Robotic Multi-Sensor Fusion

A thesis submitted in partial fulfillment of the
requirements for the degree of Bachelor of Science in Physics
at the College of William and Mary

by

Scott T. Watson

Accepted for: B.S. in Physics

_____
Adviser: Mark Hinders

_____
Gina Hoatson

Williamsburg, Virginia
April, 2008

Abstract

The primary purpose of this study is to research the effectiveness of using multiple sensors in parallel with each other, in order to create a robot that can more effectively work in a human environment and assist humans in performing various tasks. These could vary from simply retrieving an object from a location to defusing a bomb. By the use of multiple sensors, as opposed to a single sensor, the robot has a greater sense of where it is, and what tasks must be done in order for the instructions to be carried out to completion.

Robotic Multi-sensor Fusion

Table of Contents

1. Introduction

In recent times robots have become an ever-increasing part of our lives. Performing tasks for us both in the open and behind-the-scenes, robots do many jobs for humans that are extremely dangerous, monotonous and time-consuming. Due to the need for robots to increasingly be used in environments originally designed for humans, a robot that can easily adapt to work in such areas would be more useful. Making a robot interact with a human environment, along with the humans in it, is an extremely difficult task to accomplish. A robot would need to understand and be able to maneuver in the area, as well as being able to accept commands from humans, and be able to adapt to changing circumstances. One of the steps that can be taken to accomplish this task is to allow a robot to collect as much information as possible about the environment it is in. This can be done by using sensors placed on the robot, in a technique called 'multi-sensor fusion'. This technique involves combining the data from these multiple inputs in order to get a larger data set of what is currently happening, much like how a human uses their senses to collect information about their situation. In order to select a variety of sensors that are complementary, testing must be done to determine each sensor type's viability for use on a robot. Currently, research has already been performed on an infrared sensor for use on a robot in the Applied Science department [2]. This project tests a second type of sensor for the robot, a combination of a parametric array and parabolic microphone to collect acoustic signatures of objects in the robot's path.

2. Theory

2.1. *Parametric array*

In non-linear acoustic theory, it is well known that when two superimposed plane waves of differing frequencies ω_1 and ω_2 are traveling in the same direction, they create two new waves, a 'sum' wave of frequency $(\omega_1 + \omega_2)$, and a 'difference' wave of frequency $(\omega_1 - \omega_2)$. These two new waves have the ability to exist separately from the two originating waves when entering an absorptive medium, allowing us to selectively attenuate the signals. If we properly choose the medium, we can attenuate both originating signals and the sum wave that was created by them, leaving the difference wave with only slight attenuation. Westervelt [1] has done a theoretical approach of this concept applied to a sound wave that utilizes Lighthill's equation for arbitrary fluid motion, which is a rearrangement of the Navier-Stokes equation for fluid motion into a wave equation:

$$(\partial^2 \rho / \partial t^2) - C_0{}^2 \nabla^2 \rho = (\partial^2 / \partial x_i x_j) T_{ij} \tag{1}$$

Where ρ is the density of fluid, C_0 is the speed of sound within the fluid, and the stress tensor is:

$$T_{ij} = \rho u_i u_j + p_{ij} - C_0{}^2 \rho \delta_{ij} + D_{ij} \tag{2}$$

Where D_{ij} is the stress tensor of the medium. By assuming no attenuation of the difference wave (due to its lower frequency compared to the other waves), that the pressure attenuation coefficients are the same for the originating waves, and that nonlinear attenuation is negligible, we can arrive at an equation for the pressure of the resultant difference wave (p_s) from the originating waves (p_i):

$$\square^2 p_s = -\rho_0(\partial q / \partial t) \tag{3}$$

Where $\square$ is the D'Alembert operator, and

$$q = \rho_0^{-2} C_0^{-4} [1 + \tfrac{1}{2}\rho_0 C_0^{-2} (d^2 p / d\rho^2)_{\rho=\rho_0}](\partial / \partial t)\rho_i^2 \tag{4}$$

This can give us the general equation for the pressure of the difference wave, which given by

$$p_s(\vec{r}) = -\frac{i\omega_s \rho_0}{4\pi} \int \frac{q e^{ik_s|r-r'|}}{|\vec{r} - \vec{r}'|} dV' \tag{5}$$

By assuming the originating waves have the form of

$$p_i = P_0 e^{-\alpha x} [\cos(\omega_1 t - k_1 x) + \cos(\omega_2 t - k_2 x)] \tag{6}$$

we can simplify the equation to one along the axis of the sound signals, which allows us to arrive at the final equation for the pressure of the difference wave:

$$p_s = (\omega_s^2 \rho_0 A / 8\pi R_0) e^{ik_s R_0 - \omega_s t} / [i\alpha + k_s \sin^2(\theta/2)] \tag{7}$$

Where the angle from the axis is

$$A = -i\omega_s \rho_0^{-2} C_0^{-4} [1 + \tfrac{1}{2}\rho_0 C_0^{-2} (d^2 p / d\rho^2)_{\rho=\rho_0}] P_0^2 \tag{8}$$

With this equation, we can determine the pressure of the difference wave at a distance x from the originating source.

2.2. *Other Research*

Robotics is a mature field of research, but the field of multi-sensor robotics is relatively new, only having emerged after computers became powerful enough to handle the large amounts of data from multiple inputs. Even so, there are numerous papers about the subject already out in the world. Felhman [2] has done research into using passive thermal-imaging for objects in an unstructured environment for object

classification. Other papers include using sonar as a method of distinguishing objects in outdoors environments [3][4][5].

For this study, however, we are trying to focus on the use of a parametric acoustic array. To that end, there have been a few notable papers researching this topic. The theoretical basis for such a speaker was done by Westervelt, (and is outlined in section 2.1.) While, Bennett [6] applied Westervelt's ideas to a speaker using air as its medium, which is notable if only for the fact that we too are using our speaker in the atmosphere. This idea was then applied to several areas, including the consumer sector, such as the Sennheiser AudioBeam [7] used in this project, and by the Army [8].

3. Experimental Technique

In order to collect the data for the project, we used a parametric array, a parabolic microphone, an amplifier, a laptop for MATLAB data collection, a second laptop for weather monitoring, and the stands and items required to properly set them up (see figure 1). The parametric array is used as a speaker to create an audible sound wave, which we can use to 'bounce' the sound wave off of whatever object that we have it aimed at. We used a Sennheiser AudioBeam as our parametric array. The parabolic microphone is used to collect the reflected sound waves from the environment. It connects to the laptop used for MATLAB through a standard headphone audio plug. The laptop is used not only for data collection, but also for generating the desired audio signal to be produced by the speaker. It then uses the connected microphone to collect the sound waves, and then save both a MATLAB (.mat) file, and a standard audio (.wav) file. The secondary laptop used in the setup was used to monitor the current weather by connecting to the William and Mary Keck Weather station at Lake Matoaka. This weather station was chosen due

to its close proximity to our recording location, and its frequent updates. The second laptop was used for this task only because a wireless internet connection was required to access the weather station's web page, as the laptop used for handling MATLAB did not have the required hardware to access it.

Figure 1: Image of testing phase setup.

In order to make our setup more compact, and remove the need for our setup to constantly be attached to an electrical outlet, we attached all of our equipment to a computer UPS, which itself was connected to a large battery (see figure 2).

Figure 2: Image of current setup.

We first created the signal in MATLAB using several programs developed in the lab, originally for use in detecting hidden weapons using a parametric array for security purposes [9]. The first program, ToneBurst.m, provided a way to create a sine wave signal of a desired time length, frequency, bit-depth, and sampling. The second, RecordTone.m, used an automated method to first generate the desired audio signal, and then allow the user to manually control the recording start and end times. It would then save a .mat file containing the raw recorded data, along with a wav file of the resultant sound signal.

When collecting data, we would note the type of car that passed by, e.g. a sedan, minivan, SUV, etc., and in which lane it was traveling. We would then use the secondary

laptop, which was monitoring the weather, to record the current weather conditions, so that we could account for frequency shifts from wind, moisture in the air, or other weather phenomena.

We collected data from cars on Richmond Road next to the Undergraduate Admissions building. First we would place the apparatus a measured distance from the edge of the street, and a measured distance to the right or left of a designated mark. We would then measure out two marks from the apparatus along the road, to mark a place to start and stop recording as a car would pass over them. A diagram of our layout is shown in figure 3. We would then wait for an individual car to pass by, to minimize sounds from other cars and people passing by. Once we had isolated a car, we would begin recording the moment its front bumper reached one of the marks, and stopped recording once the car's back bumper had passed over the opposing mark. MATLAB would then generate and save a .mat and .wav file of the recorded data. The type of car that passed by, whether it was in the near or far lane with respect to the position of the apparatus, and the current weather conditions were then recorded in the lab notebook, and on the laptop. This process was then repeated for each individual car that passed by.

