Sam ESPALLAC 26/12/2022

Report

Sound design of an electric car

Abstract

Sound design of electric cars is nowadays very important to design in order to prevent accidents with pedestrian and to give to the driver feedback of the behaviour of the car. In this project, we designed the sound of an electric Škoda car from a provided simulation video and data. We used the phase modification of a sine wave according to RPM of the engine and acceleration data. We also used some filters to adapt the sound to a real car sound. Finally, we obtained a good result that matches well with the video but there are still some ways of improvement like adapting the sound to the human earing system or smoothing a part of it to make it sounds more natural.

Introduction

The sound design of electric cars is nowadays an increasingly common issue in our daily lives. *"How do you warn pedestrians of the approach of a vehicle that has an engine that makes almost no noise? How do you produce an audible warning that is both effective and benign? How do you create a signature sound design for electric vehicles that is exclusive to the brand? Since the beginning of electric cars, the French brand Renault has been a pioneer in the development of a specific so called VSP (Vehicles Sound for Pedestrian). This VSP became a reality in 2012 with the production of ZOE electric car." [1]* Renault sound engineer team developed the sound of the car in the way that it evokes the brand and warns the pedestrian that a car is nearby without scaring them. Our goal in this project, is to design the sound of a driving car showed in the video [2] in the same way. To create the sound, a large panel of data was available [3]: rotations per minutes of the engine, acceleration of the vehicle in 3D, his velocity and so on. A code has been created on MATLAB, that adapts the sound regarding these data.

Method

The starting point was to get the data, plot them in the time domain and see what we could do with. The method that has be chosen was to modify the phase of a sine wave depending on the rotation per minute of the engine. In fact, the sound of the car and the rotation of the engine are two parameters highly correlated. For each data point, we changed the frequency of our sine wave. It has been necessary to smooth RPM curve in order not to change the frequency of our sine wave too abruptly. Some parameters have been used to make the sound closer to a real engine. In another hand we used the acceleration data to give to the sound a rough timber when the driver accelerates. To do that we set some threshold levels regarding values of acceleration. Finally, the engine sounded a bit too electronical so we used a low and high pass filter.

Results

On figure 1 is shown the rotation curve of the engine and its smoothed one. On figure 2 is presented the acceleration curve and its smoothed one. Then, on figure 3 is shown the output signal we obtain before and after filtration. The audio file is quite fine and fulfil the conditions we set in the introduction. It corresponds well to the visual perception we have in the video and it is able to help the driver to feel like if he was in a regular combustion engine car. The fact that the acceleration parameter has been used makes the sound rougher at times. Therefore, when acceleration increases sharply, the car sounds like a race car, a sound characteristic that sometimes can be appreciated by the driver.

