HELICOPTER VIBRATION SENSOR SELECTION USING DATA VISUALISATION

Waljinder S. Gill, Ian T. Nabney

Nonlinearity and Complexity Research Group
School of Engineering and Applied Sciences
Aston University, Birmingham, UK

Daniel Wells

AgustaWestland Ltd.
Yeovil, UK

ABSTRACT

The main objective of the project[†] is to enhance the already effective health-monitoring system (HUMS) for helicopters by analysing structural vibrations to recognise different flight conditions directly from sensor information .

The goal of this paper is to develop a new method to select those sensors and frequency bands that are best for detecting changes in flight conditions. We projected frequency information to a 2-dimensional space in order to visualise flight-condition transitions using the Generative Topographic Mapping (GTM) and a variant which supports simultaneous feature selection. We created an objective measure of the separation between different flight conditions in the visualisation space by calculating the Kullback-Leibler (KL) divergence between Gaussian mixture models (GMMs) fitted to each class: the higher the KL-divergence, the better the inter-class separation. To find the optimal combination of sensors, they were considered in pairs, triples and groups of four sensors. The sensor triples provided the best result in terms of KL-divergence. We also found that the use of a variational training algorithm for the GMMs gave more reliable results.

Index Terms— Condition monitoring, vibration, signal processing, flight condition, sensor selection, KL-divergence, data visualisation

1. INTRODUCTION

The main objective of the project is to enhance the HUMS for helicopter airframes by analysing structural vibration. Past approaches to helicopter structural health monitoring with vibration data have used simple features with direct classifiers and had too many false positives to be practical. Thus there is a necessity to develop a more sophisticated approach to achieve a significant advance in predictive maintenance for helicopters, improving safety and reliability at less cost.

Vibration information during flight is provided by sensors located at different parts of the aircraft. Before structural health can be inferred, features (i.e. sensors and frequency bands) must be chosen which provide the best information on the state of the aircraft. These selected features will be then used to infer the flight modes and eventually the health and deviations from the normal state of the aircraft. The purpose of this paper is to propose and evaluate a novel selection process. The data provided by AgustaWestland Ltd. is continuously recorded vibration signals from 8 different sensors during flight. Each sensor measures the vibration in a particular direction at chosen locations on the aircraft. During test flights, the aircraft carries out certain planned manoeuvres: our goal is to infer flight condition from the vibration data only, since this will be required for a practical health monitoring system. The construction of flight state models from vibration data is completely novel; indeed, to our knowledge, there is no prior work on models of different flight modes for helicopters (as opposed to fixed-wing aircraft) and vibration analysis has mainly been used to monitor engine and transmission system condition, rather than airframe integrity.

Our approach is to study the (non-stationary) frequency information by applying a short-time Fourier transform. In this way, it is possible to detect certain signatures or intensities at fundamental frequencies and their higher harmonics. Many of the key frequencies are related to the period of either the main or tail rotor. The intensity at these frequencies is greater during certain periods of time and these periods can be associated to flight conditions and transition periods. Figure 1 shows the STFT and flight-state transitions. To understand the nature of the data and to extract more information, the high-dimensional STFT data was projected to a 2-dimensional manifold with the help of machine-learning algorithms. For this purpose we used the Principal Component Analysis (PCA), Generative Topographic Mapping (GTM) and a variant of GTM which supports simultaneous feature selection. GTM provided better structure than PCA in the visualisation therefore only GTM will be discussed. To determine which sensors are the most useful, we created an objective measure of the separation between different flight conditions in the visualisation space. The remainder of this paper is structured as follows: Section 2 describes the GTM and GTM-FS (the feature-selection variant); Section 3 defines the class-separation measure we have developed; Sec-

[†]Thanks to EPSRC and AgustaWestland Ltd. for industrial CASE (1000239X) funding.

