SIMULATION OF ACOUSTIC PRODUCT PROPERTIES IN VIRTUAL ENVIRONMENTS BASED ON ARTIFICIAL NEURAL NETWORKS (ANN)

Siegel, Antje (1); Weber, Christian (1); Albers, Albert (2); Landes, David (2); Behrendt, Matthias (2)
1: Technische Universität Ilmenau, Germany; 2: Karlsruhe Institute of Technology (KIT), Germany

Abstract

Acoustic product properties are playing an increasingly important role for product developers and designers. At the one hand the sound of a product has influence on the buying behavior of customers and on the other hand manufacturers have to meet many regulations and standards regarding the sound emissions of their products. Limits for these sound emissions are defined in order to protect humans and the animal world from noise pollution. The paper deals with a concept for acoustic simulation in virtual environments so that the acoustic properties of products can be studied. The concept is based on artificial neural networks (ANNs) which are applicable for many different tasks. With help of the virtual acoustic simulation the acoustic behavior of products or of individual product components can be validated during the whole product development process. In this way, noise pollution can be reduced and a higher compatibility with human health and environment can be reached.

Keywords: Virtual Engineering (VE), Simulation, Design informatics, Artificial neural networks, Acoustic product properties

Contact:
Antje Siegel
Technische Universität Ilmenau
Department of Mechanical Engineering
Germany
antje.siegel@tu-ilmenau.de

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Permanent noise emissions can have negative consequences on humans and the animal world. Due to this, in many application areas manufacturers have to meet standards which regulate noise emissions of their products. Among other requirements, the acoustical behaviour of a product has to be considered during product development. However, the state of the art is experience-based noise control (or "sound design") with experimental analysis and optimisation rather late in the development process. Innovative development tools are necessary in order to enable early-phase assessment of the acoustical behaviour of a product and more efficient design procedures. Virtual Reality (VR) can serve as such a tool. There are a number of solutions available for early product visualisation using VR; however, there is a lack of adequate tools for the study of acoustical properties of products – despite the fact that combining acoustics with visualisations is well established in entertainment applications of VR.

Opposed to entertainment applications, using the VR technology in product development must represent the behaviour (mechanical, acoustic, …) of the product correct enough for the designer to draw valid conclusions. This is difficult because all interactions between the VR-user and the object(s) represented in VR have to happen in real-time. Therefore, some simulation methods – especially numerical methods which are otherwise quite common in product development – cannot be applied directly.

This paper deals with new methods for virtual acoustical simulation. These methods are based on Artificial Neural Networks (ANNs). ANNs offer promising opportunities and can solve very complex tasks. In the following section a technical framework for an audio visual VR environment is described. Based on this technical framework two concepts for the virtual simulation of acoustical product properties have been developed. In both these concepts a so-called autoencoder is used. An autoencoder is a special kind of ANN and is characterised by a very specific structure. The developed approaches are applicable to many technical products. One specific example of application is the simulation of sound emissions of vehicles. The paper shows how the developed methods can be used to assess noises in an existing traffic simulation.

2 TECHNICAL FRAMEWORK

The concepts introduced in this paper are based on a flexible audio-visual stereoscopic projection system (FASP) as described in (Husung et al., 2014), shown again in Figure 1. For the visualisation six projectors are used. The images of the virtual scene are projected to three screens. The two outer screens are moveable so that the FASP can be set up in three variants (power-wall, theatre installation, "CAVE"). For user detection a tracking system is installed at the top of the screens.

The speciality of the installation is that, besides visualisation, audio content can be reproduced by a sound system that consists of loudspeaker panels with up to 208 speakers. These panels are arranged around a defined area in front of the screens. In this area the user can navigate and interact with the virtual scene.