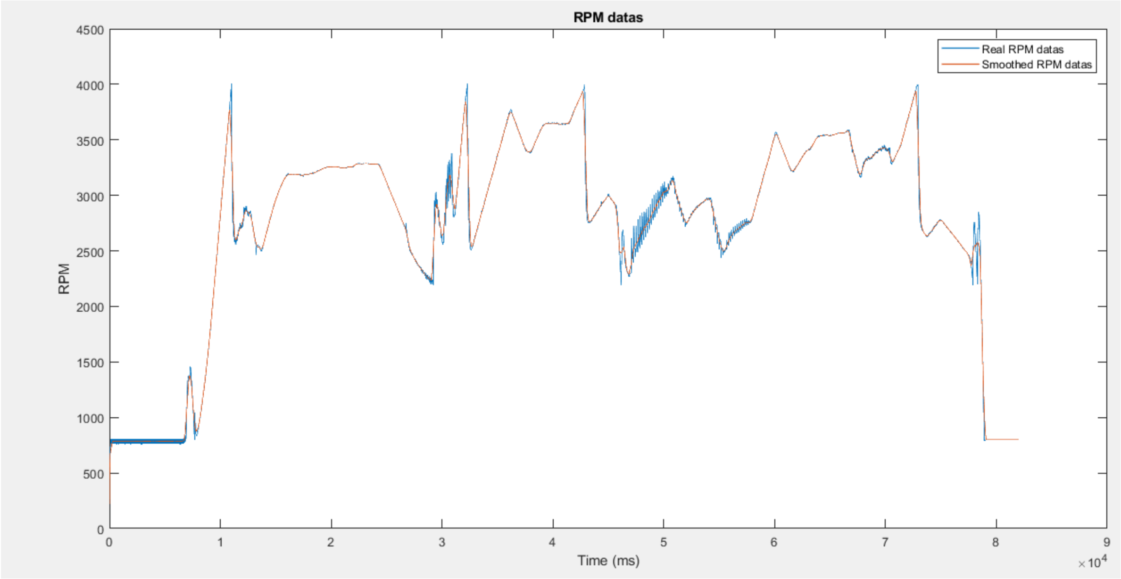

Figure 1: RPM and smoothed RPM data

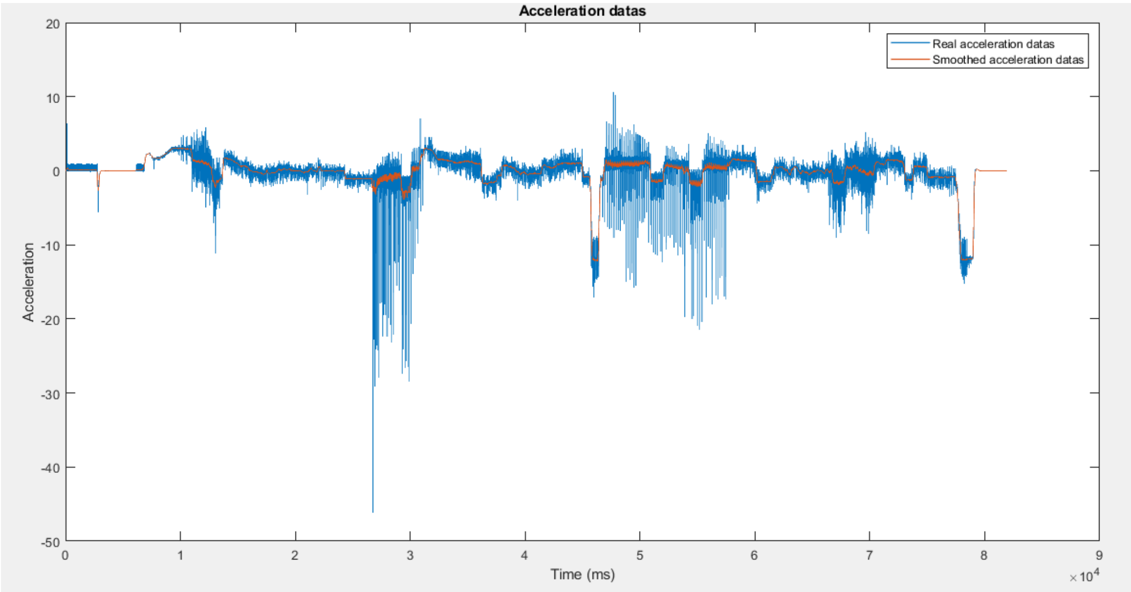

Figure 2: Acceleration and smoothed acceleration data

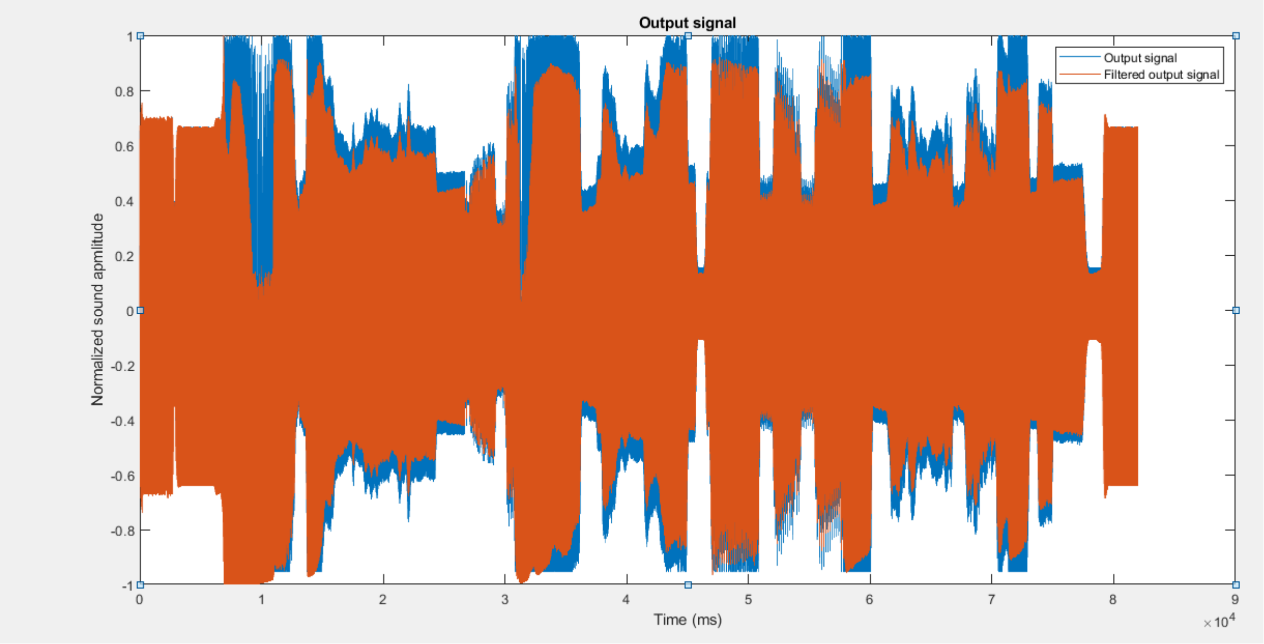

Figure 3: Output signals

Conclusion

The problematic of an electric car sound design has been encountered in this project. We have implemented a code to design the sound of a Škoda electric car according to a provided video and to provided data. We used different techniques using acceleration and rotation per minute of the engine to create a correlated sound. Finally, we obtained a sound that matches well with the provided simulation video. There are still many ways of improvement. We would have liked to adapt the sound with the frequency response of the human ear. It would have been good to reduce the sound in high frequencies for example, according to equal loudness curves, because the earing system is more sensitive in this band. Moreover, there is a weird behaviour of the sound during the last big acceleration of the engine that could be smoothed. The last improvement point is on the nature of the sound that could have been changed. We used a simple sine wave but we could have used the typical sound of an instrument to make it less electronic. However, we designed the sound on an electric car so this point can be discussed.

Musical instrument synthesis

Abstract

In this task, we synthetised the track 'Barcarolle' from 'The Tales of Hoffmann' by Jacques Offenbach using MIDI files. After using the function analyza.m to analyze the spectrum of an organ sample. We used the codes provided to recreate the track in our way using another instrument. We therefore used formant and additive synthesis. Finally, we obtained a track that sounds similar but with another sound colour.

Introduction