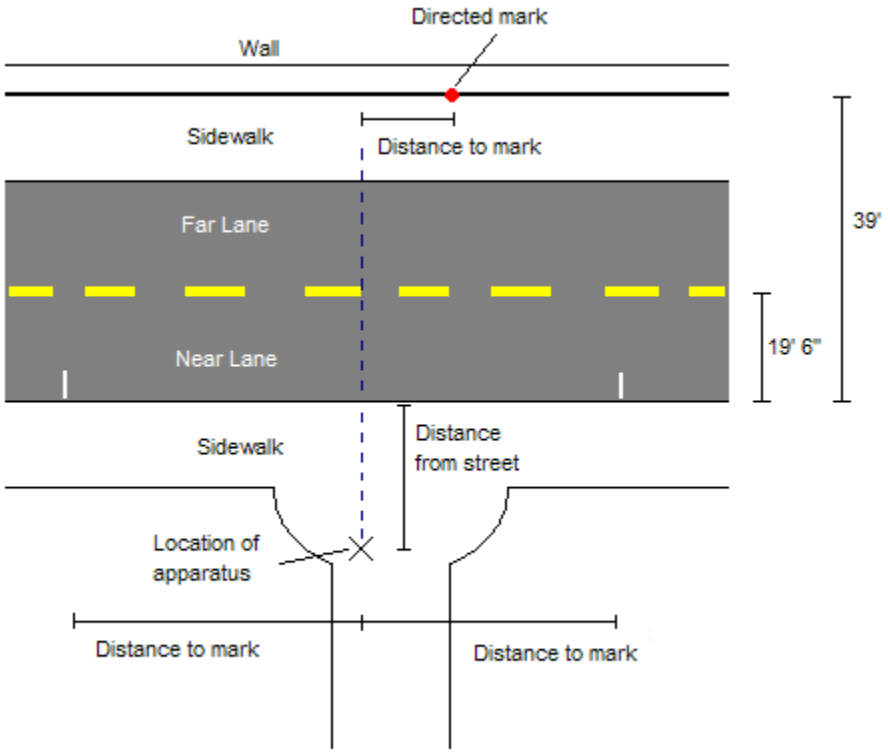

Figure 3: Diagram of setup.

4. Results

For the duration of the project, we collected 84 samples of cars passing by our setup. After initial inspection of our data we found that our setup was generating a great deal of ground-loop noise, which disappeared once the setup was plugged into a wall outlet or otherwise grounded to an object. We did find that the noise being generated was a consistent frequency, which could be filtered out, using a low-pass filter, as shown in figure 4.

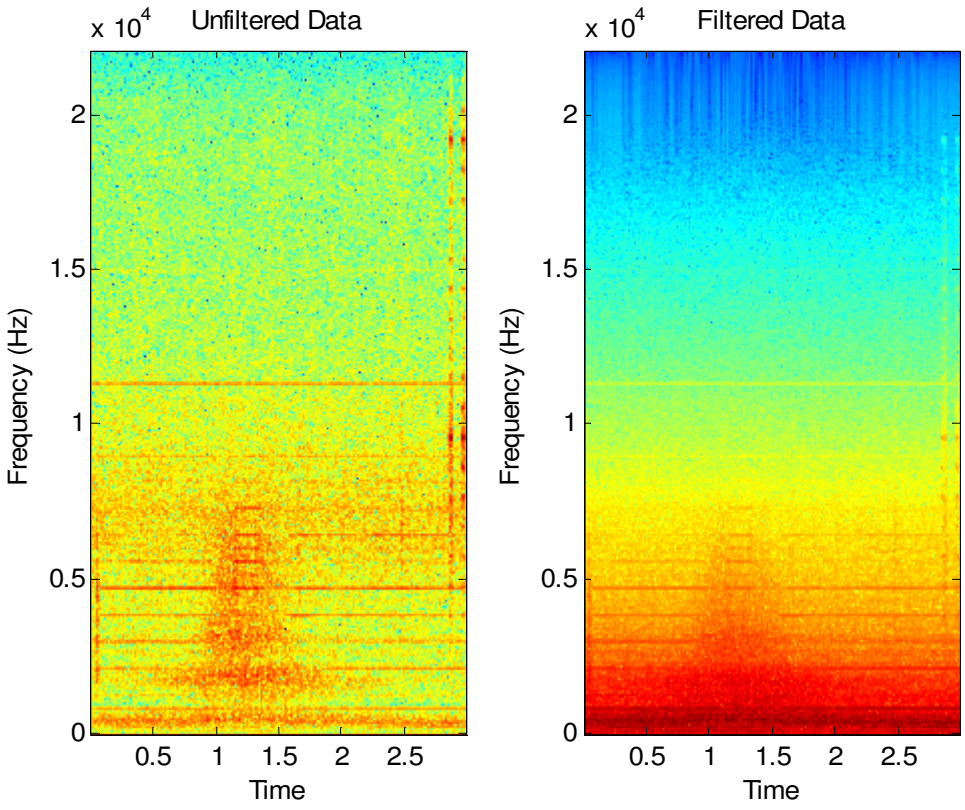

Figure 4: Image of unfiltered and filtered noise.

We also found that there was noise generated by the laptop itself, which was a periodic 'clicking' noise, generated by the normal functioning of the laptop. By listening to and plotting each of the sound files, we were able to discern the Doppler-shifted reflection of original signal that we had sent out.

Two examples of data sets are presented here, one perpendicular to the road, and one angled with respect to the road.

4.1. *Example Data Set 1*

This data set was collected on March 21, 2009, with the setup a distance 9' 9" away from the road, and 0' to the side of our designated mark. In the data set, a sedan passed by in the far lane of the road, between the 0.8 second mark, and the 1.5 second mark. For this, the car traveled 30' to reach the setup, and 30' past the setup, at which

point data collection stopped. In the spectrogram, we can see the point at which the car passes by, a repeating clicking noise generated by our laptop near the end of our signal, and the ground-loop noise generated by the setup around the 12,000 Hz mark. Near the bottom of the spectrum lies the signal, which we are trying to extract.

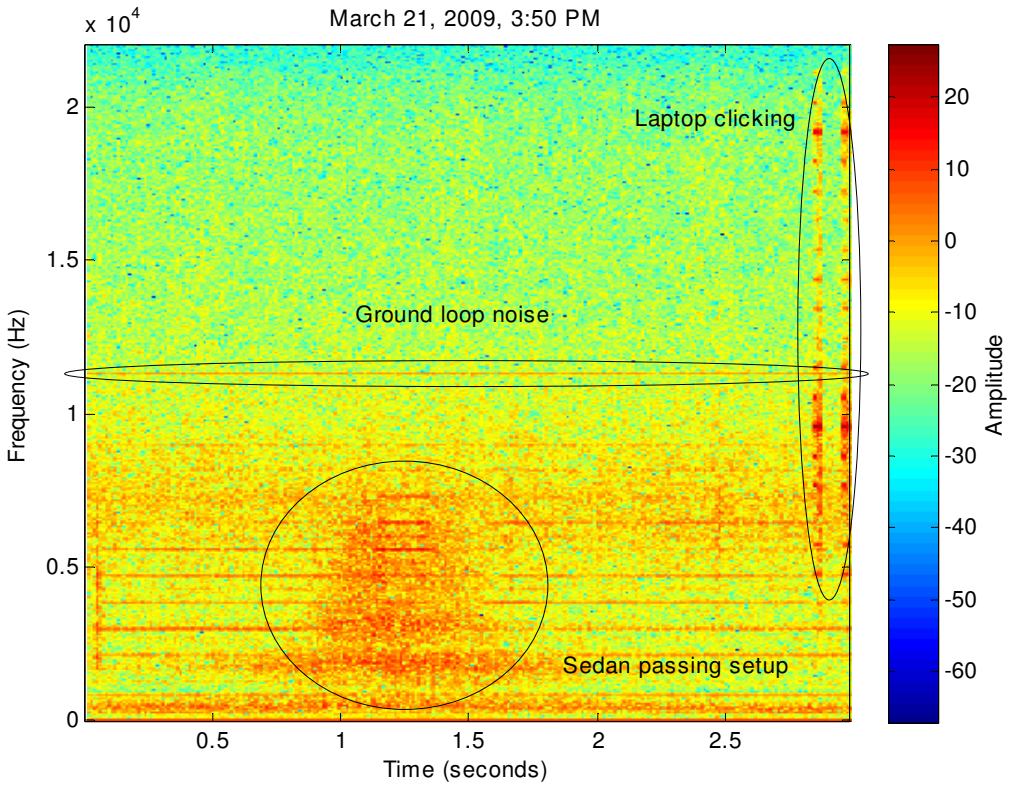

Figure 5: Spectrogram of example data set.