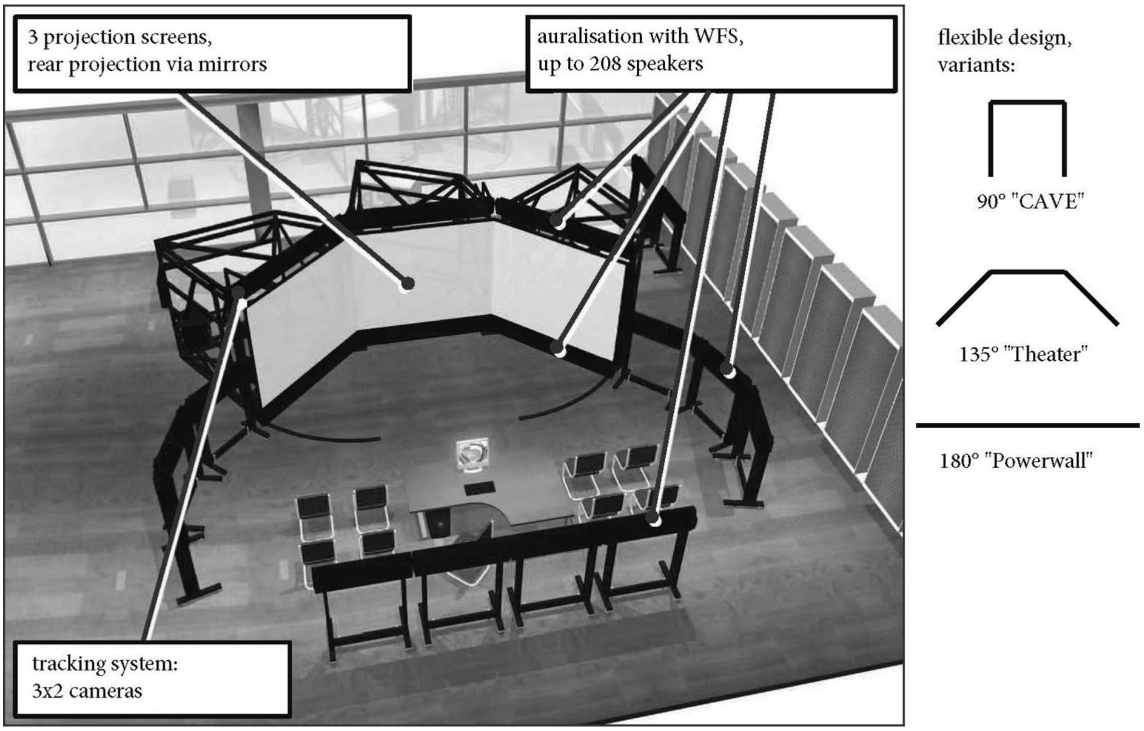

Figure 1. Technical framework

The speakers are controlled by a wave field synthesis system (WFS system). This WFS system is based on the Huygens' principle which states that any wave front can be substituted by the superposition of elementary spherical waves (Berkhout, 1988; Spors, 2007). These elementary waves are generated by the loudspeakers and in this way the entire wave fronts of sound sources can be reconstructed. In comparison to other sound systems the WFS technology provides a realistic spatial sound impression within the whole area between the loudspeaker panels – independent of the position of the listener. For this reason the calculation of head-related transfer functions (HRTFs) are no longer necessary. This is an advantage against systems which are based on binaural techniques (Lentz, 2008). A good spatial sound impression is not confined to a single sweet spot (Start, 1997; Vogel, 1993; Wittek, 2007). The sound of moving sound sources is influenced by the Doppler effect. This issue is addressed by algorithms which are also part of the WFS-system. The WFS-system works strictly object-oriented; this means that "sound objects", their positions and movements, etc. are transferred to the system; the resulting wave field is computed by the WFS-system in real-time. Thus, the actual virtual scene can be set up with standard VR software in form of a scene graph. Different kinds of scene objects can be created with this software. Every created object is stored with the respective object properties in the scene graph. In a similar way, the sound sources are also treated as objects and can be stored along with the reference properties. For every sound source a so-called sound object is created and one WAVE-file for each sound object is stored. Sound sources can be interactively activated and deactivated. In case of the activation of a sound source the respective WAVE-file is played back with the WFS system. The VR software provides the visual content as well as relevant acoustical data. This visual and acoustical data is processed separately by the visual reproduction system and the WFS-system. For this reason interfaces for the communication between the different technical components are essential. For the communication of the VR software and the WFS system a sound server was developed (Husung et al., 2014; Siegel et al., 2016). As shown in Figure 2, data is received and sent by the sound server via OSC messages (open sound control massages).