Modification of a track can be a good approach to remaster some music or to give them another colour in order to allow them to get trendy again. For our task, we looked at 'Barcarolle' track from 'The Tales of Hoffmann' by Jacques Offenbach. Our goal was to change this track using some techniques to give it a different sound colour and to learn how musical instrument synthesis works.

Methods

We used two different techniques in our code. The first is additive synthesis. It is an original spectrum modelling technique. Based on Fourier's theorem, which states that any periodic function can be modelled as the sum of individual sinusoids. Each particular sinusoid will have its particular characteristics regarding the amplitude and harmonics frequencies [4]. We chose this technique to create the organic sound. Then, we used formant synthesis. It is a famous method based on the source-filter-model. In order to produce a high-quality sound, normally, it is required to use from three to five formants. For additive synthesis, we first took a sample from the web and analyze it with analyza.m function. The, we got the amplitude values of each harmonic that we have implemented in our code. For formant synthesis, we first set different parameters like fundamental frequency, duration and so on. Then, we set the resonator coefficients with the help of a violin model characteristic provided. As soon as the signal has been filtered, an ADSR envelope is applied to smooth the resulted signal. This envelope is controlled by 4 parameters (attack, decay, sustain and release). Finally, the interpolation function is applied to select the specific position of the envelope and a normalization is done afterwards to simplify and adjust the resulting signal.

Results

The sample in time domain and the spectrum of the organ we obtained is available on figure 4 and 5, respectively. After running the main.m file, we obtained a new synthetised 'Barcarolle' track that sounds different as the provided one. The sound obtained is more clear, richer and of course, with a beautiful organic vibe.