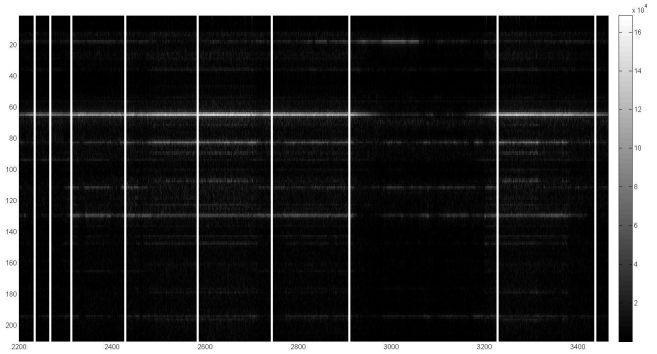

Fig. 1. STFT of sensor 7 with vertical lines at transitions between flight conditions. The x and y-axis provide the time [s] and frequency [Hz] respectively.

tion 4 describes the evaluation experiments that we have performed; finally, Section 5 contains the conclusions of the paper.

2. DATA VISUALISATION ALGORITHMS

2.1. Generative Topographic Mapping

The Generative Topographic Mapping is a non-linear probabilistic data visualisation method that is based on a constrained mixture of Gaussians, in which the centres of the Gaussians are constrained to lie on a two-dimensional space. GTM can be viewed as an improved version of the self-organising map (SOM) algorithm [1]. In this algorithm, data vectors $x_n \in \mathbb{R}^D$ in the D-dimensional data space are summarized by a set of reference vectors in a lower-dimensional space (usually in a regular grid in two dimensions to aid visualisation). Some of the drawbacks of this algorithm are: the absence of a cost function, lack of proof of convergence of the training algorithm, and the lack of density model [2].

In the GTM, a D-dimensional data point $(x_1, \ldots, x_D)$ is represented by a point in a lower-dimensional *latent* (hidden)-variable space $t \in \mathbb{R}^q$ (with $q < D$) so that it can be visualised in a lower-dimensional space q. The mapping between the latent and data space is non-linear and is achieved using a *forward* mapping function $x = y(t; W)$ which is then inverted using Bayes' theorem. This function (which is usually chosen to be a radial-basis function (RBF) network) is parameterised by a network weight matrix W. The image of the latent space under this function defines a q-dimensional manifold in the data space.

To induce a density $p(y|W)$ in the data space, a probability density $p(t)$ is defined on the latent space. The data is not expected to lie exactly on the q-dimensional manifold, a spherical Gaussian model with inverse variance β^2 is added in the data space so that the conditional density of the data is given by:

$$p(x|t, W, \beta) = \left\{ \frac{\beta}{\sqrt{(2\pi)}} \right\}^D \exp\left\{ -\frac{(\beta||y(t; W) - x||)^2}{2} \right\}. \tag{1}$$

To get the density of the data space, the hidden space variables must be integrated out:

$$p(x|W, \beta) = \int p(x|t, W, \beta) p(t)\ dt. \tag{2}$$

In general, this integral would be intractable for a non-linear model $y(t; W)$. Hence $p(t)$ is defined to be a sum of delta functions with centres on nodes $t_1, \ldots, t_K$ in the latent space:

$$p(t) = \frac{1}{M} \sum_{i=1}^{M} \delta(t - t_i). \tag{3}$$

This can be viewed as an approximation to a uniform distribution if the nodes are uniformly spread. Now equation (2) can be written as:

$$p(x|W, \beta) = \frac{1}{K} \sum_{i=1}^{K} p(x|t_i, W, \beta). \tag{4}$$

This is a mixture of K Gaussians with each kernel having a constant mixing coefficient $1/K$ and inverse variance β^2. The ith centre is given by $y(t_i; W)$. As these centres are dependent and related by the mapping, it can be viewed as a *constrained* mixture model. Provided $y(t; W)$ defines a smooth mapping, two points t^1 and t^2 which are close in the latent space are mapped to points $y(t^1; W)$ and $y(t^2; W)$ which are close in the data space.

The log likelihood for a dataset containing N points the following is given by:

$$\mathcal{L}(W, \beta) = \sum_{n=1}^{N} \ln \left\{ \frac{1}{K} \sum_{i=1}^{K} p(x_n|t_i, W, \beta) \right\}. \tag{5}$$

The parameters W and β can be found by searching for the maximum likelihood using an expectation maximization (EM) algorithm [2]. GTM has been shown to be an effective data visualisation method that outperforms linear algorithms such as Principal Component Analysis [3].