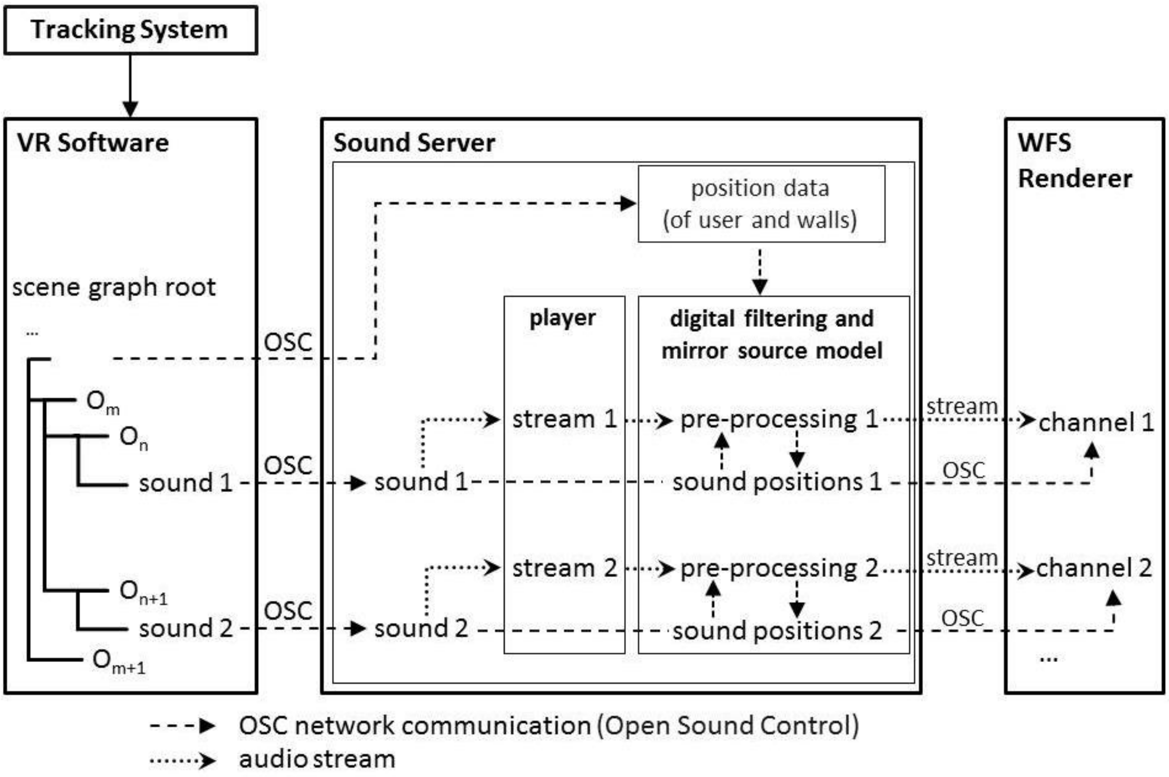

Figure 2. Structure of the sound server

Audio data is read from the WAVE-files, which are specific for each of the sound sources. Several kinds of audio pre-processing methods are available in the sound server. The audio data can, for example, be manipulated by the use of digital filters. These digital filters are also stored in the WAVE format. The filtering is done in the frequency domain. For this purpose the overlap add method is applied (Nussbaumer, 1990). Sound reflections which appear at objects are simulated with a simplified mirror source model (Siegel et al., 2016). This simplified model is applicable for sound reflections that occur especially at plane walls. These sound reflections are represented in the virtual scene by additional sound sources, the so-called mirror sound sources. For each wall one additional mirror sound source against each original sound source is created. The user position as well as the position of the walls and the position of the original sound sources are analysed and it is calculated which reflections arrive at the user's location. Corresponding to this, the right mirror sources are activated or deactivated. After that, the pre-processed audio data is sent block by block via audio stream to the WFS device. In addition to this, the current positions of the original as well as the mirror sound sources are provided to the WFS system by OSC messages.

3 ARTIFICIAL NEURAL NETWORK (ANN)

3.1 ANNs

First kinds of ANNs have already been developed in the 1940s (Rojas, 1996). Since ANNs can provide versatile solutions for many different problems, they are still very common and popular today. "Big data" applications in various fields have given new attention in recent years. ANNs are useful for many kinds of applications. For this reason a large number of approaches have been developed, regarding the structure and functionality of ANNs. The idea that all ANNs have in common is to imitate operations of neurons (nerve cells) in the human brain. However, the aim is not to fully rebuild biological brains. It is rather a matter of creating mathematical models which are oriented towards the neuronal processes in the nerve cells. The signal processing in the human brain is carried out by means of billions of neurons. These neurons are cross-linked and in this way an effective network is available for an efficient and fast signal transmission between the individual neurons. A neuron has several inputs (Figure 3a). At these inputs electrical impulses can be received from other neurons. If the impulses at the neuron’s inputs are high enough a new impulse at the output of the neuron is produced and directly utilised or can be further sent to other neurons as an input. An ANN works in a similar way.