As the time period of interest is when the car passes by, we chopped out the part of the signal with the car, and using MATLAB's filtering routines, we applied a bandpass filter to it. We applied an elliptic bandpass filter, centered around our original signal of 440 Hz, with a passband of 30 Hz on either side of it, and the stop band at 50 Hz on either side, as shown in figure 6.

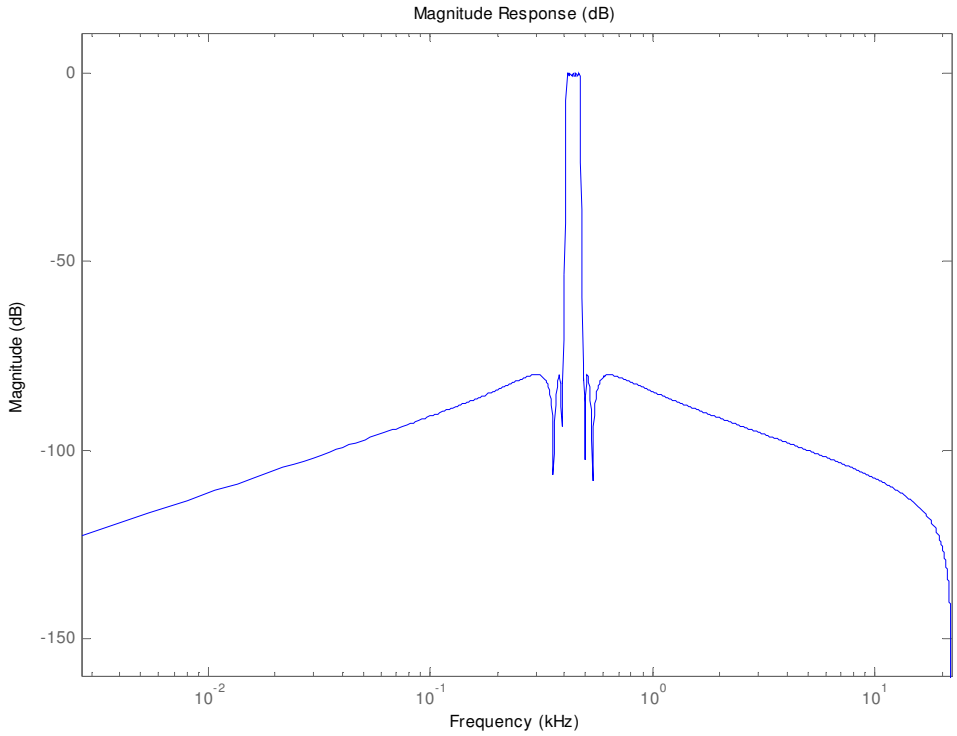

Figure 6: Magnitude response of the elliptic filter from 0 to 22.1 kHz.

This was performed to isolate the reflections near our original signal from the rest of the data. The range of the passband was selected using the speed of the cars, which could be calculated from the known distance traveled and the length of the audio signal, and calculating the maximum Doppler shifting that could take place, which fell within this range. We then plotted the FFT of our filtered signal to get the transforms in figure 7.

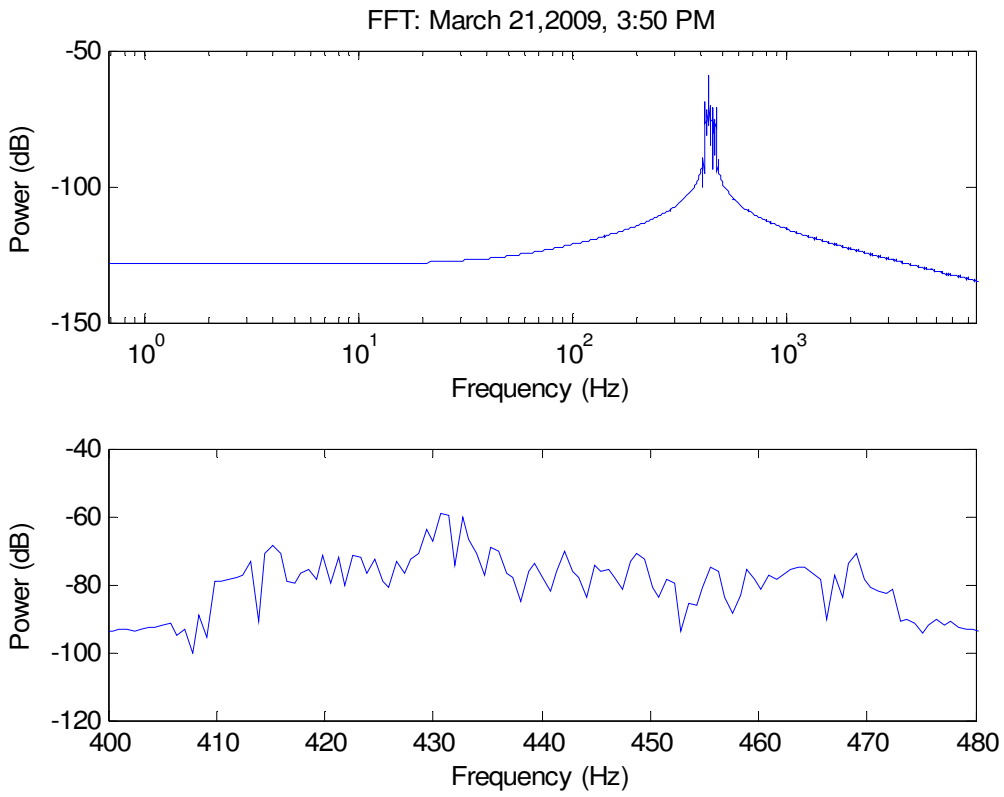

Figure 7: FFT spectrum of our chopped signal. Upper: FFT over entire signal length. Lower: Magnified view of upper FFT over the range 400-480 Hz.

In the lower FFT, we can clearly see there are numerous frequency peaks around our original signal. Since the car passing by would only have a few reflections, the amount of peaks in signal implies that we possibly had noise down in this region that went undetected, possibly from the car, or even the apparatus itself. But even among the noise we can pick out frequency ranges that are of some interest. Out of the peaks, we can pinpoint peaks at around 430 Hz, due to their higher power than the surrounding peaks, and around 460Hz, due to their continuous spectrum, as being some of the most prominent. Trials of filtering the data over varying passband lengths had shown a consistent set of peaks that would change with the placement of the passband ends,

meaning that the peaks were caused by the filtering function itself. Because of this artifact from the filtering process, we ignore the peaks at the 410 Hz and the 470 Hz mark.

4.2. *Example Data Set 2*

This data set was collected on March 22, 2009, with the setup 0' away from the road, and 29' 6" to the left of the target we were aiming towards. In this data set, a sedan passed by in the near lane of the road, between the 1.0 and 1.8 second marks. For this set, the car traveled 30' to reach the apparatus, and 50' past the setup, at which point we stopped collecting data. In the spectrogram for this data set, we can see when the car passes by the setup, the clicking generated by our laptop near the end of our signal, as well as the ground-loop noise generated by the setup, above the 10,000 Hz mark. Near the bottom, we can make out our initial signal, which we are trying to extract.

