July 2020

GENERATION OF SYNTHETIC CRASH DATA AND APPLICATION OF CRASH ALGORITHMS WITH DEEP LEARNING METHODS

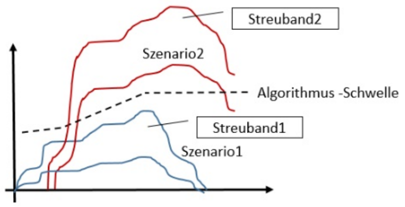
Verena Blunder
Bertrandt Ingenieurbüro GmbH

GENERATION OF SYNTHETIC CRASH DATA AND APPLICATION OF CRASH ALGORITHMS WITH DEEP LEARNING METHODS

Technical task:
FEM simulations are not able to reproduce high frequency components in the crash signal sufficiently well, so that FEM simulations can only be used for the application process to a limited extent. Usually, so-called worst case tests are performed to cover the outer limits of the scatter bands.

Disadvantages are:
- The application process is very complex and time-consuming.
- The manual application of algorithm thresholds is error-prone due to its complexity and requires intensive safeguarding measures.
- Measurement errors occur repeatedly in the data generated for the application, which are very difficult to identify due to the large amount of data.
- It is very difficult to determine the scattering range occurring in the tests by a few individual tests. This repeatedly leads to the fact that applications have to be adapted in the late project phase, which involves additional effort and costs.

Initial situation:
Current crash algorithms process a large number of signals, which are processed by airbag control unit-internal sensors and external satellite sensors in the crash algorithm.
For the correct design of the crash algorithms (application) many accident scenarios are run in the crash laboratory. Based on the sensor signals generated in the crash laboratory, the crash algorithm is "adjusted" in a complex application process so that it always makes the right decision in the scenarios that occur. The application process takes place mainly through the fine-granular adjustment of trigger thresholds.

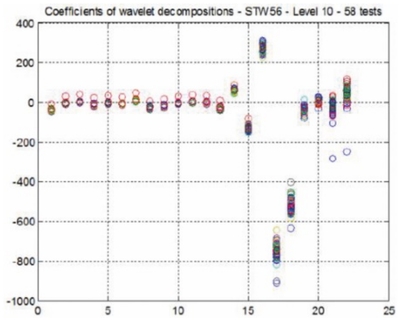


During the application process, about 15 different front crash scenarios, about ten different side crash scenarios and about five rear crash scenarios are run for data generation. In addition, numerous missuse scenarios are generated. Usually, each scenario is subject to a certain scattering caused by test tolerances, different vehicle equipment, etc.
In order to enable a correct classification, up to 40 to 50 individual algorithm thresholds are set by an application engineer in a modern vehicle/crash algorithm.
The number of crash tests run per scenario is very limited due to the high effort and costs involved.

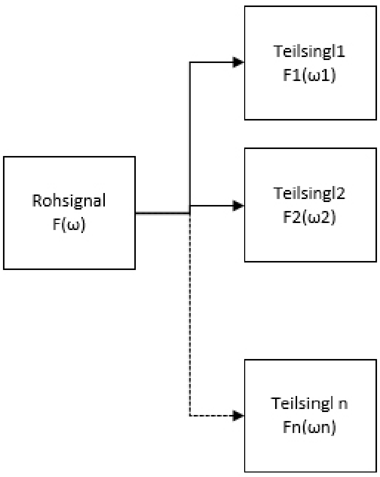
Solution:
1. Generation of synthetic crash signals
  a) Extraction of the signal scattering into a scattering model
  b) Generation of any number of synthetic crash data
2. Use of deep learning methods for
  a) Training of neural networks with the help of synthetic crash data
  b) Classification of crash scenarios using deep learning methods
    I. Plausibility check of measured data after the test has been carried out
    II. Use of Deep Learning for novel crash algorithms

