Event Detection Methods for Multi-Sensor CAE-Data

R. Iza-Teran[1], D. Steffes-lai[1] , Anh-Duy Pham, M. Pathare[1,] J. Garcke[1]

[1]Fraunhofer SCAI, Schloss Birlinghoven 1, 53757 Sankt Augustin

Abstract

Virtual product development especially in car development requires the evaluation of multiple sensors signals in the simulations as one of the tasks; the sensor data is also needed for comparison with the real product. Comparing many virtual sensors manually from many simulations turns out to be a time consuming and challenging task. We propose a methodology and workflow setting that address this challenge, allowing a similarity comparison of hundreds of sensors from hundreds of simulations detecting similar events (clusters) or very different behavior as outliers. The approach uses a method of dimensionality reduction combined with different type of clustering methods including hierarchical clustering. The dimensionality reduction reduces the virtual sensor data information such that a visual comparison of thousand sensor signals can easily be performed in 3D, the hierarchical clustering on the other hand allows a localized comparison of sensor signals. The approach is demonstrated using binout Ls-Dyna data from a frontal crash example with many model variants containing many sensor data per simulation as well as for head impact computation.

1 Introduction

Product development especially in the automotive domain relies on the evaluation of the time dependent results like acceleration, velocity and deformations, those values are used further for the evaluation of derived quantities like the head injury index (hic) which is a measure of the likelihood of head injury arising from an impactor. Another example is the crash-pulse which is the deceleration of the vehicle measured during crash. The shape, slope, maximum deceleration and duration of the crash-pulse provides significant information over the occupant motions during a crash phase and provides a measure of crash severity. Normally for single product variants, several hundred of them at different points in the structure are evaluated. A comparison of several product variants providing information about similarities and outliers is challenging. A further complication arises from the temporal dependency wherein specific regions of a signal have to be compared.

We propose a methodology and workflow setting that address this challenges, allowing a similarity comparison of hundreds of sensors from hundreds of simulations detecting similar events or very different behavior as outliers. The approach uses a method of dimensionality reduction combined with clustering. The dimensionality reduction reduces the virtual sensor data information such that a visual comparison of many sensor signals can be easily performed in 3D, the hierarchical clustering allows also a flexible comparison of sensor signals. The approach is demonstrated for variants of a frontal crash simulations and for head impact evaluation

Note that **`*KEYWORDS`** may be written with the associated template “keyword”.

2 Method

A dimensionality reduction is used as a pre-processing step for a workflow since it allows a more easier clustering and outlier detection, several approaches for dimension reduction are made available from standard tools, namely PCA, t-SNE, Diffusion Maps, LTSA, etc.. In addition we have implemented a new method for dimensionality reduction using as a basis the approach described in [1], that has been successfully used for data analysis in crash simulation [1], [2]. This approach is not directly applicable to our setting, nevertheless we discuss how the same mathematical principles can be used.

2.1 Laplace-Beltrami Operator on Curves

The methodology in [1] relies on being able to compute an approximation to a Laplace Operator on a mesh and using the mathematical property that this operator is under certain conditions invariant to deformations of the surface that preserve distances along the shape. This means that once the operator is computed, the eigenvectors of it are a convenient basis for all deformations by using the projection coefficients to the basis and the property that the deformation main changes are well described by very few coefficients. In the space of the coefficients, the so called spectral space, clustering and outlier identification is much easier as the dimensionality of the space has been reduced to very few dimensions.

Generalizing the mentioned approach from 3D surfaces to curves is certainly possible by measuring distances along the curve, this is called the segment length. An operator similar to a Laplace-Operator can be build using such distances and we further assume that the curves deformations (or changes) are distance preserving as before. The resulting eigenvector basis computed from the operator can again be used as basis wherein many curves from sensor data can be projected. Main changes are concentrated in very few coefficients making it convenient for using as dimensionality reduction method. See Figure 1 for a simplified representation of the approach.

