchical clustering of materials with defects using impact-echo testing

Jorge Igual

***Abstract*—Signals obtained from impact-echo techniques can be used to detect and classify defects in damaged materials. The defects change the wave propagation between the impact and the sensors producing particular spectrum elements which define the feature vector. We propose a hierarchical clustering method that models the feature vector as a mixture of Gaussians (MoG) for every class and then merge the different clusters using as a distance measure the Kullback-Leibler (KL) divergence. Since there is not a closed form solution to the KL divergence between MoG, some approximations are introduced. We apply the hierarchical clustering algorithms to signals obtained from real specimens made of aluminum alloy. The samples are classified in four classes according to the state: homogeneous (no defect), one hole, one crack and multiple defects. We compare the performance of the different approximations and discuss the dendrograms that are obtained. Similar kinds of defects are clustered first and more importantly, the high level hierarchy is able to distinguish between the defective and non defective materials.**

***Index Terms*—Impact echo, mixture of Gaussians, hierarchical clustering, sensors, Kullback-Leibler divergence, classification.**

I. Introduction

NON Destructive Testing (NDT) analysis consists of obtaining useful information about the state of the material under study preserving the integrity of the tested specimen. Non destructiveness **is required** in applications where the analyzed material is irreplaceable, such as historical buildings or artistic works, or in applications where the cost of the destruction of the sample is very high, such as in the marble industry.

Some other popular NDT applications are the inspection and defects characterization in power plants, medicine, aerospace, military, fuel storage and transportation (see [1] for a review of engineering materials and composites applications). There is also a plethora of NDT techniques based on optical [2], audio, radiological, electromagnetic, laser, chemical, termographic and other types of signals [3]. The most common acoustic NDT techniques are ultrasonic and impact echo (IE) [4],[5]. In IE applications, a hammer hits the material and its acoustic response is recorded by sensors located on the surface of the material.

Traditional material studies are based on time and/or frequency analysis of the signals [2],[6],[7],[8],[9]. More recently, the machine learning approaches have received a lot of attention. However, most of these efforts are based mainly on the application of artificial neural networks (ANN): monitoring of rotary machinery systems [10], defective states of catenary support devices [11], fault detection of induction motors [12], identification of the moisture content in brick walls of historic buildings [13], prediction of the concrete compressive strength and thickness of concrete structures [14], prediction of the internal grouting quality of prestressed ducts [15] and identification of the pull-off adhesion of the concrete layers in floors on the basis of parameters evaluated on the structural layer surface [16].

In previous work [17], we presented a machine learning based semisupervised classifier to determine the kind of defect in the structure of the material using IE signals. We introduced a Bayesian classifier based on the modeling of the class conditional probabilities by a mixture of Gaussians (MoG). In Figure 1 we summarize how the models for each class are obtained. 1881 executions of the IE test from different specimens made of aluminum alloy series 2000 of dimensions 7x5x22 cm (width, height, and length, respectively) are carried out and the signals are recorded by seven sensors. In order to simulate defective materials, up to three defects per piece were drilled in different locations of some pieces. The defects consisted of holes (10 mm ϕ cylinders) and cracks (5x20 mm cross-section parallelepipeds). The spectrum of the IE recorded signals, preprocessed by Principal Component Analysis (PCA) in order to reduce the dimensions of the problem [18] is the input data to the Bayesian model. PCA is a common technique in defect classification problems [19]. We choose the number of principal components such that 95% of the variance was retained. As a result, the feature vector had seven dimensions. The next step consists of obtaining the class conditional distributions for each class, i.e., the MoG models. We use a semisupervised variation of the EM algorithm. In order to simulate different real conditions, we introduce a supervision parameter in the EM algorithm. It indicates the percentage of samples with a pre known class used during the training of the models. The feature vectors with a known class are used only to learn the corresponding model. The feature vectors with an unknown preassigned class are used in all models weighted by the corresponding posterior probability. We split the samples randomly into two groups: a training set containing the 80% of the data and a testing set with the rest of samples. We apply the EM algorithm to the training data, obtaining four class models according to its defective status: homogeneous, one defect (one hole or one crack) or multiple defects class. In [17] we used those models for classification purposes: the classification problem consists of assigning each sample in the testing data set to the class with a higher posterior probability (Bayesian maximum a posteriori classifier). For a detailed explanation about the specimens, IE experiments and measurements, and perfor-

The author is with the Departamento de Comunicaciones, Universitat Politècnica de València, Valencia 46022, Spain (e-mail: jigual@dcom.upv.es).

mance analysis of the classifier, see [17]; for more information about the spectra of the recorded signals, how PCA is applied and how the feature vector is obtained, see [20].

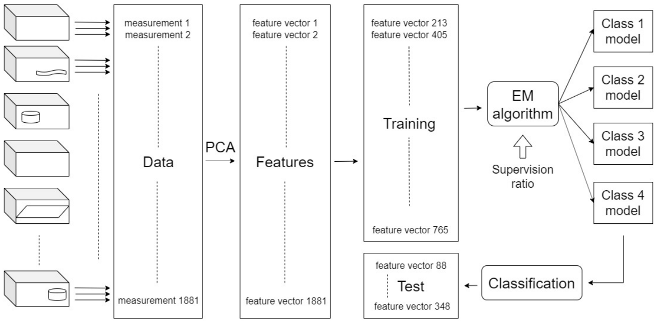

Fig. 1. Method to obtain the mixture of Gaussians model for each class. The measurements are preprocessed by PCA to obtain the 7 dimension feature vector. Features are shuffled and split in training (80%) and testing (20%) data sets. The training data are used to train the models by EM and the testing data for classification. The supervision ratio controls the percentage of samples that have a pre known class during the learning of the models.

The most important advantage of obtaining a generative model such as a Bayesian classifier with respect to discriminative classifiers such as those based on ANN is to obtain posterior probabilities for every class, so this probability can be used in many different ways, not only for classification purposes as it is in Figure 1 and [17].

In this paper, we will assume that the models are already estimated. We will use the class conditional distributions to present some hierarchical clustering procedures in order to solve different granularity classification problems without the need to train new models. For example, using the models obtained to solve the four classes problem (homogeneous materials with no defect, with one kind of defect, with another kind of defect or with multiple defects), how to merge hierarchically the four classes into a hierarchy of clusters with at the end only two groups (homogeneous and with any kind of defect) using as the starting point the 4-classes model and some distance measurement as the variable to establish in which order the classes must be grouped.

It is important to note that the goal of the paper is not to improve the classification results obtained by the Bayesian classifier, but to study if the class models obtained to solve the four classes problem can be used to solve other problems, such as a hypothetical two classes problem (homogeneous or defective). If that is the case, it will show how the clustering naturally, without supervision, groups physically similar cases (defective materials vs. homogeneous blocks) and, more importantly, it will reduce the time in material quality assessment since the models learned to solve one problem can be used to solve higher level problems.