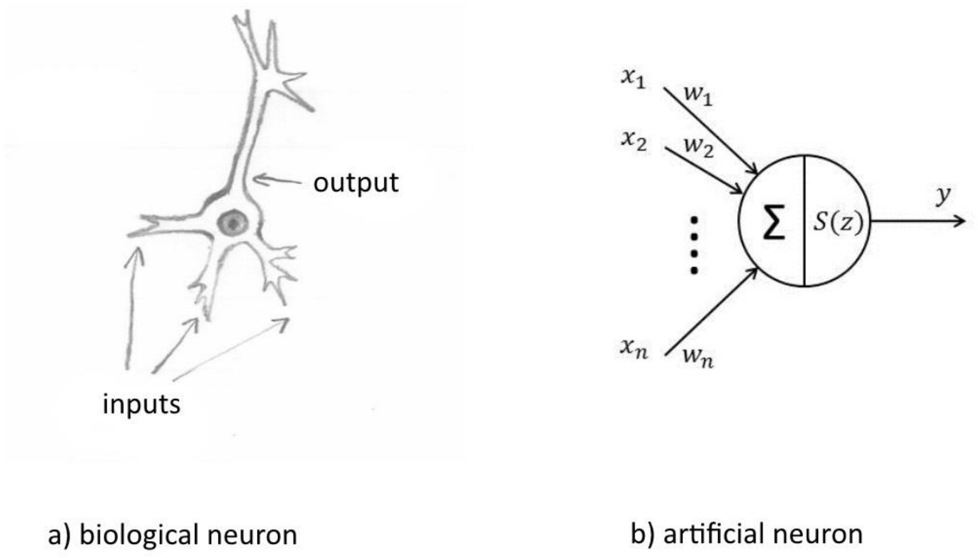

Figure 3. a) Biological neuron, b) Artificial neuron

The key elements of an ANN are also called neurons. The functioning of such an artificial neuron is shown in Figure 3b. Analogous to a biological neuron an artificial neuron also has several inputs, in Figure 3b indicated with x_1 to x_n. The input signals of natural neurons are electrical impulses. However, ANNs are used for the processing of digital data that is usually represented by discrete sequences of numbers. Therefore, the inputs of an artificial neuron are represented by numbers. These numbers at the inputs are weighted. That means that they are multiplied with respective weight values (represented in figure 3b as w1, w2, … wn). The Result of this multiplication is summed up, and if the value of the sum is high enough the artificial neuron will produce an output. This output is also represented by a number. Artificial neurons are usually arranged in so-called layers. An ANN consists of an input layer, one or more hidden layers and of an output layer. The neurons of the individual layers are linked with the neurons in the adjacent layer (Figure 4), more precisely the output of each neuron is linked to the inputs of neurons in the next layer and these inputs are weighted as just described.

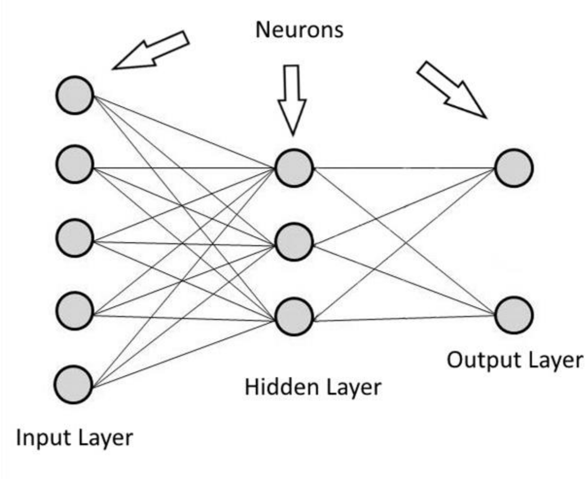

Figure 4. Artificial neural network (ANN)

The weights influence and control the output of an ANN. A desired behaviour of the ANN can only be reached by finding reasonable values for the weights. Suitable weights are determined by special algorithms which can be a very time consuming procedure. The whole process of determining suitable weights is called training. A very common application for ANNs is the classification of image data. The task here of the ANN is to recognise the content of pictures, e.g. the recognition of handwritten digits.

For the training different images of handwritten digits are provided to the ANN. The ANN first tries to learn which digits are shown on the pictures and adjust the weights inside the network accordingly. The results of the ANN are verified by calculating a kind of error rate that is used to adjust the weights through an iterative process. In that way the performance of the ANN is optimised until the error rate is decreased to a predefined/desired value. A well trained ANN can not only recognise the content of pictures that were used for the training, but also of pictures not shown in the training. The ANN is, so to speak, able to learn and generalise. For example, every person has an individual and unique handwriting. However, once a human has learned to write and read he/she is able to interpret the handwritings of other people. The same effect can be achieved for ANNs with an extensive training. For that purpose, pictures of handwritten digits of tens of thousands of people with different cultural background have to be considered in the training. That requires a certain effort but the result is an ANN that can be used easily for universal digit recognition. Handwritten material has only to be converted to a digital format, e.g. a jpeg-image. A digital image is nothing else than a sequence of discrete numbers. These numbers are put in the ANN. The ANN can produce one or more numbers at its output. These numbers are representative for a certain digit. In that way the result of the ANN can be used for further digital signal processing, if needed. The understanding of the capabilities of ANNs is very promising and provides the further motivation to integrate them in different technical frameworks.