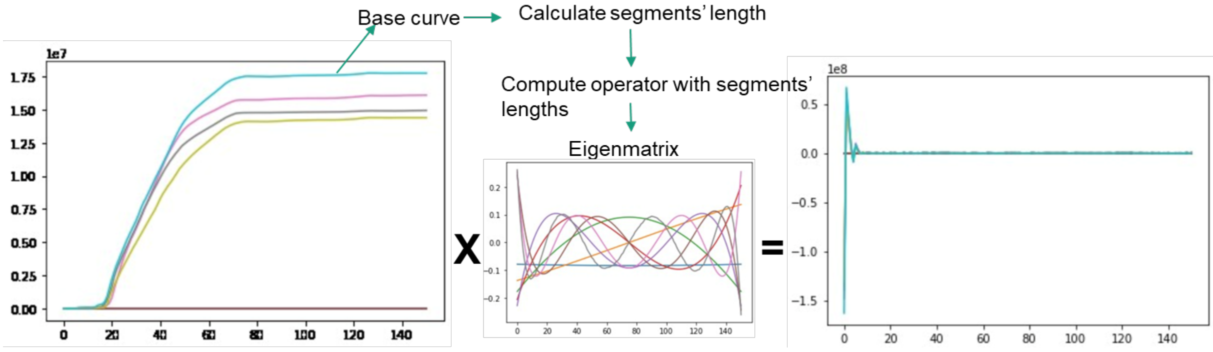


Fig.1: Simplified description of method of dimension reduction.

As seen in Figure 1 on the right, only the first projection coefficients have higher values, the curve decays very quickly to zero. It means that one case use those few coefficients as a compact features that characterize the shape of the curve. Clustering and outlier detection are the simplified due to this property. We call our method Chord Embedding Analysis (CEA).

Methods of dimensionality reduction always depends on some training data, for example for PCA as more data is available, one can recover all variance of the given data in very few projection coefficients. This is very convenient but as new sensor data become available, a re-computation of the covariance matrix is necessary. Other methods are not able to reconstruct the curves based on few coefficients, an important property from the point of view of data reduction. The CEA method does have this property so we describe its use in applications in section 4.

3 Workflow

For the implementation of clustering and outlier detection, a workflow consisting of two phases were developed. The first one is a batch non-interactive process followed by an interactive one.

3.1 Batch phase

This phase handles the sensor data, in our case this is a binary binout file. Routines for reading the information from those files are readily available in python. The next step is the computation of dimensionality reduction. As for this work we use several methods of dimension reduction as available in the tool sklearn.manifold, some of them are PCA, t-SNE, Isomap, Diffusion Maps, LTSA in addition to our CEA method.

Depending on the method used, a set of reduced coordinates or embedding's are obtained. We use up to 3 dimensions of them to get a visualization for the interactive phase. The separation in two phases follows the standard evaluation of simulations in a HPC computing cluster. As new simulations are performed, the corresponding sensor information can be processed and the low dimensional coordinates be updated. A data handling step will just compress all analysis results and transport them to a client computer. This steps in the workflow could be easily implemented in a Simulation Data

Management (SDM) system (see Figure 2 for a schematic representation of the workflow). Ideally the raw data should not be transported for the interactive phase but this depends on the method of dimension reduction chosen as mentioned.

3.2 Interactive phase

Once the non-interactive phase is finished, the dimension reduction method will contain only a reduced set of vectors or features that represents the raw data. This data can be transferred or be used in a post-processing phase where a clustering and outlier detection module can be applied. The result will be the levels of the clusters and the outliers. For visualization a dash web visualization can be used or a python GUI be developed to interactively explore the clusters and outliers (see right part of Figure 2). Each point in the reduced representation will represent one of the sensor signals. Depending on the method of dimensionality reduction this image can be generated on the client without generating an image file. For example this is the case for PCA as well as for our CEA method, if the eigenvectors of the PCA, CEA are saved, then a reconstruction of the sensor signal is possible using very few coefficients.