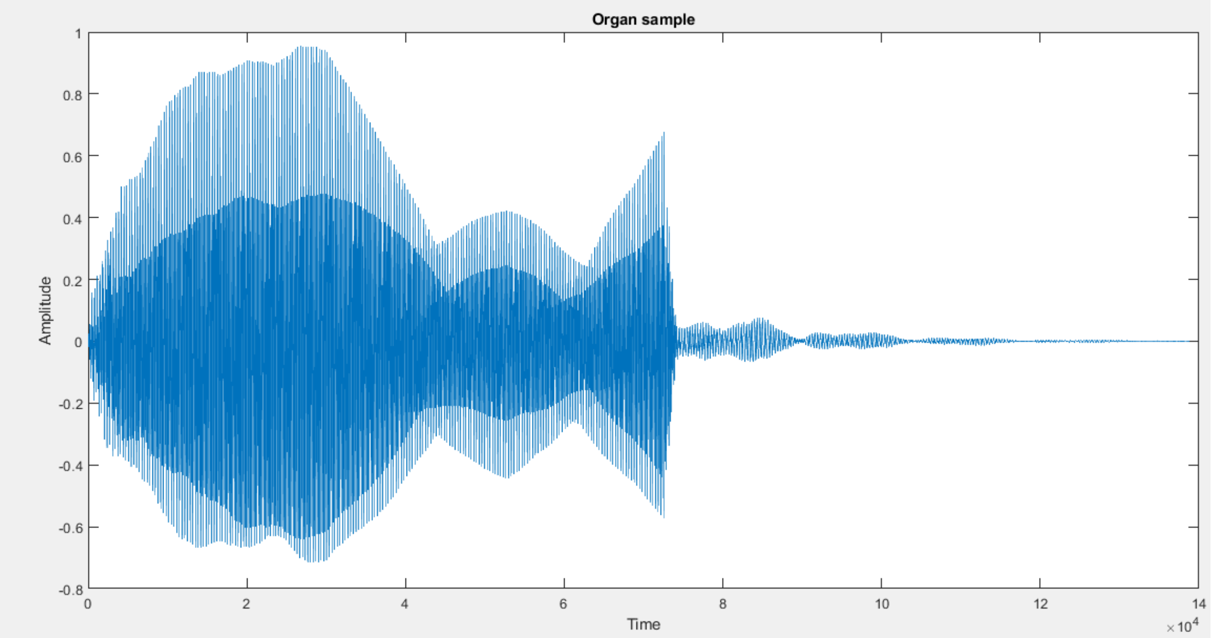

Figure 4: Organ sample in time domain

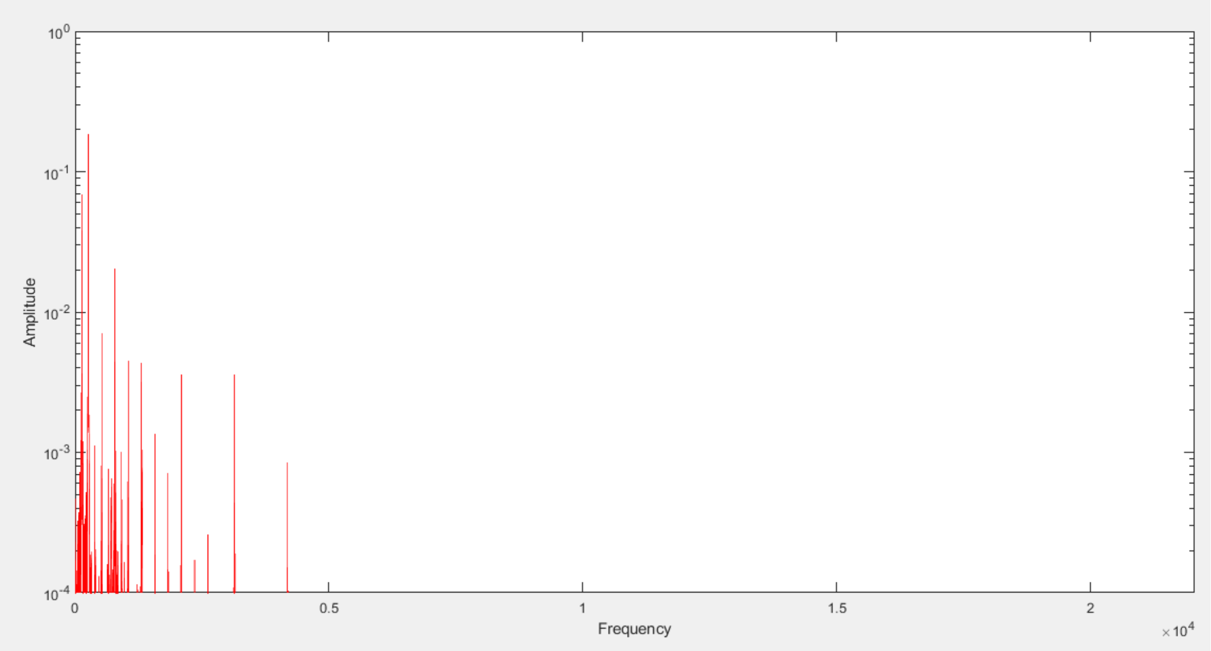

Figure 5: Spectrum of organ sample

Conclusion

During this task, we synthetised a new version of Barcarolle track with another sound colour using organ and violin samples. We have studied all the files available and understood that we can modify with an infinite number of possibilities any track. Improvements could be done in order to make the sound again richer. We also could have changed the sound by using other samples.

References

[1] Site Renault, https://www.renaultgroup.com/en/news-on-air/top-stories-2/renault-in-tune-with-sound-episode-2/

[2] Car video, https://sami.fel.cvut.cz/synthchallenge2022/car_video.mp4

[3] Data, https://sami.fel.cvut.cz/synthchallenge2022/control_signals.txt (data file), https://sami.fel.cvut.cz/synthchallenge2022/description.pdf (description of data),

[4] Matthias Robine, FAST ADDITIVE SOUND SYNTHESIS USING POLYNOMIALS. https://hal.archives-ouvertes.fr/hal-00307931/document