3.2 Audio Signal Processing with ANNs

In case of using an ANN for the acoustic simulation in VR, different aspects have to be considered. As described in Section 2, for every sound object in the virtual scene an audio file is stored. The approach, introduced in this paper, is to create an ANN that is able to manipulate these stored audio data. This would require using the stored audio data as input for the ANN. The time signal of digital audio data is represented by a discrete sequence of numbers. These numbers are called samples. Each audio sample can be put to respectively one neuron in the input layer. At the output of the ANN the manipulated audio signal should be provided. This output should be in a format that can be further processed, in order to accomplish the audio reproduction within the VR environment. Another issue to be considered is the user interaction in the virtual scene. In accordance with the user's interactions the sound of the scene has to be adjusted. For example, one could imagine that the user can choose between different materials within the virtual product model. Therefore, parameters have to be found that represent the current acoustic behaviour of the chosen product materials. For this purpose the sound absorption coefficients of the different materials could be used as parameters. Also factors which are correlated directly with noise origination should be considered, such as the speed of an engine in an electrical device. For a successful virtual acoustic simulation an ANN has to be designed that is able to interpret these parameters.

3.3 Autoencoders

To meet the requirements described in Section 3.2 the authors created two concepts which are based on the use of autoencoders. An autoencoder is a special type of ANN. The typical structure of an autoencoder is shown in Figure 5. It consists of two components which are called the encoder and the decoder (Vincent et al., 2010).

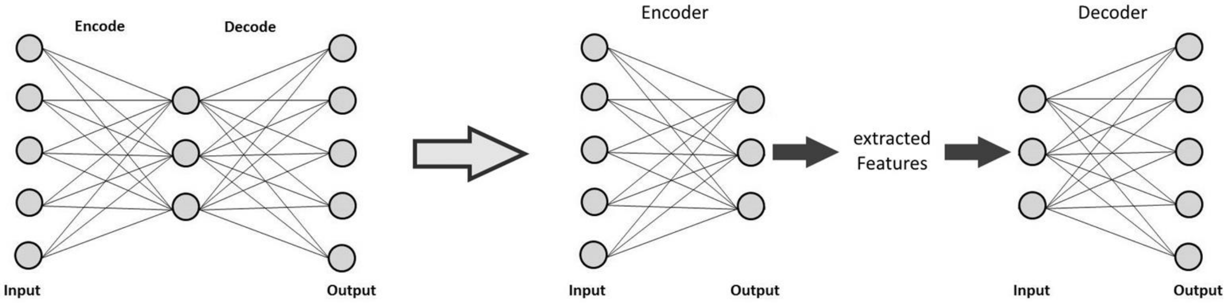

Figure 5. Autoencoder

In the encoder the input data is reduced and characteristic features are extracted. In the case of using the autoencoder for the processing of audio data an audio file is used as input. The number of layers can vary in accordance with the complexity of the desired application of the network. In that way, the input data is stepwise compressed. That means that the amount of numbers at the input is reduced. The numbers thus generated at the output of the encoder are called features. In this regard, often the expression "feature extraction" is used. It is understood that the extracted features are in some way characteristic for the input data. The extracted features are used as input for the decoder. The decoder's design is usually the inverse of the encoder (Figure 5). Due to this, the original input data can be reconstructed by the decoder. During training the output of the decoder is compared with the encoder's input. An error is calculated which is representative for the differences between the two signals. The network is optimised until the error is smaller than a user defined limit to reach an adequate functionality.
One application of autoencoders is so-called denoising. Denoising is used, for example to reduce artefacts or "noise" in pictures (Vincent et al., 2010). A "noisy" picture is put into the autoencoder and the encoder extracts features which are characterising the actual image content. These features are used in the decoder to generate an image which is less noisy. In best case the actual picture content is fully reconstructed without any noise. For a satisfying result enough information of the original image content has to be available in the image. Otherwise the extracted features are too imprecise. In the same way an autoencoder can be used for noise reduction in audio signals. An interesting aspect in respect to acoustical VR simulation is that the autoencoder's output has the same format as the input. Several kinds of ANNs exist, which can process an audio signal at the input. However, in most cases at the output of ANNs individual numerical values are provided that do not represent the samples of an audio signal. In contrast, an autoencoder can process audio data at the input and generates an audio signal at the output. That makes autoencoders applicable for acoustic simulation. How to precisely use an autoencoder for VR applications is describes in the following sections.

4 CONCEPTS

Applications for the use in VR have to work in real-time. Only then can the user properly interact with the virtual scene. Audio reproduction for VR application is generally realised by audio streams. Accordingly, audio data is processed in blocks. The two concepts, introduced in this section, therefore are designed in such way that audio data can be put into an autoencoder block by block. In both approaches it is assumed that the acoustical behaviour even of complex products can be characterised by different individual sound sources. For each of these sound sources an audio file has to be stored. These sound files are loaded and looped.