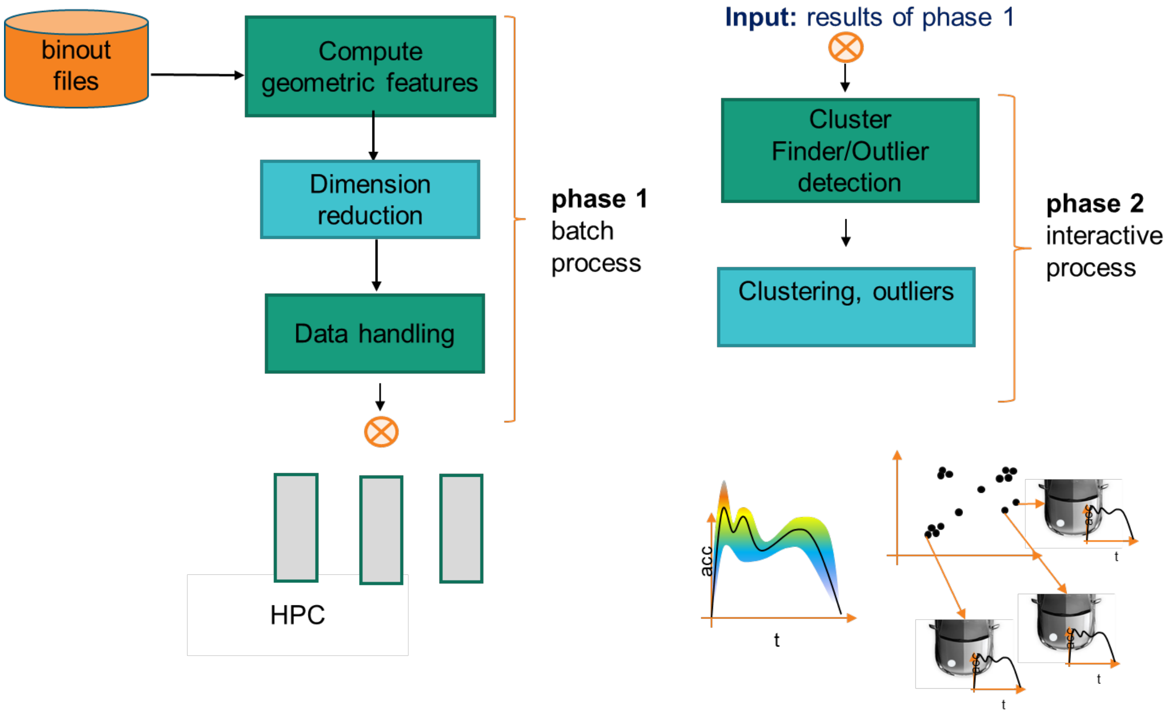


Fig.2: Workflow scheme for the analysis of sensor data.

3.2.1 Clustering

Clustering data in useful ways is challenging. We propose to use two clustering methods that can be used for different goals. A density based clustering can be used to obtain clusters and outliers automatically and a hierarchical clustering can be used to adaptively find more or less clusters.
Specifically DB-SCAN is a density-based clustering that identify distinctive groups/clusters in the data automatically, based on the idea that a cluster in data space is a contiguous region of high point density, separated from other such clusters by contiguous regions of low point density.
Hierarchical clustering starts by treating each observation as a separate cluster. Then, it repeatedly executes the following two steps: (1) identify the two clusters that are closest together, and (2) merge the two most similar clusters. This iterative process continues until all the clusters are merged together. The main output of Hierarchical Clustering is a dendrogram, which shows the hierarchical relationship between the clusters.

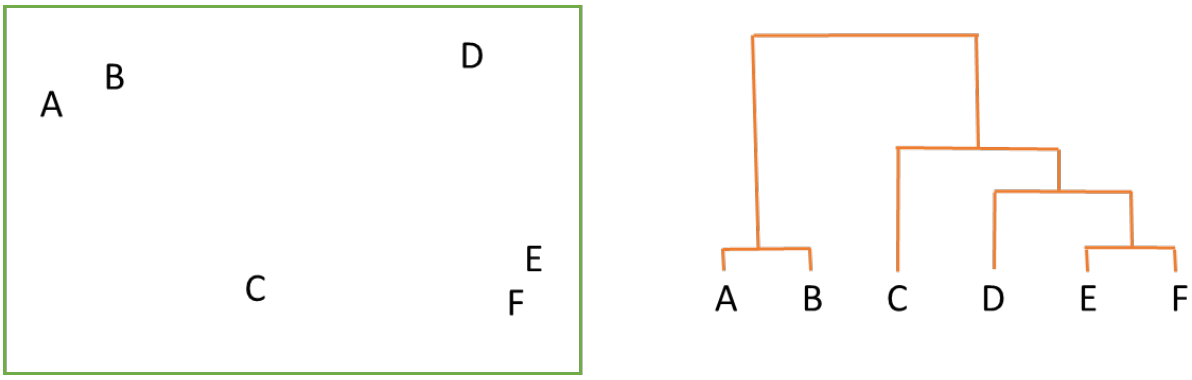


Fig.3: Schematic representation of hierarchical clustering, points are represented as letters a dendrogram is shown on the right figure.