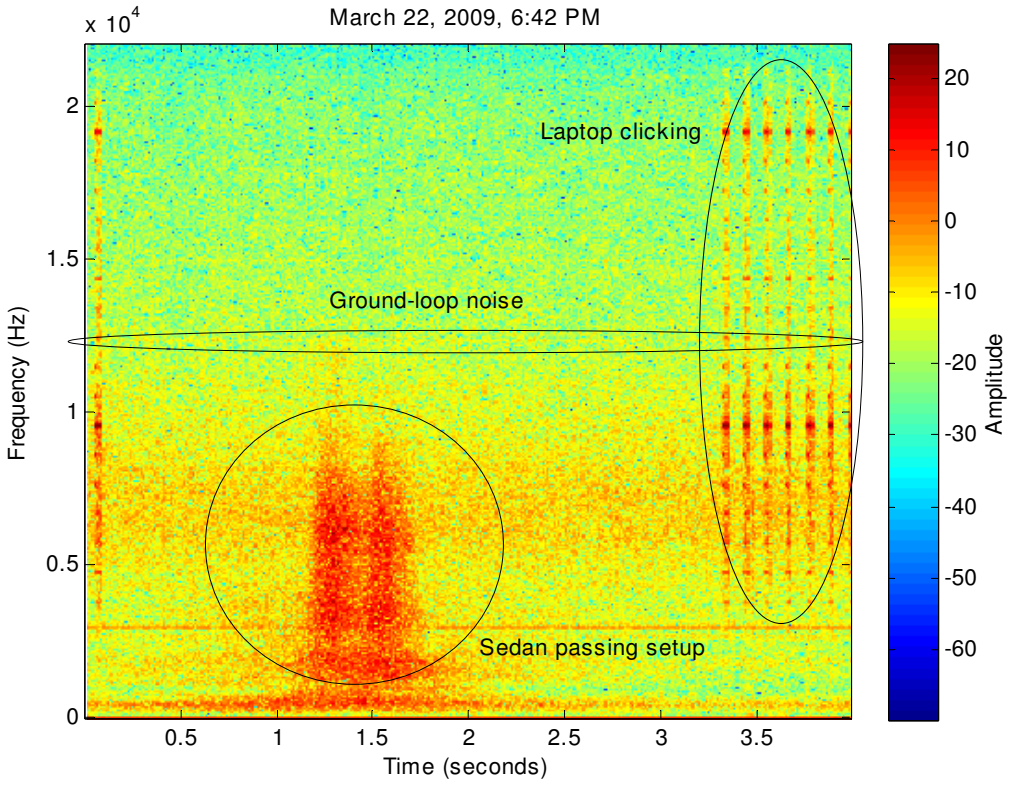

Figure 8: Spectrogram of example data set.

Using the same methodology we had used for our previous example, we filtered our chopped signal, with the same elliptic bandpass filter as shown in figure 6. Plotting the fast Fourier transform of our signal, we got the following plot:

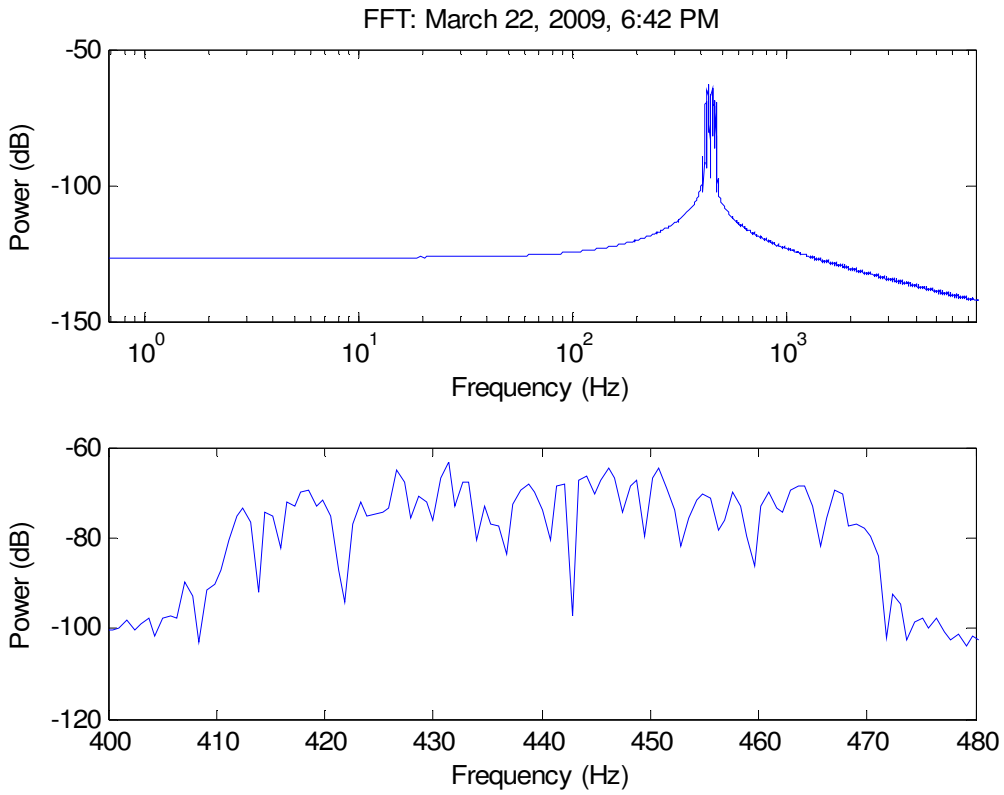

Figure 9: FFT spectrum of our chopped signal. Upper: FFT over entire signal length. Lower: Magnified view of upper FFT over the range 400-480 Hz.

Once again, the first image in our spectrum displays the FFT over the entire length of our chopped signal, and the second image displays the FFT magnified over the frequency region of interest. For this signal, we can see there are several peaks of frequencies around our original signal. Due to the random nature of the peaks in the second plot, we can see that noise has filled in this signal as well, either from the car, or the apparatus itself. However, we can still pick out peaks around 420 Hz and 430 Hz,

due to their higher power compared to the surrounding peaks, and 460 Hz, due to its continuous spectrum of peaks, as being more prominent. An interesting valley in our spectrum is near our original signal of 440 Hz. This implies that little of our original signal was reflected back to us, most likely due to the angle (with respect to the approaching car) we were taking data at.

4.3. *Histogram analysis*

To see if there were any prominent frequencies in our signals as a whole, we performed a histogram analysis on the sedans and SUVs that we had collected data on, which is shown in appendix B. This was done in order to find possible patterns in our data, while simultaneously reducing the effect of noise on individual peaks by binning our frequencies. We did this by categorizing our data samples based on car type, and filtering them using the method in the examples. We then set our histogram bin size to be 1 Hz in width, in order to remove small discrepancies due to the atmosphere. Station wagons, minivans, and trucks were left out of the analysis, due to their extremely small sample sizes.

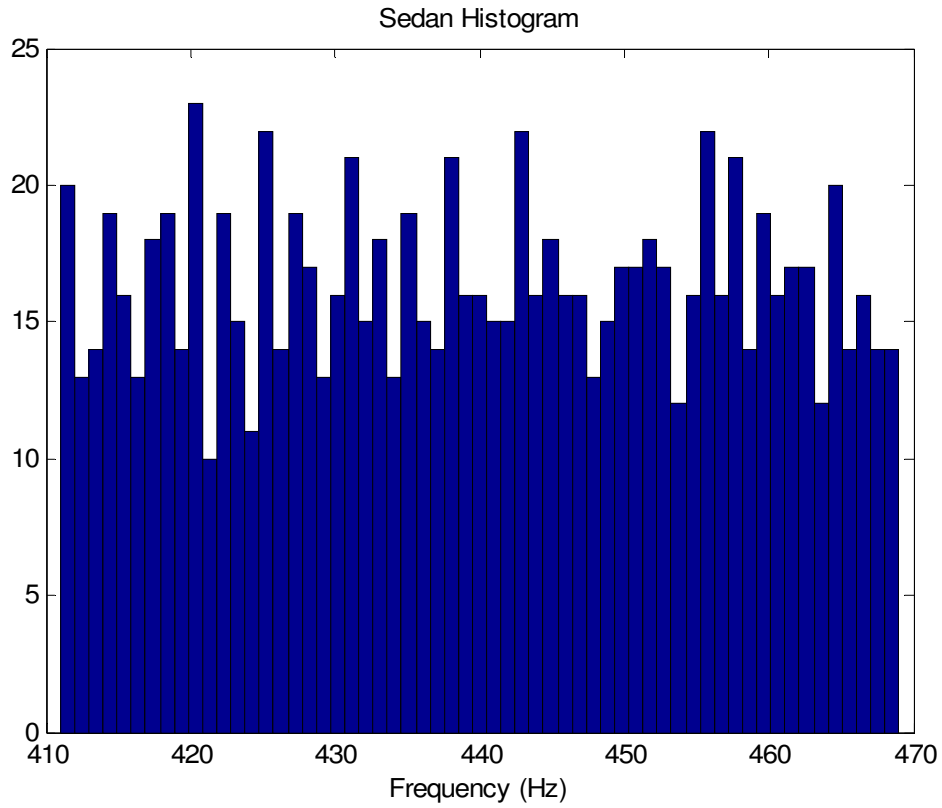

Figure 10: Histogram of our sedan data.

For the sedans, we had we can see that a noticeable peak arises between 420 Hz and 421 Hz, with others between 425 Hz and 426 Hz, 442 Hz and 443 Hz, and 456 Hz and 456 Hz.