II. Hierarchical clustering

The starting point are the MoG class conditional distributions for each class obtained by the EM semisupervised algorithm. The next step is how to merge the classes in order to obtain an agglomerative hierarchical clustering. The results of these groupings we call clusters, and the original bottom level of the hierarchy (the input to the hierarchical clustering) we call classes. As an example of clustering in NDT, see [21], where clustering is used for honeycomb detection in concrete. Note that in that paper, as it is common in clustering, the samples are grouped based on a distance criterion. However, in our proposal, we group classes, not samples.

A proximity or similarity measure is the basis for most clustering algorithms [22]. This measure between clusters at one level in the hierarchy (also referred to as distance) is used to determine which of them will be merged. The distance between two clusters can be estimated between pairs of data objects of each of the clusters or between probabilistic relationships of the data densities of the two clusters. Since we have a probabilistic model for every class, we follow this approach.

The distance between a cluster l and a new cluster formed by the merging of two clusters i and j is $D(C_l,(C_i,C_j)) = \alpha D(C_l,C_i) + \beta D(C_l,C_j) + \gamma D(C_i,C_j) + \epsilon |D(C_l,C_i) - D(C_l,C_j)|$

By manipulating the coefficients α, β, γ and ϵ several agglomerative hierarchical algorithms of clustering based on distances between data objects can be derived. Note that, if $\alpha = \beta = \epsilon = 1/2$ and $\gamma = 0$, (1) becomes the complete linkage method: $D(C_l,(C_i,C_j)) = \max(D(C_l,C_i), D(C_l,C_j))$, while $\alpha = \beta = 1/2$, $\gamma = 0$, and $\epsilon = -1/2$ corresponds to the single linkage method $D(C_l,(C_i,C_j)) = \min(D(C_l,C_i), D(C_l,C_j))$.

The probabilistic approaches to hierarchical clustering consider model-based criteria or Bayesian hypotheses to decide on merging clustering rather than using an ad-hoc distance metric. Basically, there are two approaches to derive the hierarchy: hierarchical generative modelling of the data or hierarchical ways of organizing nested clusters. Methods of the first approach include the following hierarchical generative models, for instance: Gaussian-based [8], Dirichlet-based [9]. In [11], an agglomerative algorithm to merge Gaussian mixtures is presented. It considers a virtual sample generated from the model at a level and uses expectation-maximization (EM) to find the expressions for the mixture model parameters for the next level that best explain the virtual sample. In [23] an alternative based on Independent Component Analysis mixers is proposed.

III. Hierarchical clustering of mixture of Gaussians

The conditional probability density function (pdf) of an observation vector $\mathbf{x}$ for cluster C_k^h, $k = 1, 2, ..., K - h + 1$ at level $h = 1, 2, ..., K$ of the hierarchy is $f\left(\mathbf{x}/C_k^h\right)$. This pdf is modeled by a MoG for every class at level $h = 1$. The MoG (or Gaussian mixture model) is a weighted sum of Gaussians with mean μ_k, covariance matrix Σ_k and dimensionality d.

$$f(\mathbf{x}) = \sum_{k=1}^{K} \alpha_k N_k\left(\mathbf{x}; \mu_k, \Sigma_k\right)$$

$$N_k(\mathbf{x}; \mu_k, \Sigma_k) = (2\pi)^{-d/2} \left|\Sigma_k\right|^{-1/2} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)} \tag{1}$$

Each Gaussian contributes to the mixture model in the proportion or mixing coefficient α_i, with $\alpha_i \geq 0$ and $\sum_{k=1}^{K} \alpha_i = 1$. These weights can also be interpreted as priors, indicating the prior probability of the data coming from the corresponding Gaussian of the mixture. When an observation is available, we can apply the Bayes theorem to calculate the posterior probability, i.e., the responsibility that the observation comes from each component of the mixture model.

The estimation of the parameters is obtained by the EM algorithm, where the new estimated parameters become the guess for the next iteration. Given N observations $\mathbf{x}_n, n = 1, \ldots, N$, each iteration consists on two steps. The expectation E step calculates the posterior probabilities $p(k/\mathbf{x}_n)$, i.e., the responsibility that the k^{th} distribution takes for generating the n^{th} observation:

$$p(k/\mathbf{x}_n) = \frac{\alpha_k N_k\left(\mathbf{x}_n; \mu_k, \Sigma_k\right)}{\sum_{k=1}^{K} \alpha_k N_k\left(\mathbf{x}_n; \mu_k, \Sigma_k\right)} \tag{2}$$

The maximization M step updates the parameters:

$$\begin{aligned} \mu_k &= \frac{\sum_{n=1}^{N} p(k/\mathbf{x}_n) x_n}{\sum_{n=1}^{N} p(k/\mathbf{x}_n)} \\ \Sigma_k &= \frac{\sum_{n=1}^{N} p(k/\mathbf{x}_n)(\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T}{\sum_{n=1}^{N} p(k/\mathbf{x}_n)} \\ \alpha_k &= \frac{1}{N}\sum_{n=1}^{N} p(k/\mathbf{x}_n) \end{aligned} \tag{3}$$

The algorithm is applied iteratively until convergence. Once the distribution for every class is estimated, we can measure their similarity and merge them according to some criterion obtaining a hierarchical clustering. Since we are using a probabilistic framework, the most logical similarity measurement is the differential relative entropy or Kullback-Leibler KL divergence $D(f\|g)$, although it is not a true distance measure since it is not symmetric, i.e., $D(f\|g) \neq D(g\|f)$. A simple extension is the symmetric KL divergence $D(f\|g) + D(g\|f)$, which is used in the work presented here.

The KL divergence between two distributions $f(x)$ and $g(x)$ is [24]:

$$D(f\|g) = \int f(\mathbf{x}) \log \frac{f(\mathbf{x})}{g(\mathbf{x})} d\mathbf{x} \tag{4}$$

In the case of Gaussian distributions, $f(\mathbf{x}) = N\left(\mathbf{x}; \mu_f, \Sigma_f\right)$, $g(\mathbf{x}) = N\left(\mathbf{x}; \mu_g, \Sigma_g\right)$, the KL divergence is:

$$\begin{aligned} D(f\|g) = \ & \tfrac{1}{2}[\log \tfrac{|\Sigma_g|}{|\Sigma_f|} + Tr(\Sigma_g^{-1}\Sigma_f) - d + \\ & +(\mu_f - \mu_g)^T \Sigma_g^{-1}(\mu_f - \mu_g)] \end{aligned} \tag{5}$$

However, there is not an analytical solution to the KL divergence between MoGs. This fact implies that the KL divergence must be estimated in an approximated way.