4 Results and Discussion

We have applied the workflow, as described, for data from several crash simulation results. We present here results for frontal crash energy curves and head impact accelerations.

4.1 Frontal crash simulation

For the frontal crash a total of 210 simulations of a truck were run [3]. We apply the workflow to the energy curves of around 200 parts. That means, one has to compare 42000 curves.
Following the steps in the workflow, the binout data for all simulations for all the parts are read and a method of dimension reduction is applied. We chose to use our own method due to the advantages mentioned in section 2. The 3D coordinates obtained for the low dimensional representation can be clustered in phase 2. Figure 4 represents a snapshot of the interactive phase wherein the energy curves clusters can be visualized.

For the acceleration, Figure 5 shows the acceleration of all simulations in raw format. An evaluation of events in the car crash simulation is very difficult taking the raw data as it is. From the binout we read the acceleration curves and apply a Butterworth filter with cutoff frequency Wn = 0.01 and apply the dimension reduction of phase 1 to the data. After obtaining a low dimensional representation, clusters are evaluated in the clustering step. For this case two clusters are obtained that can be correlated with the deformations of the crash simulation. That is, according to this analysis, a cluster of accelerometers measurements can be ordered to a specific deformation mode. This result can be seen in Figure 6.

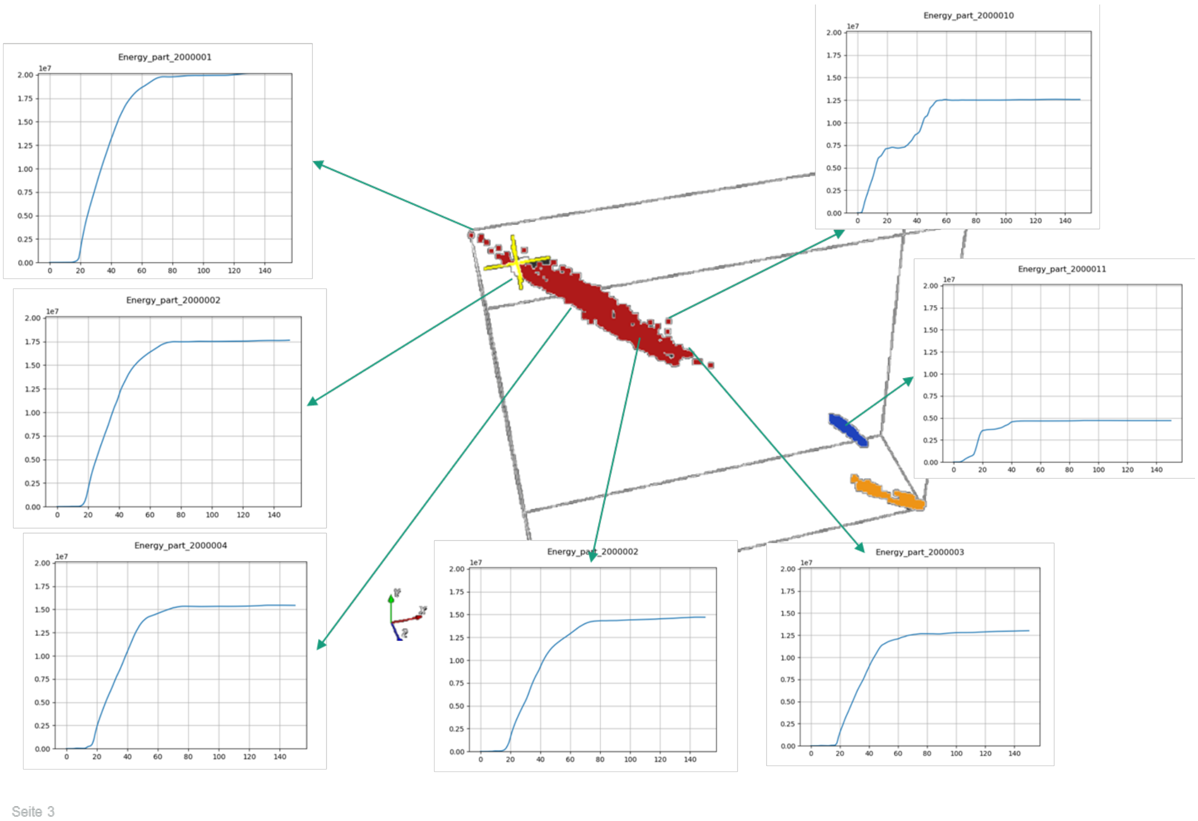


Fig.4: Clustering of energy curves from 210 simulation for 200 parts (showing only the clusters with highest energy for clarity)