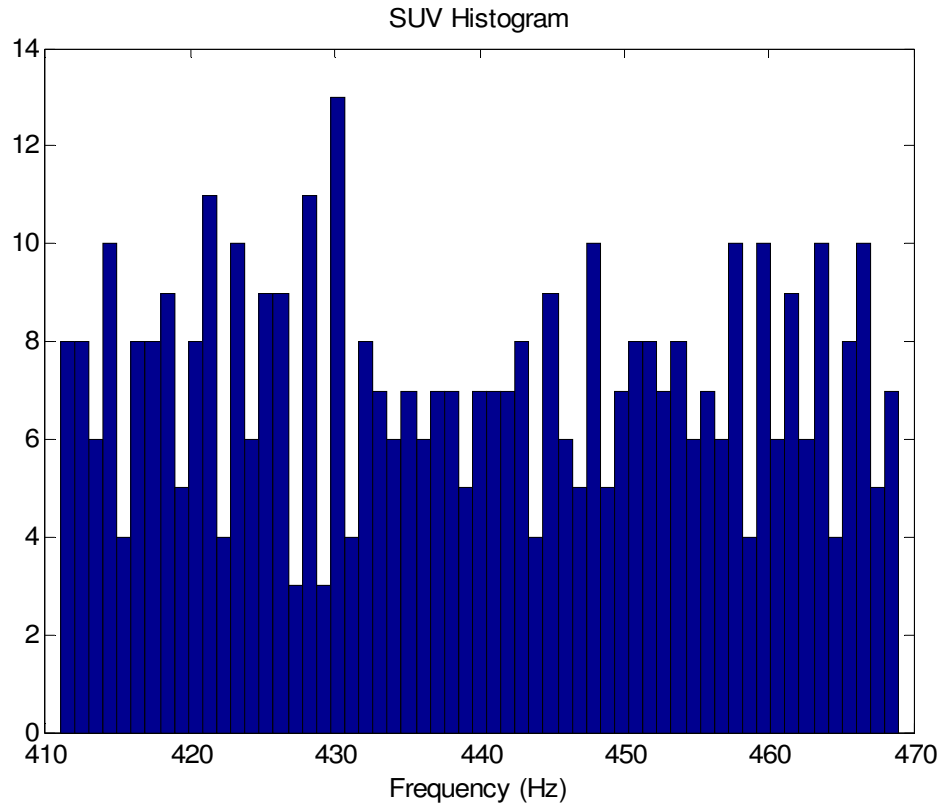

Figure 11: Histogram of our SUV data.

For the SUVs, the most noticeable peak was between 430 Hz and 431 Hz, followed by peaks between 428 Hz and 429 Hz, and 421 Hz and 422 Hz.

From these histograms, we can spot that while there is quite some noise in our signals, there are some noticeable peaks emerging, especially around 430 Hz, which we had noted earlier in our example data sets. The height of the peak indicates that it frequently occurs in our data sets, indicating that this is a repeating feature of our signals, instead of random noise artifacts.

5. Conclusion

From our data, we can conclude that this method of determining car types by frequencies is promising. While our signal was filled with noise, most likely either generated by the apparatus, or from the cars being sampled, the fact that we were able to

retrieve patterns in our data sets implies that the method does produce measurable results, even in extremely noisy environments. The cars were reflecting signals dissimilar enough from our original signal that they could not be considered errors in our signal itself. Furthermore, because the SUVs and sedans were producing different peaks in our spectrums, we can use this to determine if a car that is passing by is large or small, a first step in determining car type, with a moderate level of confidence. By refining this method further, we should be able to determine more signature frequencies that can be used to refine our determination of car types.

6. Future work

For future work, there is much that can be done. First and foremost, as always when working in noisy environments, is to reduce the impact of noise on our experiment. The issue of the noise coming from the apparatus is a major problem, especially with the issue of grounding. Plugging the apparatus in removed at least part of the problem, but further action may be needed to remove possible noise in the lower end of the spectrum. Secondly, a laptop that does not produce a "clicking" noise would be better, as noise of the hardware parts operating wouldn't interfere with the broadcasting of the signal. Finally, a higher frequency signal may be more appropriate for use with the cars. As seen in the spectrograms, the frequency spectrums of the cars reach down to the lower end perhaps even to the 440 Hz region we were using. Thus, a frequency at least higher than 1000 Hz may be more appropriate, to get our signal above the frequency spectrum of the cars. Next, data collection of more car types is required. Currently we mostly have data collected from sedans as SUVs, as those were the most common cars that were recorded by our setup. Other cars, including minivans, trucks, station wagons, were used too, but

their sample sizes were too small to derive conclusions for exclusively those groups. To counter this, more data for those cars would be appropriate, to get a better idea of their frequency spectrums. For the experimental procedure itself, further refinement is possible. For this project, position of the apparatus was determined by where along the sidewalk the apparatus would not roll away, instead of a structured setup. Using set distances would allow for more rigorous data collection for distances. Further more, we could simultaneously collect data on car distances by starting the signal as the car passes by, instead of beforehand, which was done in this project. Finally data analysis could also be improved. The FFTs produced in this paper were done using MATLAB's basic FFT routine. MATLAB also provides other methods of discrete Fourier transforms, including the Welch power method, and the MUSIC method. Using these methods might be better at handling noise, and might single out specific frequencies more readily. Furthermore, filtering could be improved by reducing the amount of ripple, which might give false results in frequency peaks.

7. Acknowledgements

I would like to thank Professor Hinders for his copious help and wisdom with researching and life in general, without which I never would have been able to accomplish much of what I did this year. I would also like to thank Crystal Bertoncini, who helped greatly in pinpointing the variables that I needed to isolate, in order to make my experimental setup successful.

Appendix A

A.1 Data Sets

Here, we display the spectrograms, FFTs, and time-domain plots of our other samples, to show the distribution of frequencies of our reflections. Each figure has the type of car, which lane (near or far) it was in according to our diagram, the distance from the street to the apparatus, and the angle with which we were at with respect to the road.

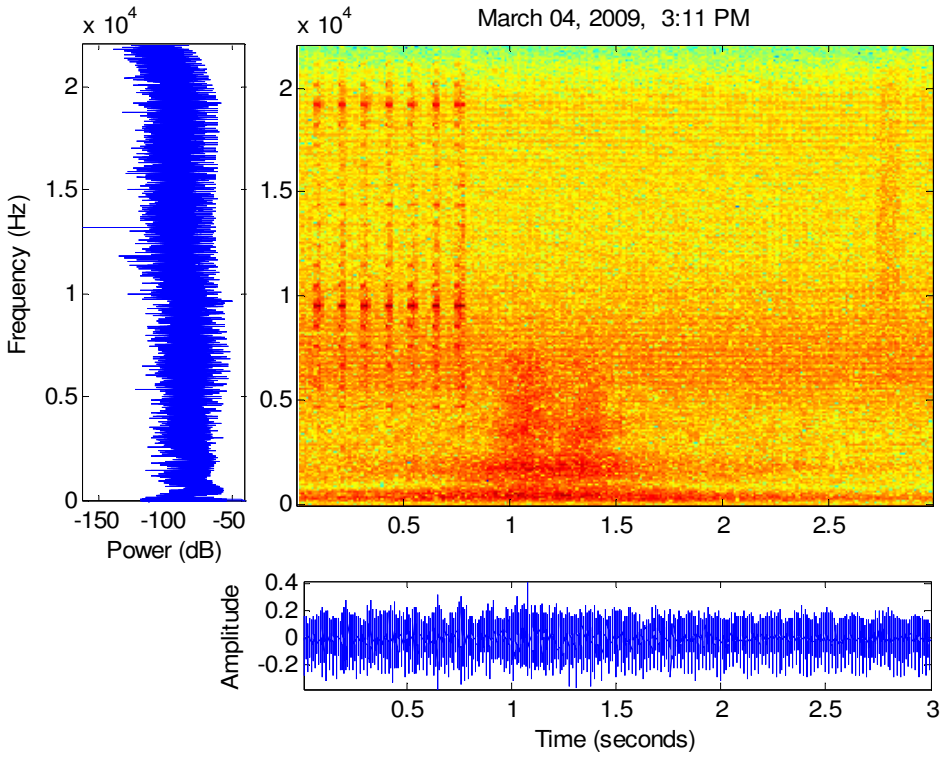

Figure 12: Minivan, near lane, 11 ft. 2 in. from road, 90° with respect to road.