IV. Estimation of the Kullback-Leibler divergence between mixture of Gaussians

There are three main approaches to estimate $D(f\|g)$: first, transforming (4) in a tractable equation using simplified versions of the MoG, e.g., reducing the mixture to just a single Gaussian; second, using approximations to the KL divergence definition; third, estimating (4) numerically, e.g., using Monte Carlo or another kind of sampling.

A. *Approximations based on the modification of the mixture of Gaussians model*

The simplest way to approximate a MoG is its substitution by a single Gaussian. The most intuitive way to carry out this simplification is to approximate the MoG model in (1) by a single Gaussian $N(\mathbf{x}; \hat{\mu}, \hat{\Sigma})$. This is the same problem that clustering Gaussians with a single Gaussian. The optimal parameters $(\hat{\mu}, \hat{\Sigma})$ minimize the cumulative differential relative entropy between the single Gaussian and the components of the MoG. The values are given by [25]:

$$\begin{aligned} \hat{\mu} &= \sum_{k=1}^{K} \alpha_k \mu_k \\ \hat{\Sigma} &= \sum_{k=1}^{K} \alpha_k (\Sigma_k + (\mu_k - \hat{\mu})(\mu_k - \hat{\mu})^T) \end{aligned} \tag{6}$$

As it was expected, the mean of the single Gaussian is the weighted average of the means of the Gaussians of the mixture. However, the covariance is not just the average of the covariances; this value is modified to take into account the distance between the mean of the corresponding Gaussian of the mixture and the mean of the single Gaussian.

After this simplification, since every class is modeled by a single Gaussian, the estimation of the KL divergence $D_G(f\|g)$ is obtained using (5):

$$\begin{aligned} D_G(f\|g) = \ & \tfrac{1}{2}[\log(|\hat{\Sigma}_g|/|\hat{\Sigma}_f|) + Tr(\hat{\Sigma}_g^{-1}\hat{\Sigma}_f) - d + \\ & +(\hat{\mu}_f - \hat{\mu}_g)^T \hat{\Sigma}_g^{-1}(\hat{\mu}_f - \hat{\mu}_g)] \end{aligned} \tag{7}$$

B. *Approximations based on modifications of the KL divergence*

Another option is, instead of reducing the MoG distribution to a Gaussian one, to modify the definition of $D(f\|g)$.

In this case, we keep the MoG model, but the distance measure is modified to obtain a closed form equation. One option is to define the distance $D_{\min}(f\|g)$ as the KL divergence between the components of f and g that have minimum divergence:

$$D_{\min}(f\|g) = \min_{i,j} D(f_i\|g_j) \tag{8}$$

Like in (6), the final expression is the KL divergence between two Gaussians. In other words, this approximation is equivalent to reducing the original distributions by single Gaussians, in this case the ones that are closest in the KL divergence sense.

Note that these approximations can obtain very poor results; e.g., when the closest components in the mixture model of f and g are the same Gaussians but the rest of components are very far away; in this case, f and g are very different, but $D_{\min}(f\|g) = 0$. The good news in these simplifications is that they are very attractive considering the computational cost, since they provide an analytical solution. In addition, they can be helpful in situations where the number of modes of the mixture is reduced and they are close.

In the case that distributions f and g have the same number of components $N_f = N_g = N$, an upper bound of KL divergence can be calculated [26], $D_{Do}(f\|g) \geq D(f\|g)$, where:

$$D_{Do}(f\|g) = \sum_{i=1}^{N} \alpha_i \left(\log \frac{\alpha_i}{\beta_i} + D(f_i\|g_i) \right) \tag{9}$$

with β_i the weights of the $g(x)$ pdf. $D_{Do}(f\|g)$ is a weighted sum of distances between the components of the two mixture models. It means that the value that is obtained depends on the way the components are ordered. Since (9) is an upper bound, we can define a matching function between Gaussians in every mixture model such that the summation in (9) is minimum. This is the approximation proposed in [27], $D_{Gold}(f\|g)$. If we alleviate the $N_f = N_g$ restriction and define the matching function such as $j(i) = \arg\min_k (D(f_i\|g_k) - \log(\beta_k))$, we obtain:

$$D_{Gold}(f\|g) = \sum_{i=1}^{N_f} \alpha_i \left(\log \frac{\alpha_i}{\beta_{j(i)}} + D(f_i\|g_{j(i)}) \right) \tag{10}$$

Note that, since $N_f \neq N_g$, $D_{Gold}(f\|g)$ is not necessarily an upper bound of the true KL divergence.

Analyzing the definition of $D(f\|g)$, it is clear that another family of estimators can be obtained when the likelihood function included in it, $E_f[\log g] = \int f(\mathbf{x}) \log g(\mathbf{x}) d\mathbf{x}$ is approximated. For example, we can use Jensen's inequality $E[\phi(f)] \geq \phi(E[f])$ with the convex function $\phi(.) = -\log(.)$, as it is used in information theory to demonstrate that the KL divergence between two distributions is greater or equal than zero. Then, an upper bound of the likelihood $E_f[\log g]$ substitutes the true value in equation (4), obtaining an approximation $D_{prod}(f\|g)$ that underestimates the KL divergence:

$$D_{prod}(f\|g) = \sum_{i=1}^{N_f} \alpha_i \log \frac{\sum_{l=1}^{N_f} \alpha_l \int f_i(\mathbf{x}) f_l(\mathbf{x}) d\mathbf{x}}{\sum_{j=1}^{N_g} \beta_j \int f_i(\mathbf{x}) g_j(\mathbf{x}) d\mathbf{x}} \tag{11}$$

Taking into account that all the terms in the integrals of (11) are Gaussians, the integrals become the normalizing factor in a product of Gaussians. Therefore, the value of the integrals in the numerator and denominator are, respectively:

$$\begin{aligned} \int f_i(\mathbf{x}) f_l(\mathbf{x}) d\mathbf{x} = \quad & (2\pi)^{-d} |\Sigma_{f_i} + \Sigma_{f_l}|^{-1/2} \\ & \cdot e^{-\frac{1}{2}(\mu_{f_i} - \mu_{f_l})^T (\Sigma_{f_i} + \Sigma_{f_l})^{-1} (\mu_{f_i} - \mu_{f_l})} \\ \int f_i(\mathbf{x}) g_j(\mathbf{x}) d\mathbf{x} = \quad & (2\pi)^{-d} |\Sigma_{f_i} + \Sigma_{g_j}|^{-1/2} \\ & \cdot e^{-\frac{1}{2}(\mu_{f_i} - \mu_{g_j})^T (\Sigma_{f_i} + \Sigma_{g_j})^{-1} (\mu_{f_i} - \mu_{g_j})} \end{aligned} \tag{12}$$