Generation of synthetic experimental data

Any number of synthetic signals can be generated from a single crash signal.
Step 1: Decomposition of the crash signal into high and low frequency components. This can be done e.g. by wavelet transformation or band-pass filtering. The following example shows wavelet coefficients for several signals of a load case scenario:



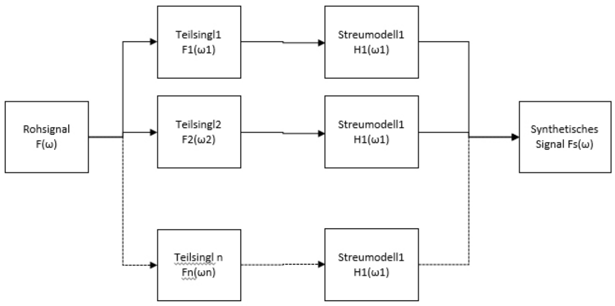
The low-order coefficients represent signals in the lower frequency range, the higher-order coefficients represent signals with higher frequencies.
The following diagram illustrates in principle the decomposition of a raw signal into different partial signals with different frequency bands:



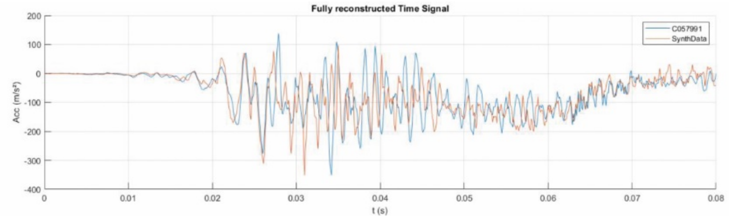
Step2: The different signal components are modified with suitable scattering models. Crash signals show different types of scattering depending on the frequency band:

- In the low-frequency range (up to 50Hz) the signal varies mainly due to macroscopic influences, e.g. other engine variants, equipment variants etc.
- In the high-frequency range (up to 400Hz), stochastic effects, e.g. component variations, experimental variations etc., increasingly occur which are not reproducible. Due to sampling effects, stochastic signal components also occur in the useful signal (after filtering).
- Depending on the frequency band of the crash signal, different scattering models are used
- The low-frequency signal component can be varied very well with FEM simulation. Different vehicle variations are simulated and the resulting scattering width is described by mathematical models, e.g. by wavelet coefficients or statistical distribution functions.
- In the high-frequency range, the variation of the signals is performed stochastically. The necessary parameters can be determined from previous projects.

The following diagram illustrates the procedure in principle. The scattering width of Model1 is determined, for example, with the help of FEM simulations, with which the partial signal1 is varied. The scattering width of Model2 is determined from previous projects, with which the partial signal2 is varied.



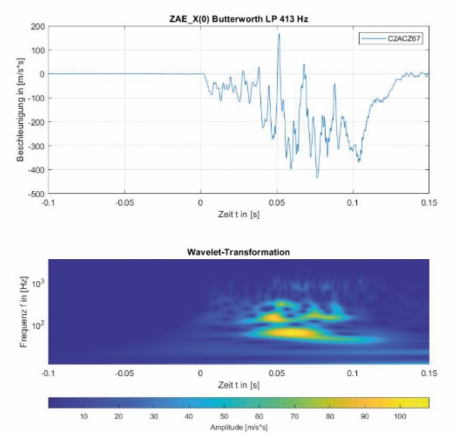
After variation of the partial signals, they are reconstructed to a synthetic signal.
Example of a signal synthetically generated with the method:



With this method any number of synthetic crash data can be generated, which are subject to the previously defined stochastic characteristics. The synthetically generated data can be used in the application process to further increase the robustness of the application and as training data for deep learning procedures.