In the first approach (Figure 6) the audio data is pre-processed before the autoencoder is applied. This pre-processing is done depending on different parameters which characterise the product and the scene (e.g. parameters like sound absorption coefficients of different product materials). The sound of a product changes according to these just mentioned parameters. These changes in sound are approximately simulated by the pre-processing. This pre-processing includes actions like scaling the sound intensity, adjusting the playback speed or applying digital filtering. Such methods can be implemented relatively easily, but the results are rather imprecise. At this point the autoencoders can help considerably by improving the results of the pre-processing. The underlying idea is that the autoencoder operates very similar to the denoiser that was mentioned in section 3.3. In that way, the sound of a product can be recorded and stored as audio file. Material changes at the product will lead to another acoustic behaviour of the product. These new acoustic properties of the product could be simulated with the autoencoder. The stored audio file is pre-processed in accordance with the sound absorption coefficients of the changed materials. The further processing in the autoencoder generates an audio signal that represents the changed sound of the product. In that way the sound emissions of different product variants can be simulated.

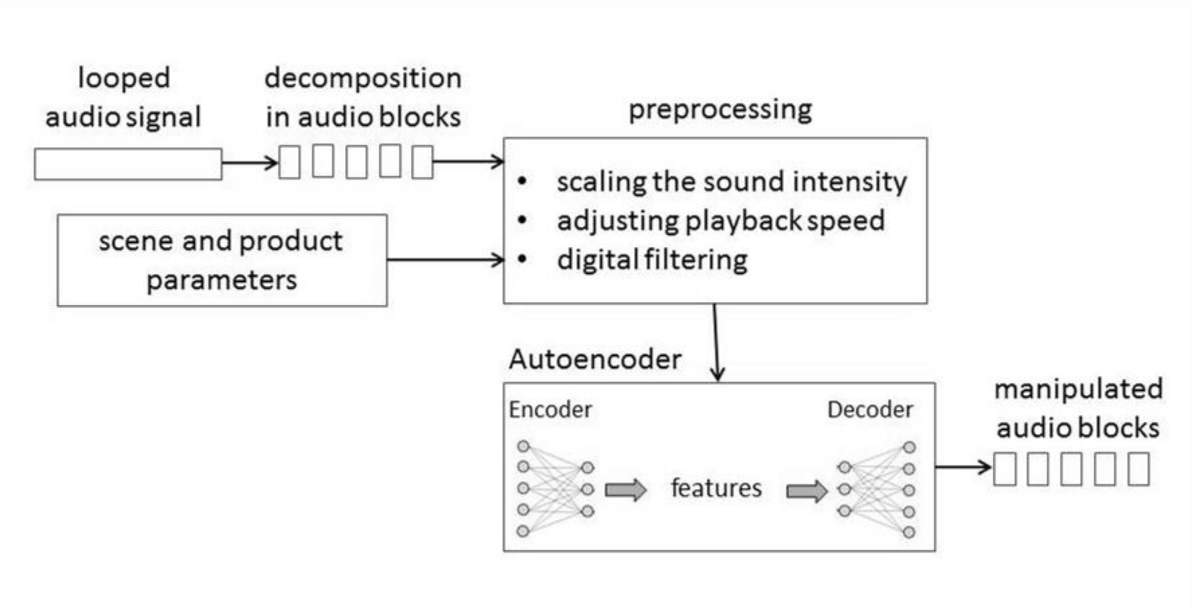

Figure 6. Concept with pre-processing

The second approach is to process the decomposed audio blocks directly in the autoencoder (Figure 7). In contrast to the first concept, operations which depend on product parameters and the scene are not done with the actual audio data but with the extracted features at the encoder's output. In that respect, further investigations have to be made. The purpose of these investigations is to find rules for the manipulation of the features. The features are characteristic for the audio signal at the input. It is expected that acoustic properties of different materials are in some way represented in the extracted features. The aim is to determine dependencies between the extracted features and different design parameters for the product. In that way, similar to the first approach, one recording for a certain product type is sufficient to simulate the acoustic behaviour of most diverse variants of the product. The recorded audio signal can be manipulated in the encoder by adjusting the extracted features according to changes of the design parameters.