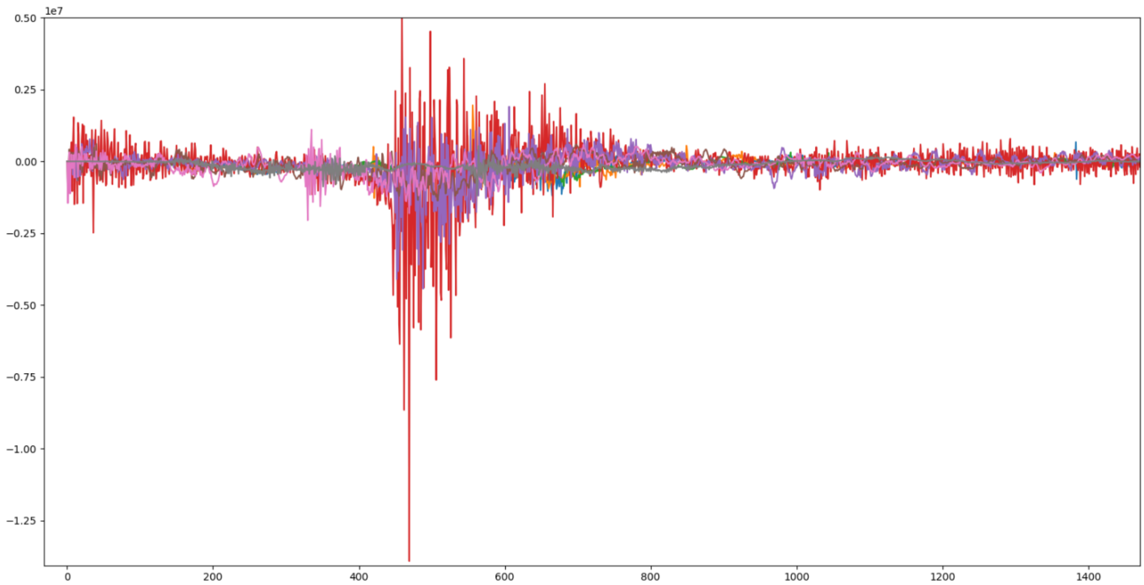


Fig.5: Acceleration curves for 210 simulations as read from the binout files

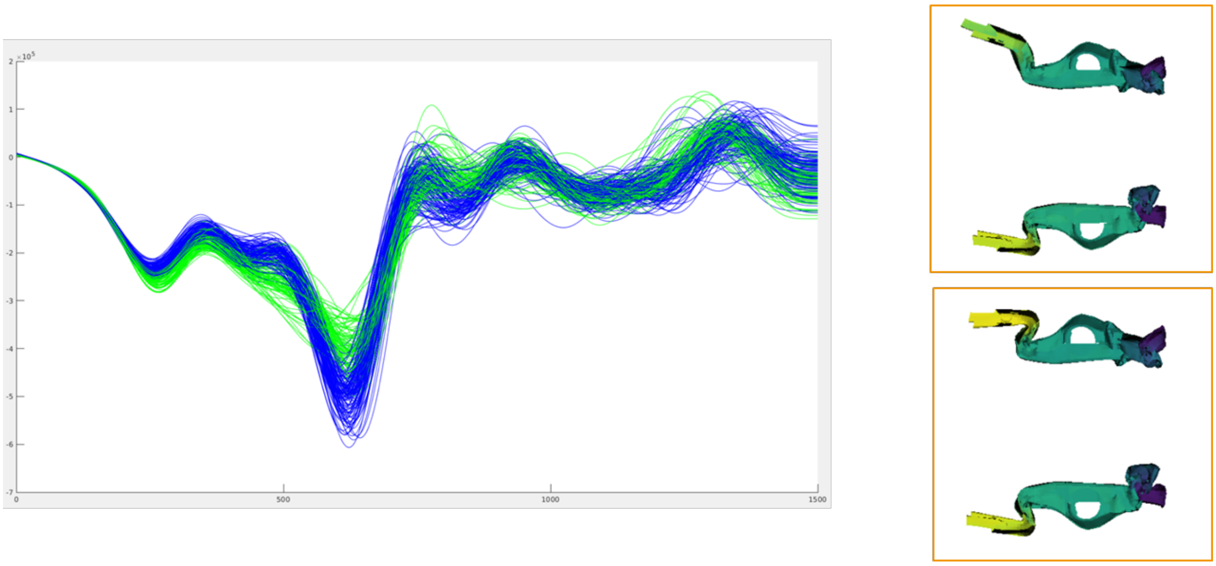
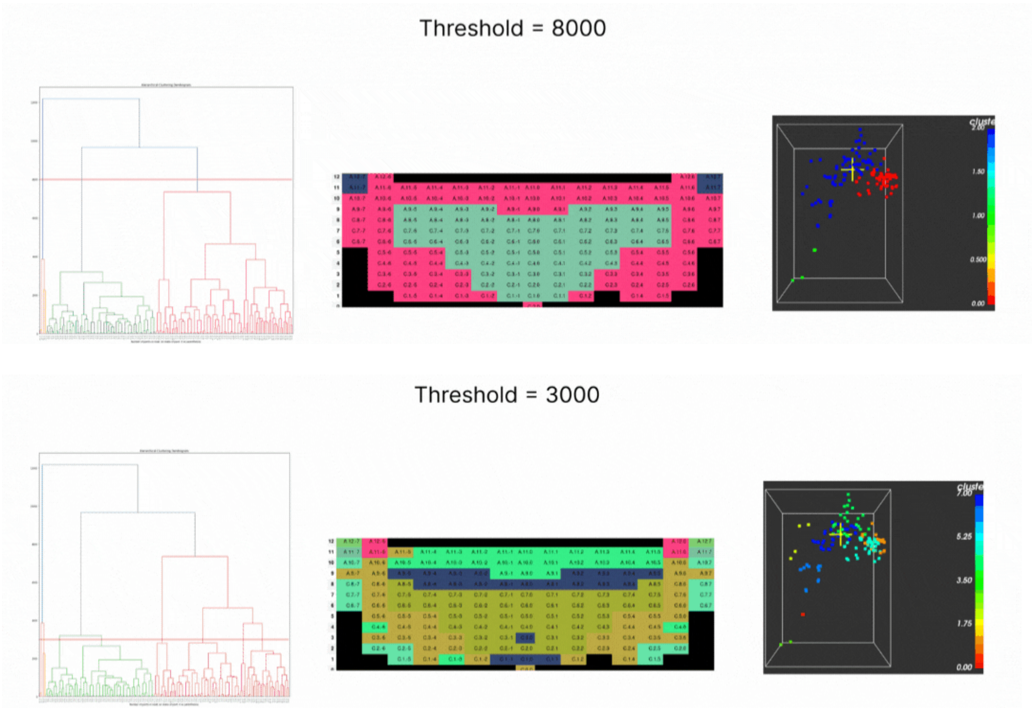


Fig.6: Clustering of filtered acceleration curves for 210 simulations showing two different deformation modes

4.2 Head Impact simulation

Lsdyna simulations for the evaluation of 180 head impact points have been used for demonstrating the use of the workflow. As before in the batch phase our own method of dimension reduction has been used.