The next approximation is inspired in the opposite idea. Instead of using Jensen's inequality to take the log out of the integrals in the calculation of the expectations $E_{f_i}[\log g] = \int f_i(\mathbf{x}) \log g(\mathbf{x}) d\mathbf{x}$, we use it to put the log into the summation of the mixture models when calculate the likelihood $E_f[\log g] = \int f(\mathbf{x}) \log \sum_{j=1}^{N_g} \beta_j g_j(\mathbf{x}) d\mathbf{x}$, with $g_j(x) = N(\mathbf{x}; \mu_j, \Sigma_j)$, and analogously with $E_f[\log f]$. Doing this, we obtain a lower bound of the likelihood, since $E_f[\log g] \geq \int f(\mathbf{x}) \sum_{j=1}^{N_g} \delta_{j|i} \log \frac{\beta_j g_j(\mathbf{x})}{\delta_{j|i}} d\mathbf{x}$, where $\delta_{j|i}$ are the variational parameters introduced to maximize the lower bound, with the constraints $\delta_{j|i} \geq 0$ and $\sum_j \delta_{j|i} = 1$. We proceed in the same way with $E_f[\log f]$; in this case the parameters are $\epsilon_{k|i}$, so $E_f[\log f] \geq \int f(x) \sum_{k=1}^{N_f} \epsilon_{k|i} \log \frac{\alpha_k f_k(\mathbf{x})}{\epsilon_{k|i}} d\mathbf{x}$. Using the values of $\delta_{j|i}$ and $\epsilon_{k|i}$ that optimize the approximation, the KL divergence becomes $D_{\text{var}}(f\|g)$ [28]:

$$D_{\text{var}}(f\|g) = \sum_{i=1}^{N_f} \alpha_i \log \frac{\sum_{k=1}^{N_f} \alpha_k e^{-D(f_i\|f_k)}}{\sum_{j=1}^{N_g} \beta_j e^{-D(f_i\|g_j)}} \tag{13}$$

$D_{\text{var}}(f\|g)$ is a variational version of $D_{Gold}(f\|g)$, where instead of defining a strict matching function between every component in the different mixture models we prefer to average over all the components using the variational parameters.

There is another way to apply a variational approach. It consists **on** introducing the variational parameters not only in the components into the log term in the likelihood function but in the distribution itself, i.e.:

$$f(\mathbf{x}) = \sum_{i=1}^{N_f} \alpha_i f_i(\mathbf{x}) = \sum_{i=1}^{N_f} \sum_{j=1}^{N_g} \delta_{j|i} f_i(\mathbf{x}) \tag{14}$$

with non negative parameters summing up to the mixture coefficient for every component, i.e., $\sum_j \delta_{j|i} = \alpha_i$.

In the case of g, we have:

$$g(\mathbf{x}) = \sum_{j=1}^{N_g} \beta_j g_j(\mathbf{x}) = \sum_{i=1}^{N_f} \sum_{j=1}^{N_g} \epsilon_{i|j} g_j(\mathbf{x}) \tag{15}$$

with $\sum_i \epsilon_{i|j} = \beta_j$.

Using these distributions and Jensen's inequality again, it is possible to obtain an upper bound of $D(f\|g)$ [28]:

$$D_{vub}(f||g) = D(\delta||\epsilon) + \sum_{i=1}^{N_f} \sum_{j=1}^{N_g} \delta_{j|i} D(f_i||g_j) \geq D(f||g) \tag{16}$$

with $D(\delta||\epsilon) = \sum_{i=1}^{N_f} \sum_{j=1}^{N_g} \delta_{j|i} \log \frac{\delta_{j|i}}{\epsilon_{i|j}}$. The parameters are obtained **by** minimizing (16):

$$\begin{aligned} \epsilon_{i|j} &= \frac{\beta_j \delta_{j|i}}{\sum_k \delta_{j|k}} \\ \delta_{j|i} &= \frac{\alpha_i \epsilon_{i|j} e^{-D(f_i||g_j)}}{\sum_l \epsilon_{i|l} e^{-D(f_i||g_l)}} \end{aligned} \tag{17}$$

The updating equations for the parameters are applied iteratively until convergence. Since these equations are multiplicative, it is important that the initial values are not zero, e.g., $\epsilon_{i|j} = \delta_{j|i} = \alpha_i \beta_j$.

All the solutions presented up to now, excepting (16), obtain a closed form to estimate the approximated value of the KL divergence; in other words, they obtain an analytical solution. Only the variational solution includes an iterative algorithm to obtain the solution. Another approach to estimate $D(f\|g)$ is using numerical approximations.

C. Approximations based on sampling

Since there is not an analytical solution to $D(f\|g)$, another approach to estimate it is using sampling, e.g., Monte Carlo methods. The goal is to generate Q samples $\{\mathbf{x}_i\}_{i=1}^{Q}$ from distribution $f(x)$ and then estimate the expectation $D(f\|g) = \int f(\mathbf{x}) \log \frac{f(\mathbf{x})}{g(\mathbf{x})} d\mathbf{x}$ numerically:

$$D_{MC}(f\|g) = \frac{1}{Q} \sum_{i=1}^{Q} \log \frac{f(\mathbf{x}_i)}{g(\mathbf{x}_i)} \tag{18}$$

We can use as samples the observations obtained with the sensors or artificial samples generated according to the MoG models estimated in previous subsection. In this case, a simple option is to generate a responsability variable following the prior probabilities α_i and, then, generate the observation sample $\mathbf{x}_i$ with the probability function of the corresponding Gaussian component.

A related method consists in using the unscented transform [29]. In this case, the values where the log function is estimated are not obtained according to the Gaussian mixture models, but they are deterministic. The estimated value of the KL divergence is $D_u(f\|g)$:

$$D_u(f\|g) = \frac{1}{2K} \sum_{i=1}^{N_f} \alpha_i \sum_{k=1}^{2K} \log \frac{f(\mathbf{x}_{i,k})}{g(\mathbf{x}_{i,k})} \tag{19}$$

where the $\mathbf{x}_{i,k}$ points are, for $k = 1, \ldots, d$, $\mathbf{x}_{i,k} = \mu_i + (d\lambda_{i,k})^{1/2} e_{i,k}$ and $\mathbf{x}_{i,k+K} = \mu_i - (d\lambda_{i,k})^{1/2} e_{i,k}$, with $\lambda_{i,k}$ the eigenvalues of the covariance of the component $f_i(\mathbf{x})$, and $e_{i,k}$ the corresponding eigenvector.

Monte Carlo sampling is a very effective method, but the drawback is that it requires a lot of samples to guarantee a good estimate.