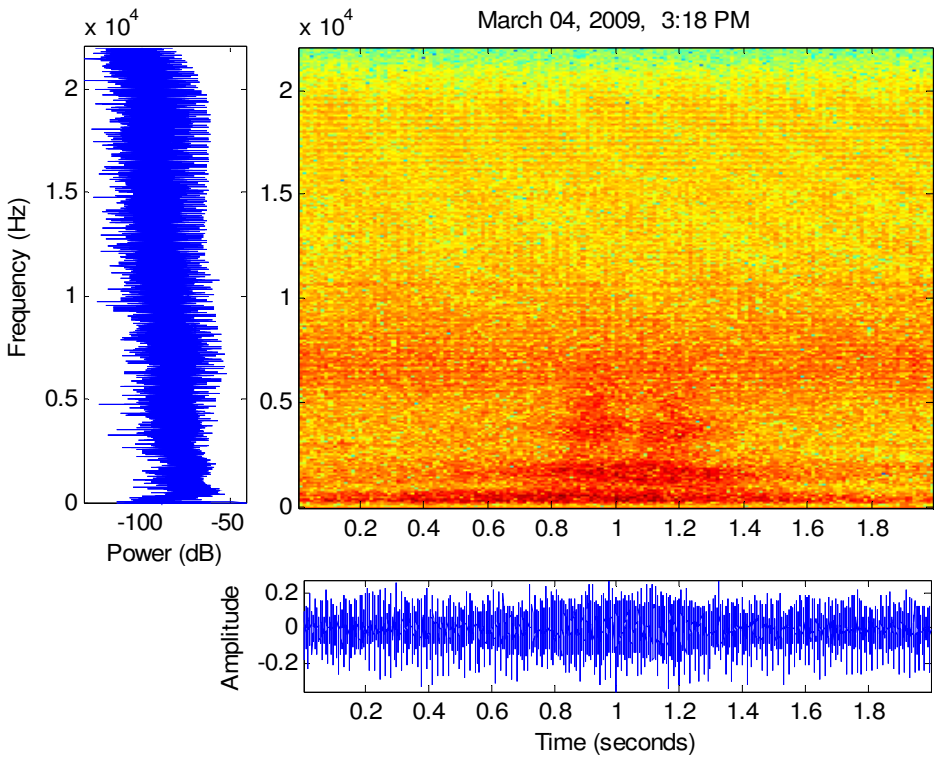

Figure 13: Sedan, near lane, 11 ft. 2 in. from road, 90° with respect to road.

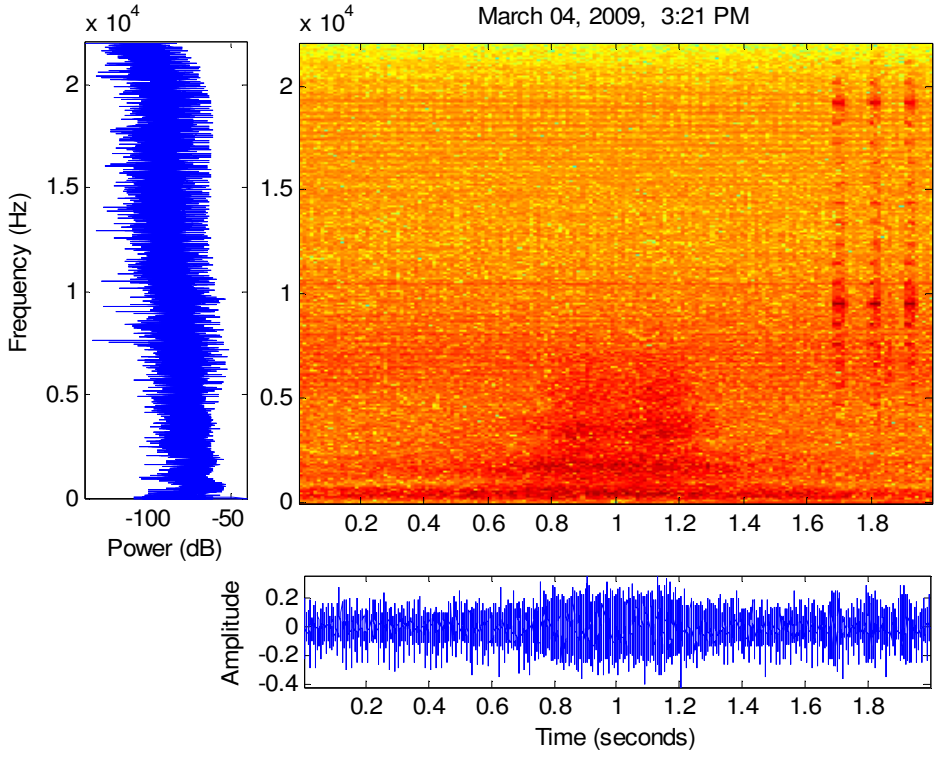

Figure 14: Sedan, near lane, 11 ft. 2 in. from road, 90° with respect to road.

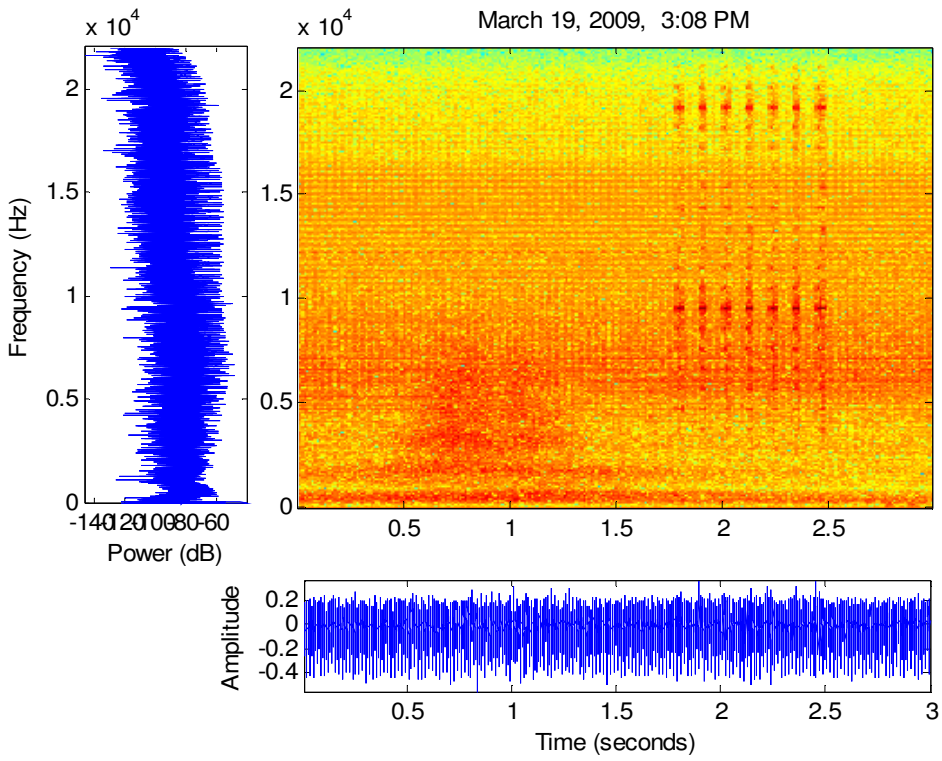

Figure 15: Sedan, far lane, 10 ft. from road, 90° with respect to road.

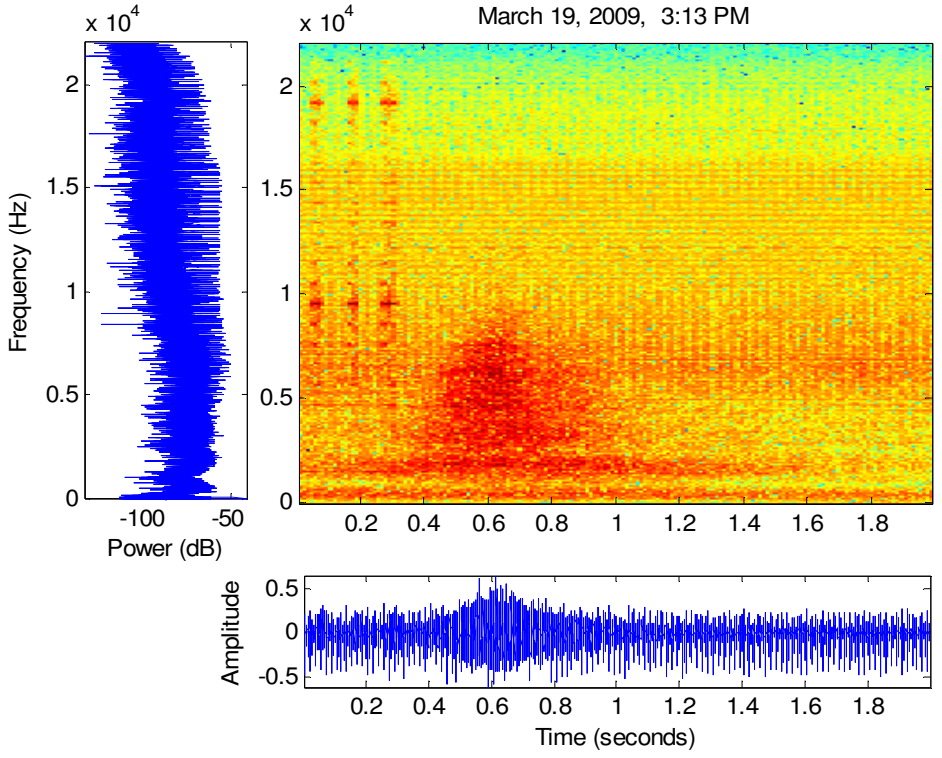

Figure 16: Sedan, far lane, 10 ft. from road, 90° with respect to road.