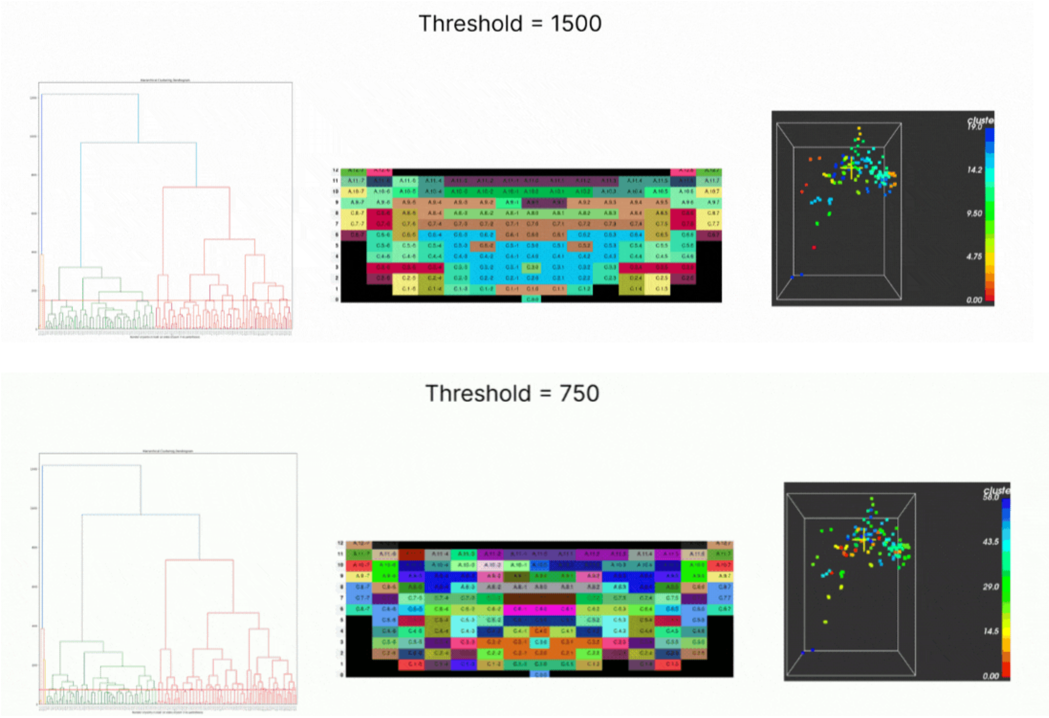


Fig.7: Hierarchical Clustering of head impact curves showing: left the dendrogram, middle the impact points and right clustering in low dimensional space

Hierarchical clustering allows to group similar acceleration curves according to a threshold that can be adapted as needed. The higher the threshold the coarser the higher the number of curves that are grouped in a cluster. On the left of Figure 6, the dendrogram is shown for different positions of the threshold horizontal line. The intersection of this line with the corresponding vertical ones determine the simulations that are considered to be part of a cluster. On the right of Figure 6, the low dimensional representation of the curves is shown as points wherein the colour corresponds to a given cluster. In the middle, the evaluation points for head impact are shown also with a colour representing the clusters. This type of visualization is useful to identify specific signals on specific locations of the evaluation grid.

5 Summary

Clustering and outlier detection of many sensor data for automotive applications has been shown using a workflow with a non-interactive and interactive part. In the interactive part only a compact representation of the analysis results is transferred for convenient processing at the client side. A method for dimensionality reduction is also presented. Density based clustering and hierarchical clustering are applied to the reduced features. The workflow use is presented for two applications in automotive one for frontal crash and the other one for head impact evaluation.

6 Literature

[1] Iza-Teran R. and J. Garcke. “A geometrical method for low-dimensional representations of simulations”. SIAM/ASA Journal on Uncertainty Quantification, 7(2):472-496, 2019.

[2] Garcke, J, Iza-Teran, R. (2017). “Machine learning approaches for data from car crashes and numerical car crash simulations”, NAFEMS World Congress 2017

[3] National Highway Traffic Safety Administration, https://www.nhtsa.gov/crash-simulation-vehicle-models