2.2. GTM-Feature Selection

In this paper there are about 102 frequency bands for each of the eight sensors: thus there are nearly 800 features altogether. Clearly, we would prefer to work in a lower-dimensional space while still representing all of the important information in the signal so as to avoid being distracted by irrelevant features and noise. So, to attain optimal results while visualizing the data, relevant features should be extracted from the data set. The GTM-FS model uses GTM-based

visualisation simultaneously with a measure of feature importance [4]. As discussed earlier in section 2, GTM uses a mixture of spherical Gaussians to model the data distribution. GTM-FS associates a variation measure with each feature by using a mixture of diagonal-covariance Gaussians. This assumes that the features are conditionally independent. The probability density function is given by:

$$p(X_n \mid K, \theta) = \sum_{k=1}^{K} \frac{1}{K} \prod_{d=1}^{D} p(x_{nd} \mid \theta_{kd}), \tag{6}$$

where K is the number of mixture components, $p(x_{nd} \mid \theta_{kd})$ is the probability density function for the dth feature for the kth component, and $\theta_{kd} = \{y(t; W), \beta\}$ with β being the corresponding variance.

The dth feature is irrelevant if its distribution is independent of the component labels, i.e. if it follows a common density, denoted by $q(x_{nd}|\lambda_d)$ which is defined to be a diagonal Gaussian with parameters λ_d. Let $\{\Psi = (\psi_1, \ldots, \psi_D)\}$ be an ordered set of binary parameters such that $\psi_d = 1$ if the dth feature is relevant and $\psi_d = 0$ otherwise. Now the mixture density is:

$$p(x_n|\Theta) = \sum_{k=1}^{K} \frac{1}{K} \prod_{d=1}^{D} [p(x_{nd} \mid \theta_{kd})]^{\psi_d} [q(x_{nd} \mid \lambda_d)]^{(1-\psi_d)}. \tag{7}$$

where parameters: $\{K, \theta, \psi\}$ are summarized by Θ. The value of the feature saliencies is obtained by firstly treating the binary values in the set Ψ as missing variables in the EM algorithm (for structure refer to [4]) and then defining it by a probability p_d that a particular feature is relevant ($\psi_d = 1$). Cheminformatics data from was analysed in [4] using GTM, GTM-FS and SOM. In GTM and GTM-FS, the separation of data clusters was better while GTM-FS showed more compact results because the irrelevant features were projected using a different distribution. In addition to the projection, the feature saliency plot derived from GTM-FS showed the the feature saliencies which gave an indication of relevant and important features whereafter it can be used in selecting features which are above a certain high saliency.

We have applied this approach to determine the most important frequency bands in the STFT data. Figure 2 shows the feature saliencies for each frequency band in the STFT dataset for a single sensor. A line is drawn at 0.7 and the features which have saliencies above this line were selected. This threshold has been chosen somewhat arbitrarily, and the issue will be revisited in the future.

3. CLASS SEPARATION METRIC

The next stage of our analysis is to develop a numeric measure of the separation of classes of flight conditions. Our method is to use the Kullback-Leibler divergence between the probability distribution of each class.

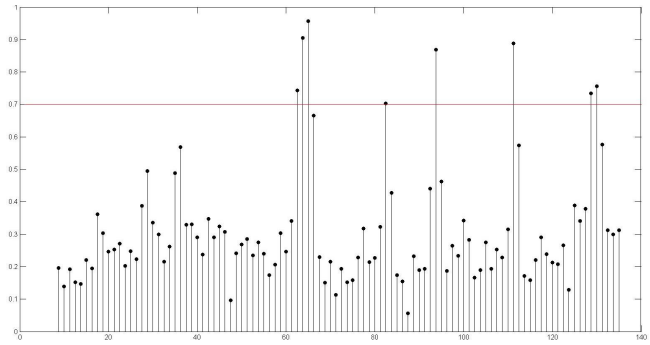

Fig. 2. GTM-FS feature saliencies for frequency bands [Hz] for sensor 7 for a single flight.