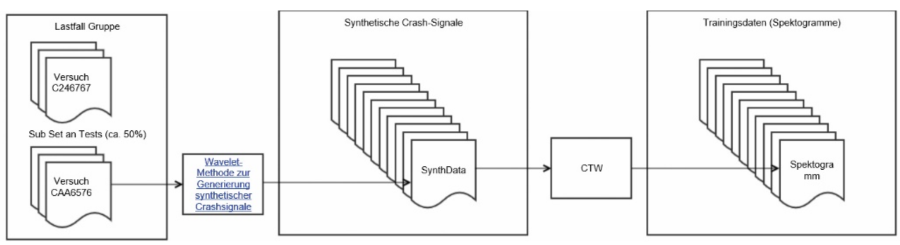
Plausibility check of experimental data, based on deep learning methods

In recent years, great progress has been made in the field of deep learning, especially in the classification of image data (e.g. differentiation of dogs and cats on an image). For the classification of image data open source networks are available, which can be trained for individual applications.
With conversion methods, such as Continuous Wavelet Transformation (CTW), or Gramian Angular Fields (GAF), it is possible to generate image data from time series signals. This method is applied to crash data.
The following example shows how a crash signal is converted into an image file using CTW.



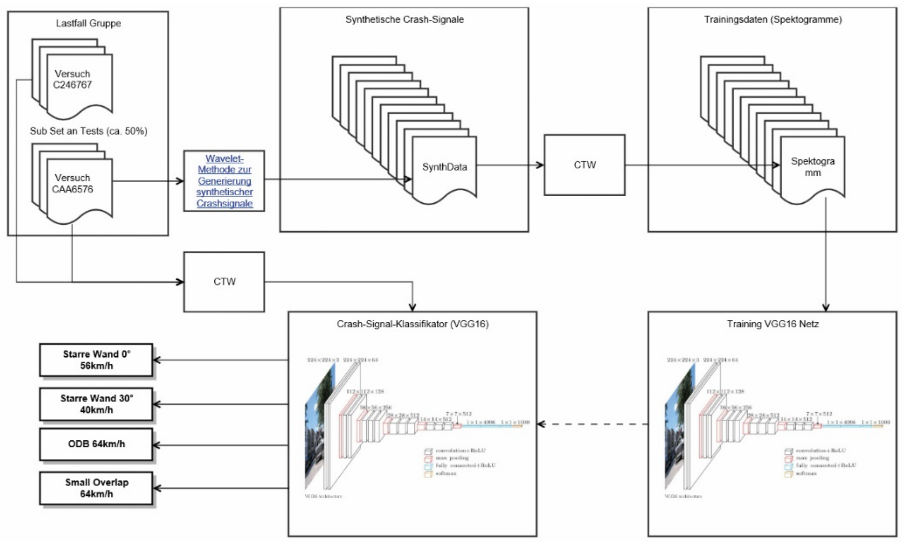
In combination with the method for generating synthetic data, training data for a Deep Convolutional Neural Network (CNN) can be generated. For the validation of the network it is necessary that only a subset of the real crash data is used for the stochastic variation of the training data. The remaining data, which were not used in the training process, are later used to validate the network.

The following figure illustrates the process for generating synthetic training data using the wavelet method and CTW:



The synthetically generated signals are based on the stochastic variation of a load case sub set. The spectrograms generated by CTW serve as input for the training process and are based exclusively on synthetically generated time series.

The next step is to optimize the CNN using training data. The open source network VGG16 is used as an example. The VGG16 network was developed at Oxford University and is specialized in image recognition. The VGG16 network is a pre-trained network, i.e. the deeper layers are already developed (the pre-training of these layers took several weeks of computing time on NVIDIA Titan Black GPU's) and only the last layers are trained application specific.



With the optimized network, the data excluded from the training process is used as validation.

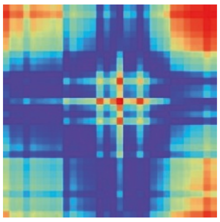
The described procedure can be used to plausibilise measurement results. For this purpose a large number of synthetic O.K. and N.O. measurement signals are generated which are then used to optimize the CNN. The optimized CNN can be used by the measurement technology directly after the crash test to effectively identify implausible measurement signals. The CNN used can be successively improved by combining it with other O.K. and N.O. signals generated during the development process. measurement signals occurring during the development process.