V. Results

The goal is to analyze the clustering of the previously estimated class models; i.e., which classes are more similar in the symmetric KL distance space. The interest from a hierarchical point of view is to: (i) merge first the one defect classes and (ii) more importantly, separate between the homogeneous and defective classes. We will use the following numbering system to identify the original classes: 1 (homogeneous), 2 (one hole defect), 3 (one crack defect) and 4 (multiple defects). For the sake of clarity and avoid confusions between the use of the words *class* and *cluster*, we will rename the original four classes as the initial four clusters at the bottom level of the hierarchy. During the first level hierarchy, a new cluster 5 will be obtained after merging two of the original classes. In the second level hierarchy a new cluster 6 will appear as a result of merging two first level hierarchy clusters. The proposed hierarchical clustering algorithm is summarized in Algorithm 1. For the first level clustering, we will use the different KL approximations explained in Section IV. For the second level, we will use different linkage methods of clusters.

To test the model complexity, several models were trained changing the number of Gaussians K per class: $K = 3, 5, 7, 9, 11, 13, 15, 17, 19, 21$. In addition, to study the influence of the supervision, i.e., the percentage of samples with a known class used during the learning of the MoG models, two different supervision ratios are analyzed: 30% and 90%. The higher the supervision ratio, the better the model, since the EM updating rules are mostly applied to the correct class conditional distribution. Finally, every MoG model and supervision case was run 40 times with different input data to test the consistency of the results (see Figure 1).

A. First level clustering

Once the different models for each class are obtained using the semisupervised EM algorithm, we calculate the symmetric KL divergence between classes and a new cluster is obtained merging the closest classes. The seven tested KL approximations are: D_G, D_{min}, D_{Gold}, D_{prod}, D_{var}, D_{vub} and D_{MC}, given by equations (7), (8), (10), (11), (13), (16) and (18), respectively.

In Figure 2 we show the results obtained with each method for the 90% supervision case. The horizontal axis shows the possible outcomes of the new cluster, i.e., the merging classes (the ones with the lowest KL divergence).

Algorithm 1: Hierarchical clustering algorithm

```
Data: The class models. Each class $c = 1,2,3,4$ is
      represented by a MoG with $K$ Gaussians
      $f_c(\mathbf{x}) = \sum_{k=1}^{K} \alpha_{k_c} N_k(\mathbf{x}; \mu_{k_c}, \Sigma_{k_c})$
Result: a hierarchical clustering.
Initialization (bottom level hierarchy): classes 1 to 4
  correspond to clusters 1 to 4.
First level hierarchy:
  (i) Calculate the symmetric KL distance between each pair of
      original classes (bottom level clusters).
  (ii) Merge the two classes with the minimum KL distance.
  (iii) Define it as cluster 5.
Second level hierarchy:
  (i) Calculate the proximity between each pair of clusters in the
      first level hierarchy.
  (ii) Merge the two more similar clusters.
  (iii) Define it as cluster 6.
Result: clusters 1, ..., 6
```

The vertical axis represents the model under study, i.e., the number of Gaussians used to model the class conditional distribution. The figure shows how many out of the 40 experiments are assigned to the corresponding cluster for each MoG model. The colormap used is at the bottom of the figure. The values range from a dark blue (0 cases) to a dark red (40 cases); cyan corresponds to around 15, green to 20, yellow to 25 and orange to 30 cases.

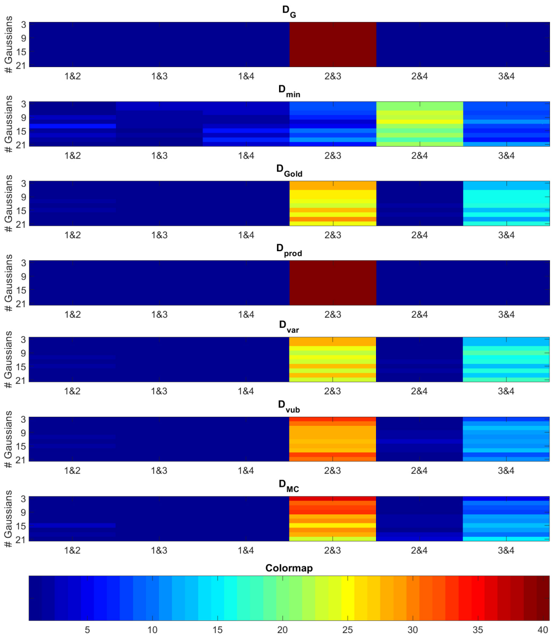

Fig. 2. First level hierarchy for different algorithms. Vertical axis: MoG model (number of Gaussians). Horizontal axis: classes merged. Bottom: colormap; e.g., algorithm D_G merges the classes 2 (one hole defect) and 3 (one crack defect) for all models and experiments (40 out of 40 for each MoG case).

Taking into account that class 1 means no defect at all, the new cluster does not have to include class 1 if the hierarchy is correct. As we can see in the figure, most clusters 1&2, 1&3 and 1&4 have a dark blue color (zero cases) for all methods but the D_{min} approximation. D_G and D_{prod} are the only two methods that always merge first the one hole and one crack classes (cluster 2&3). The worst results are obtained for the D_{min} distance, since class 1 is merged with the defective classes in some experiments, confirming that to estimate the distance between classes as the distance between the closest Gaussians is a too strong simplification.

As we also expected, D_{Gold} and D_{var} obtain similar results, as the second one can be interpretated as a variational version of the first one. The results with D_{vub} and D_{MC} are better and also similiar. The difference between these methods is the proportion of cases that join damaged materials with one hole and one crack (2&3 cluster) or materials with one crack and multiple defects (3&4 cluster).

In Table I we show the mean values in percentage for all the models. We confirm that D_{Gold} and D_{var} have a similar performance as it happens with D_{vub} and D_{MC}. The most important result is that all methods but D_{min} obtain an excellent result by not merging the class with no defects with any of the other classes (values close to 0 for columns 1&2, 1&3 and 1&4).

	1&2	1&3	1&4	2&3	2&4	3&4
D_G	0	0	0	100	0	0
D_{min}	3	2.25	6.5	16.75	52.25	19.25
D_{Gold}	0.5	0	0	64.25	0.5	34.75
D_{prod}	0	0	0	100	0	0
D_{var}	0.5	0	0	62.25	0.5	36.75
D_{vub}	0.5	0	0	72.75	1.55	25.25
D_{MC}	0.75	0	0	73.5	2	23.75

TABLE I
First Level hierarchy mean results for all models in %. Supervision ratio 90%.

B. Second level clustering

The ultimate goal is to separate in the high level clustering the pieces with no defect from the ones with any problem, i.e., to separate between class 1 and the rest of them. It means that in the second level hierarchy, the classes 2, 3 and 4 must be merged. We have seen that during the first merging, classes 2 and 3 are the closest ones for D_G and D_{prod} in all cases, while D_{Gold}, D_{var}, D_{vub} and D_{MC} in some cases prefer to merge first classes 3 and 4, and D_{min} failed in some experiments. The new cluster of the first level hierarchy is cluster 5; not considering the D_{min} approximation, cluster 5 is the merging of classes 2 and 3 (cluster 2&3) or 3 and 4 (cluster 3&4), depending on the case. It means that a correct second level merging to obtain the new cluster 6 must merge the clusters 4&5 in the case that 5 is 2&3, or 2&5 in the case that 5 is 3&4. For a perfect hierarchical classification, in no case, cluster 6 must include class 1.