3.1. Kullback-Leibler divergence

Consider visualisation plots of data drawn from several different flight conditions. Our goal is to compare visualisation plots that are based on different subsets of sensors and to choose the subset that provides the best separation (in latent space) between the flight conditions. We aim to do this in latent space rather than the original data space, since the ability to visualise the results helps to interpret them. In a good visualisation, each flight condition corresponds to a distinct cluster of data, but to compare the plots objectively, we need a quantitative measure of class separation. For simplicity, suppose that there are two clusters representing two distinct classes. The Kullback-Leibler divergence is a measure of the divergence between two probability distributions P and Q [5]: P and Q can be chosen to be models of the probability density of each of the two classes. The KL-divergence is defined as:

$$D_{KL}(P||Q) = \sum_{n} P(x_n) \log \frac{P(x_n)}{Q(x_n)}. \tag{8}$$

If the data contains more than two classes, the KL-divergences of all possible class pairs (in both orders, since KL-divergence is not symmetric) are added up to calculate the overall separation. The higher the KL-divergence, the more separated the classes are from each other.

To calculate the KL-divergence, it is necessary to fit a probability density model to each class: we have chosen to use a mixture of Gaussians. A class label is assigned to each point according to the flight condition at that time during a flight. The time points for the different conditions and the transitions between them were provided by AgustaWestland for a number of test flights. We are particularly interested in the transitions between flight conditions as these are likely to excite unusual vibration modes. For this, a time period taken before and after a transition is also analysed to see if there is any transient behaviour. However, it is possible that while labelling classes, two different labels (before or after transition) are associated with the same flight conditions data at differ-

ent time periods. To explain this, for example, suppose that a transition from 60 to 80 knots forward speed begins (class 2) at 2200 and ends at 2300 seconds and data is selected for a few seconds before (class 1) and after (class 3). To this dataset, we add another transition 80 to 100 kts (class 5) between 2400 and 2500 seconds with a few seconds before (class 4) and after (class 6) transition. So, the classes which correspond to the same flight condition (80 kts) are 3 and 4. We want to calculate the separation of different flight conditions rather than different classes with the same flight conditions. For this reason, classes representing the same flight condition were grouped together. We also analysed whether our results would generalise to different test flights, and so grouped together conditions across multiple flights. We want the members of

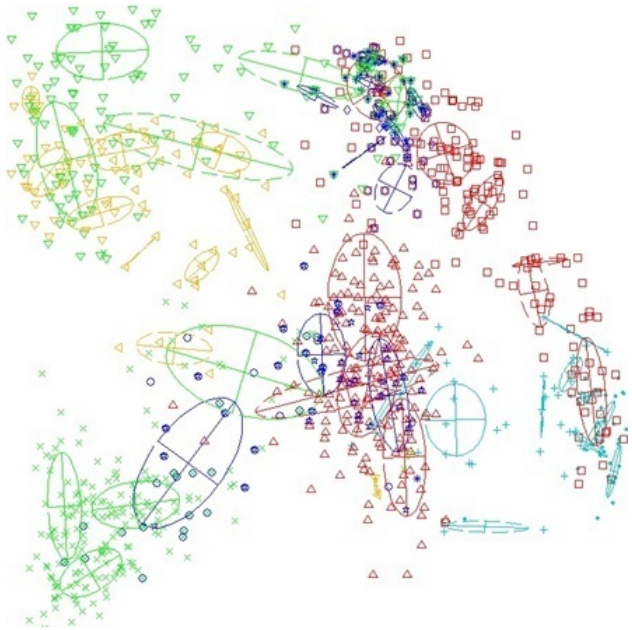

Fig. 3. GTM-FS visualisation of two flights with GMM applied with a fixed number of kernels (10) for signals 1, 2 and 7 (after frequency feature selection). Markers with the same colour are drawn from a single flight condition group. Each ellipse denotes a kernel of the GMM used to fit a cluster. Ellipses with dashed boundaries have a small mixing coefficient. KL-Divergence was 338.

the groups to lie as close as possible to each other: Figure 3 shows a typical result. The plot contains classes representing 60 kts forward speed, 60–80 kts transition, 80 kts forward, 80–100 kts transition, and finally 100 kts. These classes are spread across the visualisation space in a logical order (this is easier to see with the visualisation tool than in the plots in this paper). This figure shows that our approach can be made to work effectively. However, there is a difficulty. We have chosen the number of kernels in the GMM arbitrarily (which may cause over-fitting), and also the EM algorithm is susceptible to being trapped in local minima. In the next section we discuss how variational Bayesian methods can be used to address both of these issues.