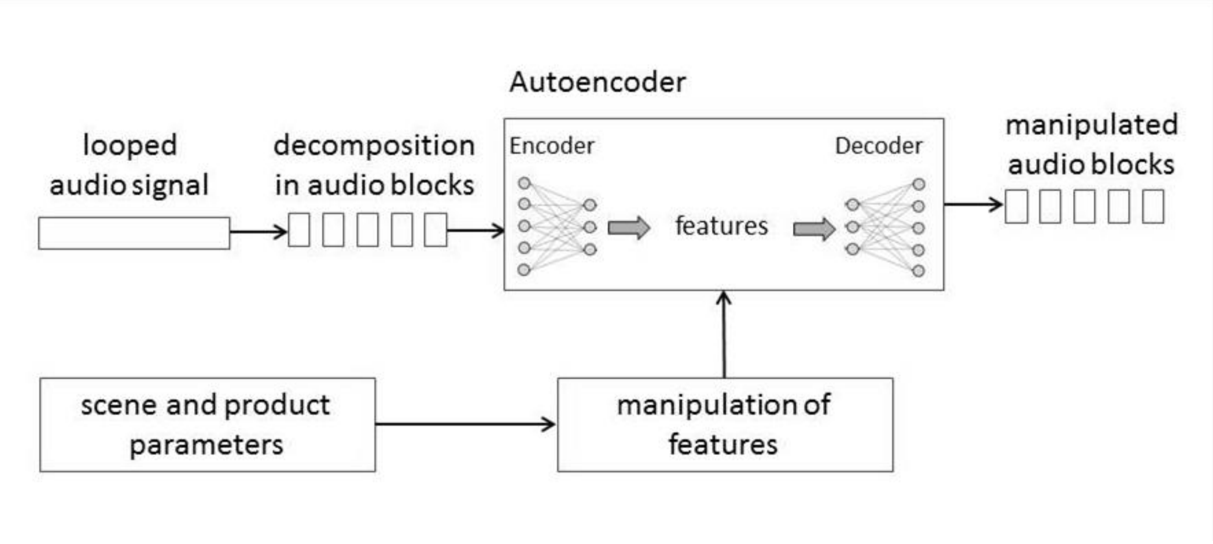

Figure 7. Concept with feature manipulation

5 APPLICATION

One industrial sector in which sound emissions play a prominent role is the automotive industry. In order to avoid humans and animals to suffer from traffic noise, car manufacturers have to meet quite strict standards and regulations. At the same time the sound of a particular make of car is contributing to the maker's image, so internal and external sound is carefully designed. Existing standards, however, only take weighted sound pressure levels into account. These sound pressure levels do not consider psycho-acoustic factors like the impulsiveness of the sound. How sound is perceived by humans is a very complex subject, and whether noise is considered disturbing (or pleasant) depends on many different parameters. At least, hearing is an individual process that is difficult to characterise by mathematical quantities like sound pressure levels etc. Acoustical simulation within a virtual environment is therefore a very helpful tool. Tests with volunteers can lead to new findings with regard to sound perception and noise control. The technical framework described in section 2 (flexible audio-visual stereoscopic projection system, FASP) is used for a traffic simulation. In that way, specifically the noise pollution caused by cars can be investigated.

For this traffic simulation an acoustical data base was created. This data base includes the audio recordings of vehicles with different drive topologies: Passenger cars with conventional internal combustion engines, with electric and hybrid drive concepts were measured on a roller test bench. A virtual scene which presents a busy road was created and can be presented to users via the FASP. For the cars in the scene a virtual acoustical model was developed which was already described in (Husung et al., 2014) and in (Siegel et al., 2016). The model is of modular design. In that way, the primary sound sources of the cars are represented separately.

For the validation of the model tests with volunteers were made. As a first step, a simplified model was subject of some informal listening tests. These tests verified that fundamental requirements for a successful realisation of the virtual acoustic model are fulfilled (Siegel et al., 2014). As a second step, the final acoustic model was evaluated by further studies with volunteers. In a second listening test, described in (Siegel et al., 2016), the subjects had to assess audio samples of simulated vehicle noises. In that way, the plausibility and realism of the virtual acoustic model was examined.

In future work the concepts, described in Section 4, will be implemented, in order to be integrated into the sound server that was outlined in Section 2. State parameters like varying vehicles' speeds or load conditions can be sent at regular time intervals to the sound server. These parameters can be used for the pre-processing in Figure 6. In order to achieve a good performance of the autoencoder extensive training is necessary. This training can be done with the above mentioned acoustical data base. Successful training will lead to an autoencoder that can also process data that was not part of the training (e.g. non-steady-state manoeuvres derived from steady-state measurements). In that way, the measuring effort for new vehicle models can be reduced.

Moreover, it will be investigated whether the second approach, illustrated in Figure 7, is applicable for the vehicle sounds. The sound of the vehicles is dependent on different parameters. For example, for vehicles with conventional combustion engine the vehicle speed, the current gear or the load condition can be used as parameters. For hybrid and electric vehicles also the degree of hybridisation or the powertrain's type could be considered. It will be investigated if dependencies between these parameters and the extracted features at the encoder's output exist. As described at the end of Section 4, rules have to be found for the manipulation of the extracted features. Similar to the first concept, only one predefined driving manoeuvre would have to be recorded for a new vehicle model. Other driving manoeuvres could be simulated by the manipulation of the extracted features.