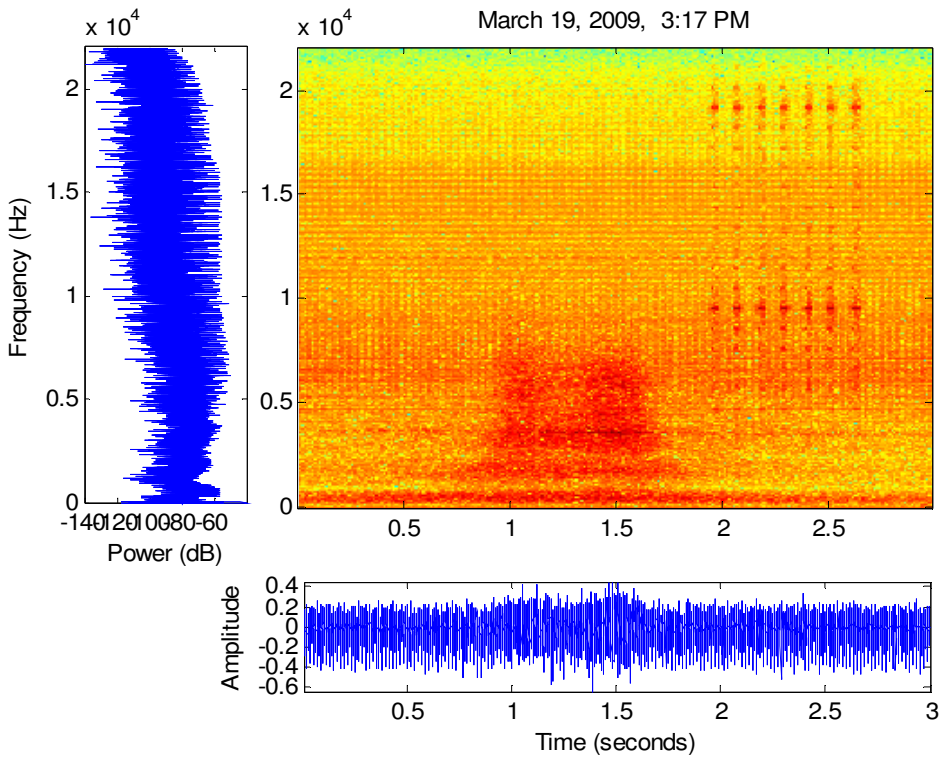

Figure 17: Truck, near lane, 10 ft. from road, 90° with respect to road.

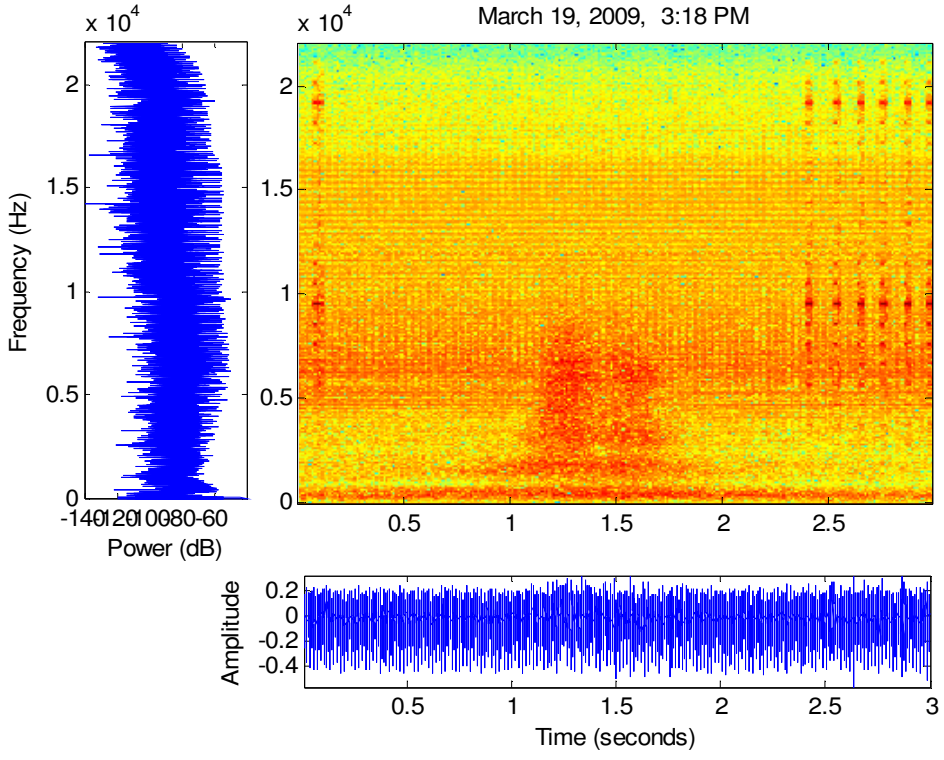

Figure 18: SUV, near lane, 10 ft. from road, 90° with respect to road.

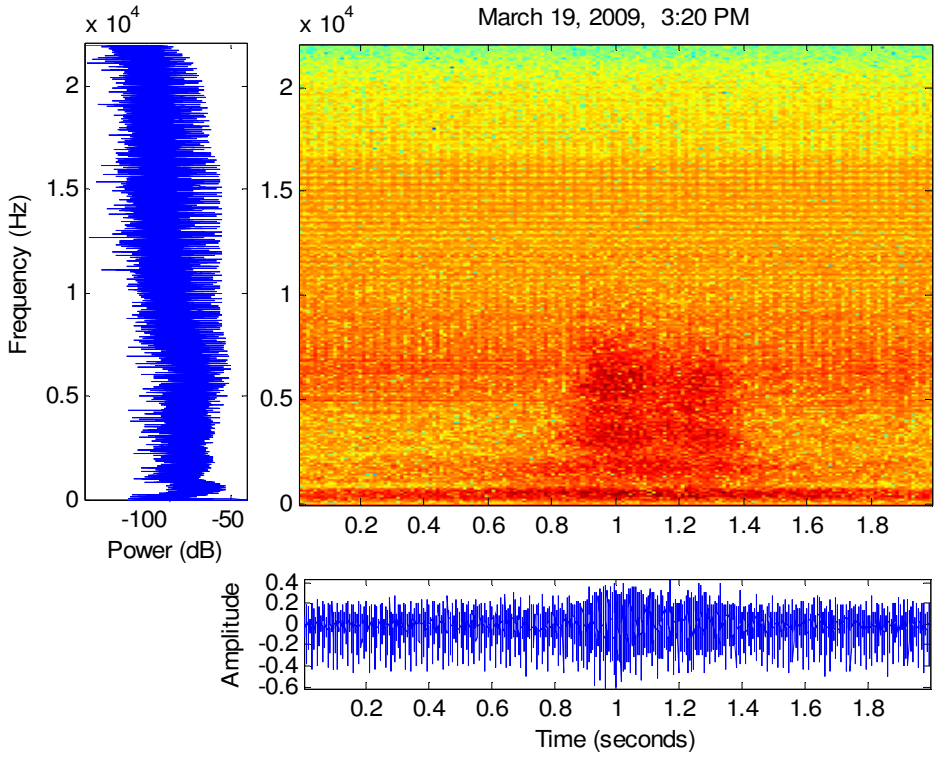

Figure 19: SUV, near lane, 10 ft. from road, 90° with respect to road.