For the first level hierarchy clustering we have used the KL distance. In the second level, since cluster 5 is composed of two classes, a measure of cluster proximity D must be defined. The two more similar clusters will be merged. We tested three different hierarchical clustering algorithms: the single linkage $D((C_i, C_j), C_l) = min(D(C_i, C_l), D(C_j, C_l))$, the complete linkage $D((C_i, C_j), C_l) = max(D(C_i, C_l), D(C_j, C_l))$ and the average linkage $D((C_i, C_j), C_l) = (D(C_i, C_l) + D(C_j, C_l))/2$ [30]. In Figure 3 we show the results of the second level hierarchy using the single linkage method.

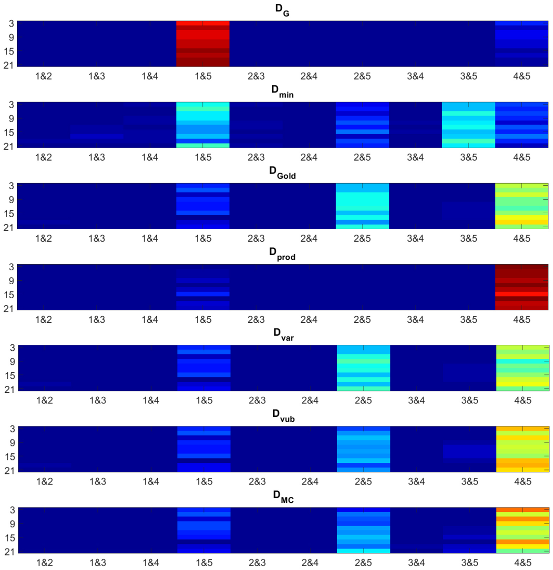

Fig. 3. Second level hierarchy for different algorithms and models. Same colormap as in Figure 2.

Observe that D_{prod} performance is almost perfect. It is the only approximation not only able to distinguish between zero, one or multiple defects in the first level hierarchy, but now it joins one defect and multiple defects classes in almost all the experiments: class 1 is not included in clusters 5 or 6, i.e., blocks with no defect at all are not mistaken by damaged materials in any level of the hierarchy. D_G method fails in this second level hierarchy. The simplification of the MoG model by only one Gaussian is the reason. The results are independent of the number of Gaussians in the MoG, since at the end all models are reduced to only one mean Gaussian when measuring distances. D_{min} also obtains poor results as we expected from first level hierarchy analysis. The rest of methods in the figure, D_{Gold}, D_{var}, D_{vub} and D_{MC}, have a similar good performance. In the cases where in the first level they joined class 2 and 3, now most of the time they add class 4 to this cluster; in the cases where they joined class 3 and 4 first, now they add mainly class 2.

The percentage results averaging across the different models are given in Table II. Summing up all the cases where class 1 is combined with any of the defective classes, the percentage is around 11% for D_{Gold}, D_{var}, D_{vub} and D_{MC}, and only 4.25% for D_{prod}. The incorrect merging is between class 1 and the previous merge of 2 and 3. In summary, most of the KL distance approximations are able to distinguish between defective and non defective materials using as a starting point the low level class distribution models.

	1&4	1&(2,3)	(2,3)&4
D_G	0	93.25	6.75
D_{prod}	0	4.25	95.75
D_{Gold}	0	11.5	52.75
D_{var}	0	11.5	50.75
D_{vub}	0	11.75	61
D_{MC}	0	12.25	61.25

	1&2	1&(3,4)	2&(3,4)
D_{Gold}	0.25	0	34.5
D_{var}	0.25	0	36.5
D_{vub}	0.25	0	25
D_{MC}	0	0	23.75

TABLE II

Second level hierarchy mean results in % for all models. Supervision ratio 90%.

We compare the single linkage hierarchical clustering method with the average and complete linkage methods. The difference is the way the proximity between clusters is defined: in single linkage it is the closest distance (the similarity between the two most similar instances in each cluster), in complete linkage the furthest distance (the similarity between the two most dissimilar instances in each cluster) and in average linkage is the average similarity between all pairs of instances. The results obtained for all methods are very similar. The only significant difference is in the D_G results; for the average and complete linkages, although the 1&5 grouping is also the most likely, there are more 4&5 mergings than when using the single linkage.

Previous figures and tables give us an intuitive understanding of the qualitative performance of the algorithms and a quantitative analysis of the merging procedure. To go deeper we must analyze the KL values in detail. The box and whisker plot for the KL obtained by D_{prod} for the 9 Gaussians model for the 40 experiments is shown in Figure 4 as a representative example. The central mark indicates the median, the bottom and top edges of the box indicate the 25% and 75%, respectively, and crosses correspond to cases out of this interval.

As we already knew, the closest classes are 2 and 3. More importantly, we can see that the distance between the median values of the defective materials, i.e., 2&3, 2&4 and 3&4 is lower than between the homogenous class and any other of the materials, i.e., 1&2, 1&3 and 1&4. It means that the KL distance between the material with no defect and the rest of classes is large enough to allow a succesful hierarchy most of the times.

In some applications it is not so easy to get a large number of samples with a pre-known class (a large supervision ratio) to train the MoG, since the sample must be broken to obtain the ground truth about its state, with the asso-

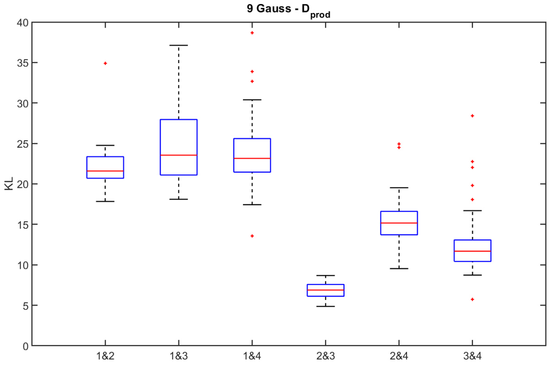

Fig. 4. KL distance for D_{prod} 9 Gaussians model.

ciated high economic cost. Therefore, it is important to know how robust the algorithms are to the quality of the samples used in the training of the MoG models. We can model this effect by reducing the supervision ratio. In Table III we show the first level percentage merging classes when the number of samples used in the EM algorithm with a known class is only 30%, instead of 90% .

	1&2	1&3	1&4	2&3	2&4	3&4
D_G	0	0	0	100	0	0
D_{min}	10.75	3.75	12.5	22	37.25	13.75
D_{Gold}	3.5	0	0	53.75	0.25	42.5
D_{prod}	0	0	0	100	0	0
D_{var}	3.5	0	0	52.75	0.25	43.5
D_{vub}	3	0	0	59.75	0.75	36.5
D_{MC}	3	0	0	70.75	1.25	25

TABLE III
First Level hierarchy mean results for all models in %. Supervision ratio 30%.