6 PRE-TESTS / PROOF OF CONCEPT

For a successful realisation of the concepts outlined in Section 4 the following essential requirements have to be fulfilled:

- The autoencoder has to be able to process audio data block by block.
- The autoencoder has to be able to decode the audio data sufficiently precisely.

In (Maas et al., 2012) it is shown that autoencoders can be used for noise cancelation in audio signals and that a sufficient decoding of audio data is possible. Although the autoencoder in (Maas et al., 2012) was trained with audio blocks, the length of the used audio blocks, however, is not mentioned. As a consequence, no statements can be made whether block lengths, which are needed for audio processing in real-time, can be put into practice. To study this question, own tests were made by the authors, as described in (Siegel et al., 2017). The testing was based on the application example that was pointed out in Section 5. An autoencoder was trained with audio recordings of vehicle noises. The audio recordings were divided into blocks of a length of 784 audio samples before the training. The task of the autoencoder was to reconstruct the audio data applied to the autoencoders input as exactly as possible. The tests showed that low and mid frequencies of the input signals could be reconstructed very well. Only for frequencies which were higher than 15 kHz small deviations between the input and the reconstructed signal could be detected.

7 CONCLUSION

In this paper two approaches for the simulation of acoustical product properties within virtual environments are introduced, both based on Artificial Neural Networks (ANNs). These approaches make use of a special type of ANN or more precisely, they make use of autoencoders. The high potential and operating principle of autoencoders is illustrated and the specific use of autoencoders for audio signal processing in VR is also pointed out. The concepts were developed with view to an audio-visual VR-framework described in section 2. Finally, an example of application based on a traffic simulation is given. In future work the concepts will be implemented and integrated in the above described framework and the results and findings for the application example are to be transferred to more generic models, which are applicable for product development in general.

REFERENCES

Berkhout, A. J. (1988), "A Holographic Approach to Acoustic Control", *The Journal of the Audio Engineering Society, 36(12), pp.977-995*. 36(12),pp. 977-995.

Husung, S., Siegel, A. and Weber, C. (2014), "Acoustical Investigations in Virtual Environments for a Car Passing Application", *ASME 2014 International Design Engineering Technical Conferences, CIE*. Buffalo, USA. https://doi.org/10.1115/detc2014-34936

Lentz, T. (2008), "*Binaural technology for Virtual Reality*", PhD thesis, Rheinisch-Westfälische Technische Hochschule Aachen.

Maas, A., Le, Q., O'Neil, T., Vinyals, O., Nguyen, P., Ng, A. (2012), "Recurrent Neural Networks for Noise Reduction in Robust ASR", *INTERSPEECH, pp. 22-25*

Nussbaumer, H. (1990), "*Fast Fourier Transform and Convolution Algorithms*", Springer-Verlag: Berlin, Heidelberg.

Rojas, R. (1996), "*Neural Networks: A Systematic Introduction*", Springer Verlag, Berlin Heidelberg.

Siegel, A., Weber, C., Albers, A., Landes, D. and Behrendt, M. (2017), "Akustische Simulation von Fahrzeuggeräuschen innerhalb virtueller Umgebungen basierend auf künstlichen neuronalen Netzen (KNN)", *Wissenschafts- und Industrieforum 2017 Intelligente Technische Systeme,* Paderborn, Germany (to be published)

Siegel, A., Weber, C., Mahboob, A., Albers, A., Landes, D. and Behrendt, M. (2016), "Virtual Acoustic Model for the Simulation of Passing Vehicle Noise", *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC2016*, Charlotte, USA. https://doi.org/10.1115/detc2016-59872

Siegel, A., Husung, S., Weber, C., Albers, A., Landes, D., Behrendt, M. (2014), "Simulation of Acoustic Properties of Technical Systems Using a Network-Based Sound-Server", *58th Ilmenau Scientific Colloquium 2014,*Technische Universität Ilmenau, Germany.

Spors, S. (2007), "*Active Listening Room Compensation for Spatial Sound Peproduction Systems*", PhD thesis, University of Erlangen-Nürnberg.

Start, E. (1997), "*Direct Sound Enhancement by Wave Field Synthesis*", PhD thesis, Delft University of Technology.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. and Manzagol, P. (2010), "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion", *Journal of Machine Learning Research*, 11 (2010) 3371-3408.

Vogel, P. (1993), "*Application of wave field synthesis in room acoustics*", PhD thesis, Delft University of Technology.

Wittek, H. (2007), "*Perceptual Differences between Wavefield Synthesis and Stereophony*", PhD thesis, University of Surrey.

ACKNOWLEDGMENTS

The authors would like to thank the members of the Zeidler-Forschungs-Stiftung for their support.