3.2. Variational mixture of Gaussians

To make the calculation of the KL-divergence more robust, we modified the algorithm that we used to fit the density model to each class. A variational Bayesian Gaussian Mixture model automatically adjusts the number of components to avoid over-fitting [6]. The result of applying this model

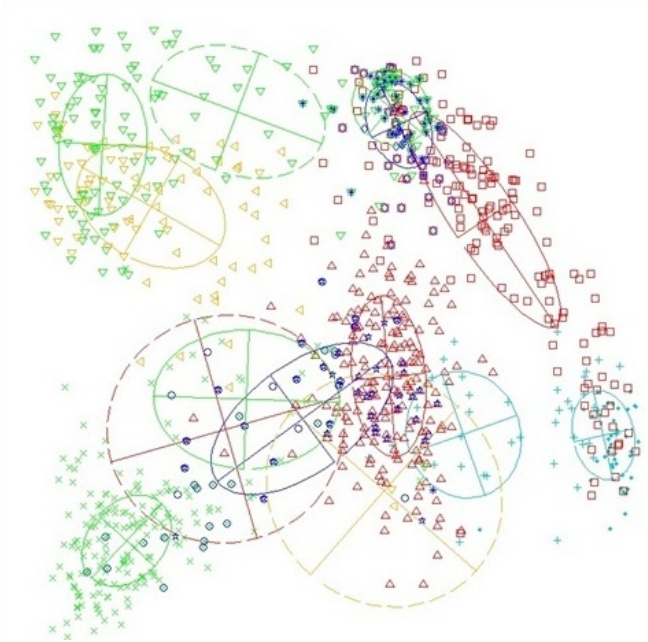

Fig. 4. GTM-FS visualisation of two flights with VMM applied optimal kernels for signals 1, 2 and 7 (after frequency feature selection). KL-Divergence was 133. For other details, see the caption to Figure 3.

is shown in Figure 4. It can be seen that the number of kernels is much lower than in Figure 3 and is different for each group. Thus we have confidence that the optimal number of components has been selected and the calculation of the KL-divergences has been improved. We now give a brief summary of the variational mixture model.

A Bayesian model is constructed for a mixture of Gaussians, in which the mixing coefficients are made random variables. A latent variable s_{in} is provided for each data point and component [7]. For example, if the data point x_n is generated by the ith Gaussian of the mixture model, s_{in} is 1, and 0 otherwise. The conditional distribution of s is given by:

$$P(s|\pi) = \Pi_{i=1}^{K} \Pi_{n=1}^{N} \pi_i^{s_{in}}, \tag{9}$$

where π_i is the ith mixing coefficient. The likelihood of the

model is given by:

$$P(W|\mu,\Sigma) = \Pi_{i=1}^{K}\Pi_{n=1}^{N}\mathcal{N}(x_n|\mu_i,\Sigma_i^{-1})^{s_{in}}, \quad (10)$$

where μ_i, Σ_i are the means and inverse covariance matrices of the ith Gaussian component. In order to complete the Bayesian model, priors are needed over the latent space variable s, means and inverse covariance matrices.

$$P(\mu) = \Pi_{i=1}^{K}\mathcal{N}(\mu_i|0,\alpha I), \quad (11)$$

where $\mathcal{N}$ is the normal distribution, α is a small valued fixed parameter which relates to prior over μ and I is identity matrix.

$$P(\Sigma) = \Pi_{i=1}^{K}\mathcal{W}(\Sigma_i|v,V^{-1}), \quad (12)$$

where $\mathcal{W}$ is the Wishart distribution, v is the number of degrees of freedom and V is the $D\times D$ scale matrix for the prior over Σ.