In spite of reducing significantly the number of samples with a pre-known class during the estimation of the conditional class probabilities, all methods but D_{min} are still able to separate class 1 from the other ones. Obviously, as more training samples have an unknown class, it is expected that the class condicional distributions are more prone to missclassifications during the test task. But a worse classification performance does not imply that the hierarchical clustering is worse. Only in 3% of the experiments the perfect class is closer to the one hole class for D_{Gold}, D_{var}, D_{vub} and D_{MC} approximations, while D_{prod} is not affected at all. In other words, to reduce drastically the quality of the training samples to learn the models has a greater impact at the classification level than at the hierarchy level, since most classification errors are merged in the first level hierarchy (class 2 and 3 are the closest ones); i.e., it is not a big problem not to have a large number of one hole and one crack samples, since they are going to be joined anyway in the first level hierarchy.

In Table IV we show the results for the second level hierarchy for the 30% supervision case. Compared to the results summarized in Table IV for the 90% supervision, the main difference is the increase in the percentage of 1&(2,3) cluster; it goes from 4.25% to 16% for the D_{prod} algorithm and from 11.5% to 19.5% for the other methods.

	1&4	1&(2,3)	(2,3)&4
D_G	0	91.75	8.25
D_{prod}	0	16	83.75
D_{Gold}	0	19.5	34.25
D_{var}	0	19.75	33
D_{vub}	0	20.25	39.5
D_{MC}	0	19.5	51.25

	1&2	1&(3,4)	2&(3,4)
D_{Gold}	0.5	0.5	41.5
D_{var}	0.5	0.5	42.5
D_{vub}	0.5	0.5	35.5
D_{MC}	0.5	0	24.5

TABLE IV
Second level hierarchy mean results in % for all models. Supervision ratio 30%.

The last issue to analyze is the influence of the granularity of the problem in the results, i.e., what happens if the starting number of classes is changed. In the 7-classes problem we split the hole and crack classes into subclasses taking into account the direction of the defect: homogeneous, X hole, Y hole, XY crack, ZY crack, ZX crack and multiple defects. Considering that the final goal is to separate perfect from damaged materials in the top level hierarchy, the dendrogram obtained must not include class 1 up to the top level of the hierarchy.

We run the EM based semisupervised algorithm again to obtain the new models. After that, the symmetric KL distance between the seven MoGs is obtained for the seven different KL distance approximations. As in previous experiments, we run 40 times the algorithms randomizing the training samples and test different models; in this case, the number of Gaussians for each model are $K = 3, 5, 7, 9, 11$.

We count the number of times that class 1 is in the top level of the hierarchy as a separate cluster (since this is a seven classes problem there are six levels in the hierarchy).

In Figure 5 we show the box and whiskers plot of the results for the five different models for each method. The value represents the percentage that class 1 was not merged until the last level of the hierarchy; i.e., in five previous levels, the six defective classes were correctly merged.

Again, D_{prod} obtains the best results; in almost 80% of the experiments, it was able to separate the homogeneous material from the six different defective materials in all levels of the hierarchy. In the cases where the homogeneous materials are merged in previous hierarchies, it would be better if the class 1 is not merged with a defective material in the first hierarchical clusterings. To show that this is

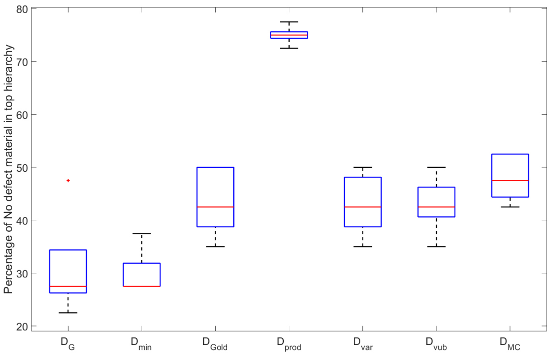

Fig. 5. Percentage of experiments where the class 1 (perfect material) is in the top level hierarchy.

the case, in Figure 6 we show the percentage where the perfect class were merged in the fifth level of the hierarchy. Adding these values to the ones in Figure 5 we can observe that most methods are not merging class 1 blocks until the last or penultimate levels of the hierarchy; i.e., the KL distance between the homogeneous specimens and the damaged ones are greater than between the damaged materials.

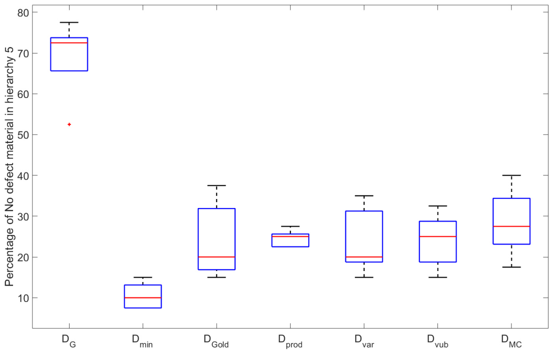

Fig. 6. Percentage of experiments where the class 1 (perfect material) is in the fifth level of the hierarchy.

Finally, in Figure 7 we plot a typical dendrogram for the D_{prod} method for the different agglomerative hierarchical clustering methods single, average and complete linkages, where we can observe that class 1 (no defect) is kept separated from the defective materials in all the clustering hierarchies up to the top level.

VI. Conclusion

We have presented a hierarchical clustering procedure based on the KL divergence between MoGs. Since we are using a probabilistic model, the KL divergence is the most natural way to measure the similarity between classes. We have introduced different ways to approximate the KL distance for the MoG distribution model.

We have applied it to the case of defective and non defective materials using IE techniques. The procedure can be extended to any other problem where we can obtain a MoG model for every class. We have seen that results depend on the assumptions of the KL approximation and if they fit the signals under analysis, i.e., the measurements. Since the MoG is a very flexible model for a large number of different measurements, the proposed methods can be applied to many different instruments and measures. The best results for our application are obtained by the D_{prod} approximation, being able to separate the homogeneous from any cluster of damaged materials most of the time in a very robust way (different class models, supervision ratio and number of classes).