Novel crash algorithm based on deep learning methods

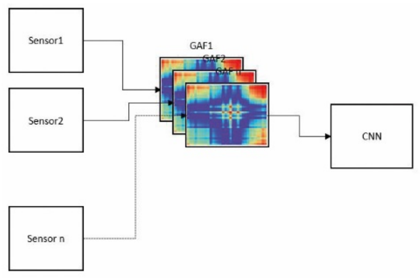
The method described above can be used in crash algorithms in the vehicle, or on a backend.

Application in the vehicle:
For the use in the vehicle it is necessary to generate the image data with the most effective method. For this purpose, e.g. GAF can be used, which is less computationally intensive than CTW.

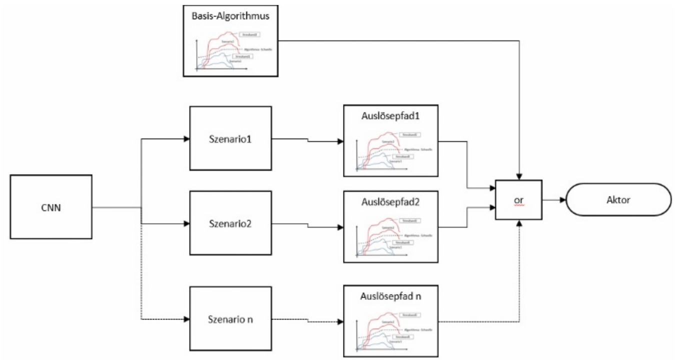
The following example shows how a crash signal is converted into an image file using GAF.



For a robust categorization of different load case scenarios it is necessary that all relevant crash sensors are considered in the CNN. Each load case scenario is therefore represented by a series of images (vector), which are derived directly from the crash sensor signals.
Conventional crash algorithms work with so-called signal features, which are extracted farom the raw sensor signals. It is necessary to focus on certain features. The commitment to certain signal features always leads to the loss of certain signal components and thus information.
In contrast to conventional triggering logic, it is not necessary to perform feature selection with the CNN method. The generated image data contains all information about amplitude, frequency and time course of the crash signals. The CNN can thus extract the information necessary for classification during the training process.
The following illustration shows the procedure in principle using the GAF method:



The network classifies the existing scenarios on the basis of the image vectors. The CNN classification is linked with conventional triggering logic in such a way that the CNN takes over a pre-classification of the incoming scenarios. The actual triggering decision is made using conventional triggering logic. This procedure shifts the complexity from conventional triggering logic to CNN. The conventional triggering logic can be implemented with little effort and serves as a second decision level.
In addition, a basic algorithm, which is built from conventional triggering logic, ensures the basic performance, for achieving legal requirements, and the field robustness. In order to achieve the peak performance required for consumer protection goals, trigger paths specialized for the respective scenario are selectively enabled by CNN.
The basic algorithm architecture is shown below:



Application on a backend:
For non-time-critical applications, the described procedure can be used on a backend. For this purpose, the crash data is sent to a backend via the radio interface located in the vehicle after an accident or light collision. With the procedures described above, a previously optimized CNN classifies the respective data. The following applications can be realized with this method
- Detection of parking damage/vandalism
- Correlation with injury data and output of an injury risk for rescue services (eCall) after an accident
- Estimation of damage severity after collision for use in insurance and after sales market

Advantages:
- Improvement of conventional algorithm/application
    - Increasing the robustness of a crash application through a larger database
- Fast plausibility check of measurement data
    - Large amounts of data can be quickly checked for plausibility -> data quality for the application of crash algorithms increases
- Novel crash algorithms based on deep learning methods
    - Effective application process through the use of deep learning networks