$$P(s) = Pi_{i=1}^{K}\Pi_{n=1}^{N}\pi_i^{s_{in}}. \quad (13)$$

The likelihood of the dataset $\mathcal{D}$ is given by:

$$P(D,\theta) = \Pi_{n=1}^{N}P(x_n|\mu,\Sigma,s)P(s)P(\mu)P(\Sigma). \quad (14)$$

All the parameters (s, μ, Σ) are summarized as θ. In order to select a Bayesian model, θ has to be integrated out. The distribution function of the data $P(D)$ (the evidence) is then maximized with respect to the mixing coefficients π_i. After the maximization, any mixture coefficients that degenerate to 0 are removed and others are kept.

Unfortunately, integrating the likelihood with respect to θ is not tractable. For this reason a variational approximation approach has been developed [7]. The assumption is made that the variational distribution can be factorized over each group of parameters.

$$Q(s,\pi,\mu,\Sigma) = Q_s(s)Q_\mu(\mu)Q_\Sigma(\Sigma). \quad (15)$$

Now the distribution Q that best approximates likelihood $P(D,\theta)$ can be computed as follows:

$$Q_s(s) = \Pi_{i=1}^{K}\Pi_{n=1}^{N}p_{in}^{s_{in}}, \quad (16)$$

where p_{in} are the variational parameters of the Q distribution.

$$Q_\mu(\mu) = \Pi_{i=1}^{K}\mathcal{N}(\mu_i|m_\mu^{(i)},|\Sigma_\mu^{(i)-1}), \quad (17)$$

$$Q_\Sigma(\Sigma) = \Pi_{i=1}^{K}\mathcal{W}(\Sigma_i|v_\Sigma^{(i)},V_\Sigma^{(i)-1}), \quad (18)$$

Once Q has been calculated we can approximate the lower bound of the log-likelihood. The mixing coefficients that maximize the lower bound are given by:

$$\pi_i = \frac{1}{N}\sum_{n=1}^{N}p_{in}. \quad (19)$$

In order to get the optimal number of components in a mixture, the variational approximation and the update of the mixing coefficients that maximize the lower bound are iterated alternately until the lower bound converges.

4. RESULTS

We evaluated our approach to sensor selection by applying it to a dataset that combined two test flights. The aim of the experiment was to select the best group of sensors to model flight state and transitions. We started by computing the KL-divergences for visualisation plots based on each sensor individually. We then combined the best four senors in pairs and repeated the measurement for each pair. This continued for triples and all four top sensors. This greedy search can be carried out in a reasonable time scale ($\leq$ 72 seconds) as shown in Figure 5. The time scale includes the time taken to calculate the STFT, training the data set, applying GMM's to groups of classes in the visualisation and finally evaluating the KL-divergences between these groups. As seen in the Figure, the time taken for each process increases when adding sensor information which is unsurprising since the computational effort increases with increasing dimensionality. Overall, as we can see, the VMM process takes a bit more time than GMM process as the alternative iteration for finding minimum lower bound as explained in section 3.2 consumes much of the processing time.

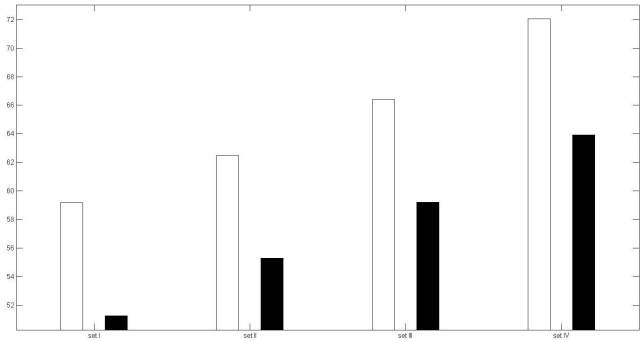

Fig. 5. Average performance (in seconds) of the GMM (black bars) and VMM (white bars) for 4 different data sets. Set I has a single, set II two, set III three and set IV has four sensor(s) information from two flights combined respectively.