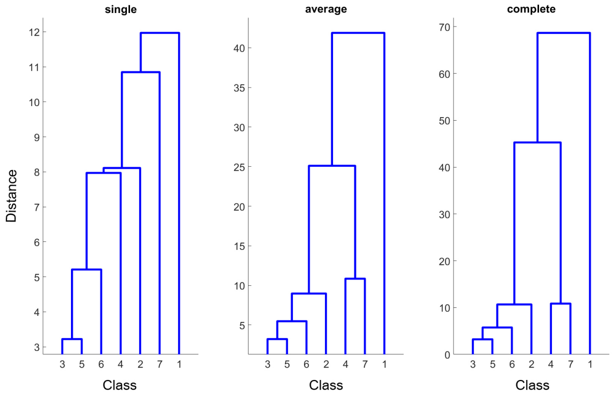

Fig. 7. Dendrogram for D_{prod} method and 5 Gaussians per class for the seven classes case. Left: single linkage; middle: average linkage; right: complete linkage.

Appendix

Mixture of Gaussians:	MoG
Kullback-Leibler Divergence	KL
Non Destructive Testing	NDT
Impact Echo	IE
Principal Component Analysis	PCA
Expectation Maximization algorithm	EM

References

[1] Sandeep Kumar Dwivedi, Manish Vishwakarma, and Prof.Akhilesh Soni, "Advances and researches on non destructive testing: A review," *Materials Today: Proceedings*, vol. 5, no. 2, Part 1, pp. 3690 – 3698, 2018, 7th International Conference of Materials Processing and Characterization, March 17-19, 2017.

[2] Rims Janeliukstis, Sandris Rucevskis, Miroslaw Wesolowski, and Andris Chate, "Experimental structural damage localization in beam structure using spatial continuous wavelet transform and mode shape curvature methods," *Measurement*, vol. 102, pp. 253 – 270, 2017.

[3] J. Hola, J. Bien, L. Sadowski, and K. Schabowicz, "Non-destructive and semi-destructive diagnostics of concrete structures in assessment of their durability," *Bulletin of the Polish Academy of Sciences Technical Sciences.*, vol. 63, no. 1, pp. 87–96, Apr. 2015.

[4] Mary J Sansalone and William B Streett, "Impact-echo. nondestructive evaluation of concrete and masonry," 1997.

[5] Nicholas J Carino et al., "The impact-echo method: an overview," in *Proceedings of the 2001 Structures Congress & Exposition*, 2001, pp. 21–23.

[6] A. M. Nicolson and G. F. Ross, "Measurement of the intrinsic properties of materials by time-domain techniques," *IEEE Transactions on Instrumentation and Measurement*, vol. 19, no. 4, pp. 377–382, Nov 1970.

[7] Alexander Gibson and John S Popovics, "Lamb wave basis for impact-echo method analysis," *Journal of Engineering mechanics*, vol. 131, no. 4, pp. 438–443, 2005.

[8] Oskar Baggens and Nils Ryden, “Systematic errors in impact-echo thickness estimation due to near field effects,” *NDT E International*, vol. 69, no. 0, pp. 16 – 27, 2015.

[9] Po-Liang Yeh and Pei-Ling Liu, “Application of the wavelet transform and the enhanced fourier spectrum in the impact echo test,” *NDT & E International*, vol. 41, no. 5, pp. 382–394, 2008.

[10] J. Sun, C. Yan, and J. Wen, “Intelligent bearing fault diagnosis method combining compressed data acquisition and deep learning,” *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 1, pp. 185–195, Jan 2018.

[11] J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, “Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network,” *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 2, pp. 257–269, Feb 2018.

[12] J. E. Garcia-Bracamonte, J. M. Ramirez-Cortes, J. de Jesus Rangel-Magdaleno, P. Gomez-Gil, H. Peregrina-Barreto, and V. Alarcon-Aquino, “An approach on mcsa-based fault detection using independent component analysis and neural networks,” *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 5, pp. 1353–1361, May 2019.

[13] Anna Hoła and Łukasz Sadowski, “A method of the neural identification of the moisture content in brick walls of historic buildings on the basis of non-destructive tests,” *Automation in Construction*, vol. 106, pp. 102850, 2019.

[14] Y. S. Cho, S. U. Hong, and M. S. Lee, “Multi sensor data fusion approach for automatic honeycomb detection in concrete,” *NDT & E International*, vol. 24, pp. 277–288, 2009.

[15] X.Y. Zhou, Z.F. Wang, and B.F. Yan, “Nondestructive testing method of grouting quality for prestressed pipe,” *China Journal of Highway and Transport*, vol. 24, pp. 64–71, 2011.

[16] Lukasz Sadowski and Jerzy Hoła, “Neural prediction of the pull-off adhesion of the concrete layers in floors on the basis of nondestructive tests,” *Procedia Engineering*, vol. 57, pp. 986–995, 2013.

[17] Jorge Igual, Addisson Salazar, Gonzalo Safont, and Luis Vergara, “Semi-supervised bayesian classification of materials with impact-echo signals,” *Sensors*, vol. 15, no. 5, pp. 11528–11550, 2015.

[18] I.T. Jolliffe, *Principal Component Analysis*, Springer Verlag, 1986.

[19] A. Malhi and R. X. Gao, “Pca-based feature selection scheme for machine defect classification,” *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 6, pp. 1517–1525, Dec 2004.

[20] Addisson Salazar, Luis Vergara, and Raúl Llinares, “Learning material defect patterns by separating mixtures of independent component analyzers from {NDT} sonic signals,” *Mechanical Systems and Signal Processing*, vol. 24, no. 6, pp. 1870 – 1886, 2010.

[21] Christoph Völker and Parisa Shokouhi, “Clustering based multi sensor data fusion for honeycomb detection in concrete,” *Journal of Nondestructive Evaluation*, vol. 34, no. 4, pp. 32, 2015.

[22] Sally C. Johnson, “Hierarchical clustering schemes,” *Psychometrika*, vol. 32, pp. 241–254, 1967.

[23] Addisson Salazar, Jorge Igual, Luis Vergara, and Arturo Serrano, “Learning hierarchies from ica mixtures,” in *Proc. IEEE International Joint Conference on Neural Networks IJCNN*, 2007.

[24] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 99th edition, Aug. 1991.

[25] JVDI Dhillon, “Differential entropic clustering of multivariate gaussians,” *Advances in Neural Information Processing Systems. The MIT Press*, vol. 19, pp. 337, 2007.

[26] Minh H. Do, “Fast approximation of Kullback-Leibler distance for dependence trees and hidden Markov models,” *IEEE Signal Processing Letters*, vol. 10, no. 4, pp. 115–118, 2003.

[27] Jacob Goldberger, Shiri Gordon, and Hayit Greenspan, “An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures,” in *In Proc. ICCV*, 2003, pp. 487–493.

[28] J.R. Hershey and P.A. Olsen, “Approximating the kullback leibler divergence between gaussian mixture models,” in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, april 2007, vol. 4, pp. IV–317 – IV–320.

[29] S.J. Julier and J.K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401 – 422, mar 2004.

[30] William HE Day and Herbert Edelsbrunner, “Efficient algorithms for agglomerative hierarchical clustering methods,” *Journal of classification*, vol. 1, no. 1, pp. 7–24, 1984.