Sensor No.	GMM KL-D	VMM KL-D
1	160	87
2	190	95
3	110	75
4	130	78
5	85	31
6	220	104
7	255	113
8	82	36

Table 1. KL-divergences for single sensors.

Sensor pair	GMM KL-D	VMM KL-D
1-2	265	120
1-6	225	135
1-7	215	127
2-6	185	118
2-7	300	122
6-7	198	171

Table 2. KL-divergences for sensor pairs.

Sensor triples	GMM KL-D	VMM KL-D
1-2-6	265	122
1-2-7	338	133
1-6-7	288	177
2-6-7	312	154

Table 3. KL-divergences for sensor triples.

Sensors	GMM KL-D	VMM KL-D
1-2-6-7	317	173

Table 4. KL-divergences for 4 sensors together.

Table 1 shows the KL-divergences for each individual sensor. The top four sensors were: 1, 2, 6 and 7. The KL-divergences for these sensor pairs and triples are shown in Table 2 and 3 respectively. From the analysis, it has been found that the group of three sensors (triples) from the top selected provide the best KL-divergence when compared to individual, pairs or 4 sensors together. The four selected sensors provide the best KL-divergence results as compared to pairs and triples with other sensors (3, 4, 5 and 8). To confirm this, KL-divergences of all possible sensor pairs and triples have been computed for single and muliple flights. It was found that no pair or triple had a higher KL-divergence than the selected four sensors.

The values of the KL-divergence computed using GMM and VMM are different: this is caused by the fact that the VMM typically uses fewer components in the mixture model. More fundamentally, it is noticeable that the sensor importances using both methods for computing KL-divergence are not the same. To investigate this, we carried out experiments to calculate the KL-divergence with both the GMM and the VMM. For a given visualisation plot, each method was run ten times with different initial conditions. We found that the KL-divergence values for the VMM were much more consistent over these replicates than for the GMM. The variability of the KL-divergences of Gaussian mixture model is higher (standard deviation $\sim$ 30-50) as compared to variational Gaussian mixture (standard deviation $\sim$ 2-5). For this reason, the sensor triple 1, 6 and 7 will be used for further analysis.

5. CONCLUSIONS

We have developed a feature selection procedure that is based on visualisation of data, feature saliency (for selecting frequency bands for a single sensor), and a KL-divergence metric to compare class separation. The selection procedure showed that sensor triples gave the best possible KL-divergences for two flights combined indicating better inference for flight conditions, maneuvers and health of the aircraft. This information is valuable since it enables us to work in a much lower-dimensional feature space which is more compuational efficient than the original data (which, using all frequency bands and sensors, would have been nearly 800-dimensional) .

Future work on this methodology will include a more systematic approach to setting the threshold for feature saliency, evaluation on a larger range of flight conditions and test flights, and consideration of generalisation to unseen flight data.

6. REFERENCES

[1] Teuvo Kohonen, "Neurocomputing: foundations of research," chapter Self-organized formation of topologically correct feature maps, pp. 509–521. MIT Press, Cambridge, MA, USA, 1988.

[2] Christopher M. Bishop, Markus Svensn, and Christopher K. I. Williams, "GTM: The generative topographic mapping," *Neural Computation*, vol. 10, pp. 215–234, 1998.

[3] Fabian Lopez-Vallejo, Adel Nefzi, Andreas Bender, John R. Owen, Ian T. Nabney, Richard A. Houghten, and Jose L. Medina Medina-Franco, "Increased diversity of libraries from libraries: chemoinformatic analysis of bisdiazacyclic libraries," *Chemical biology and drug design*, vol. 77, no. 5, pp. 328–342, 2011.

[4] Dharmesh M. Maniyar and Ian T. Nabney, "Data visualization with simultaneous feature selection," in *Computational Intelligence and Bioinformatics and Computational Biology, 2006. CIBCB '06. 2006 IEEE Symposium on*, Sept. 2006, pp. 1–8.

[5] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 2006.

[6] Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[7] Christopher M. Bishop and A. Corduneanu, "Variational bayesian model selection for mixture distribution," *Artificial Intelligence and Statistics*, pp. 27–34, 2001.