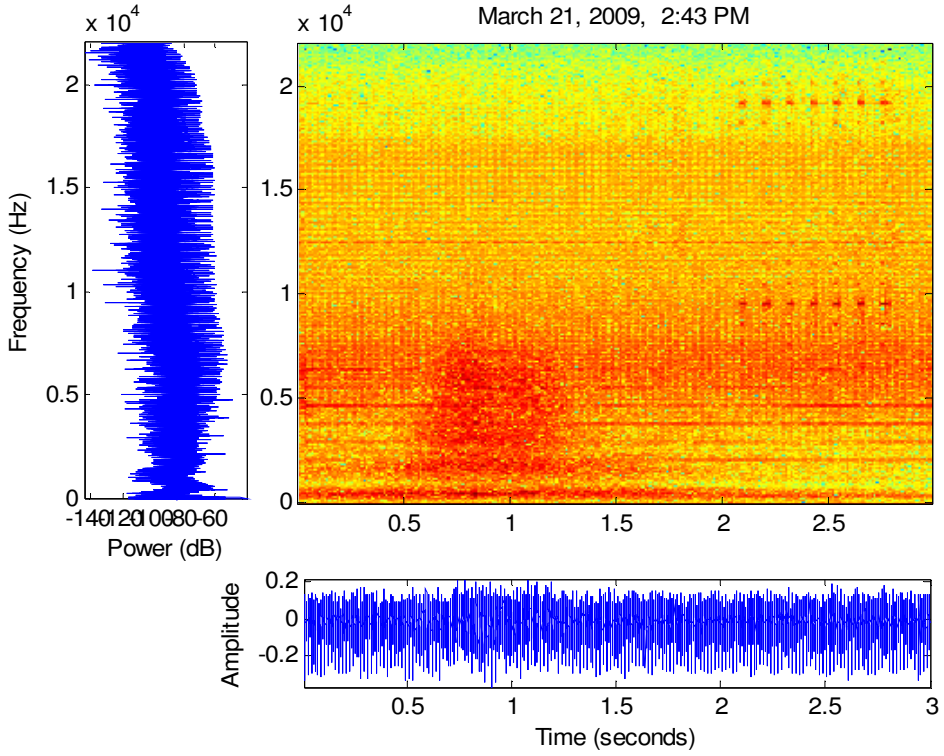

Figure 20: Truck, far lane, 12 ft. 3 in. from road, 90° with respect to road.

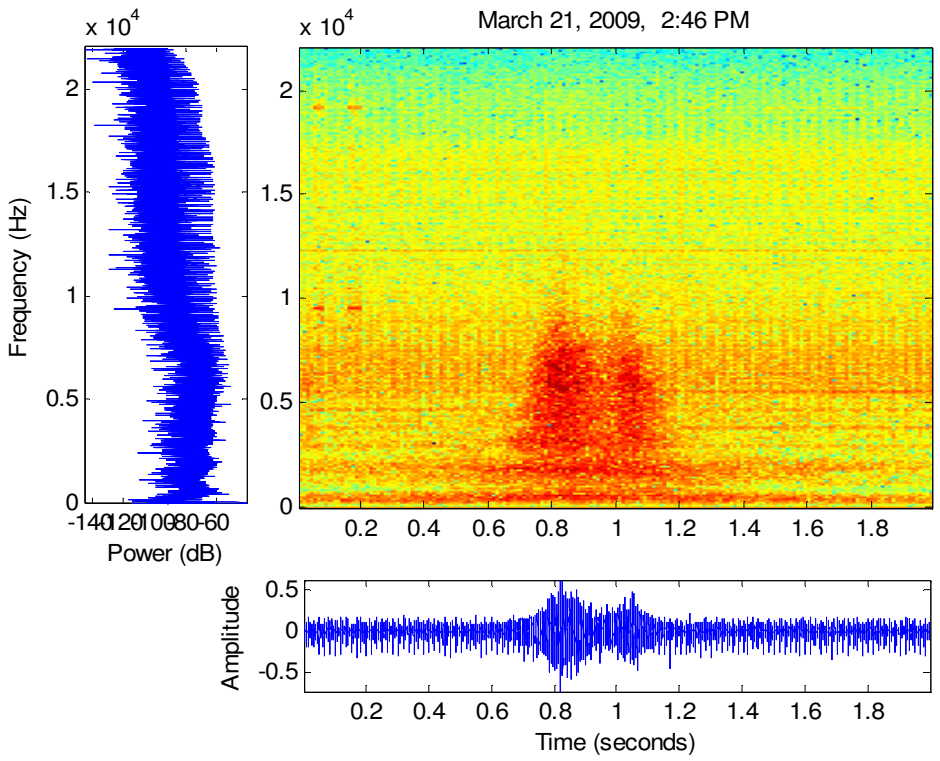

Figure 21: Sedan, near lane, 12 ft. 3 in. from road, 90° with respect to road.

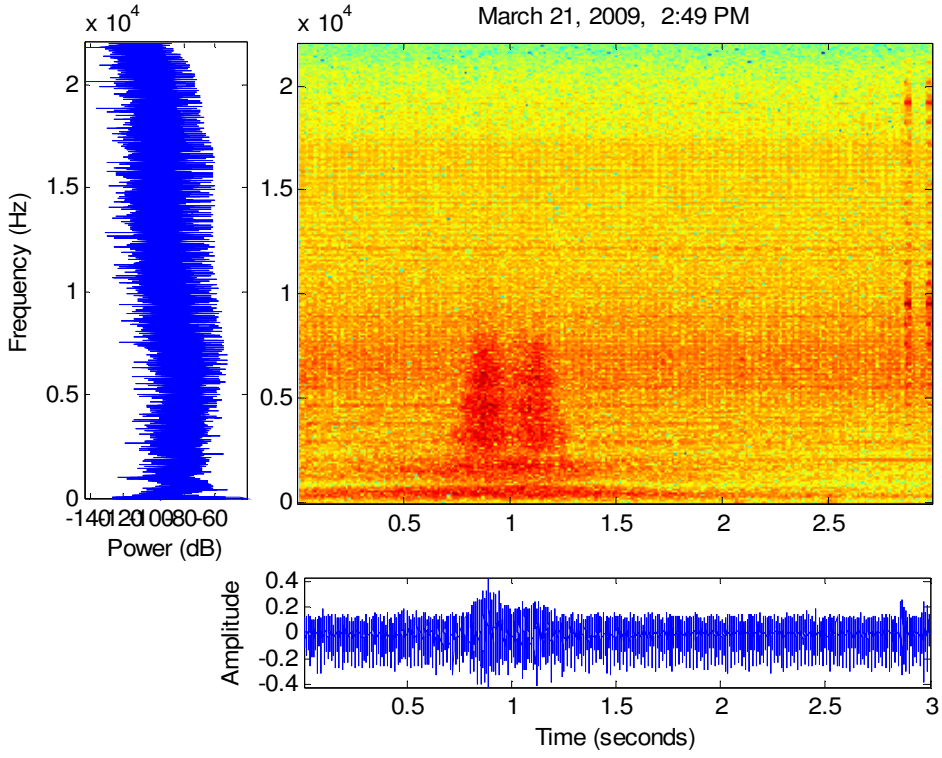

Figure 22: SUV, near lane, 12 ft. 3 in. from road, 90° with respect to road.

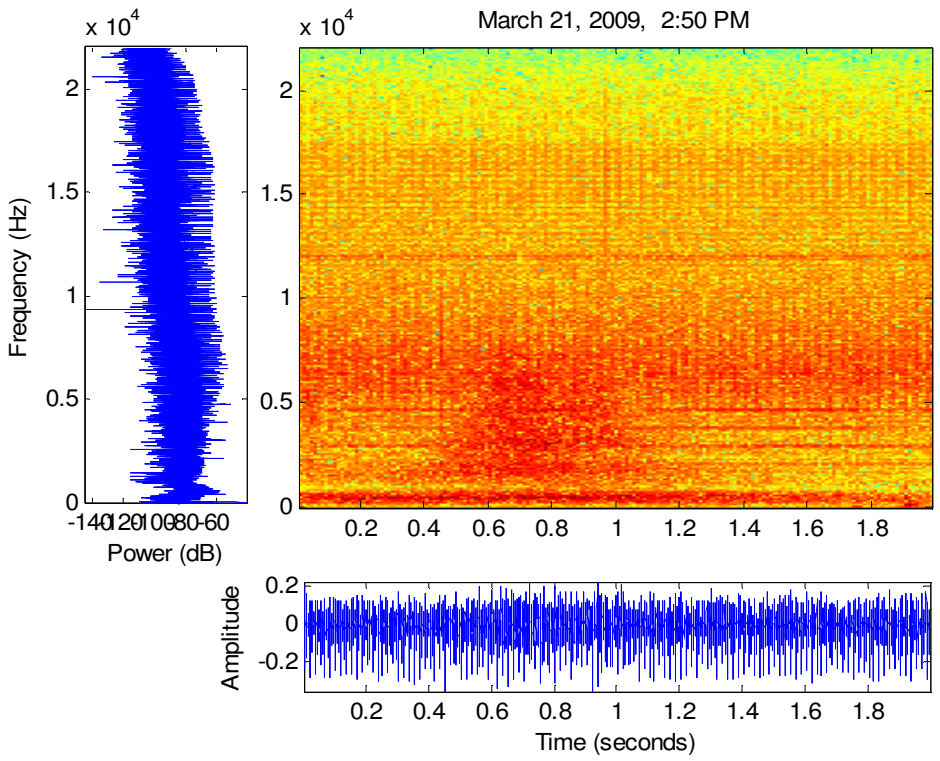

Figure 23: Sedan, near lane, 12 ft. 3 in. from road, 90° with respect to road.

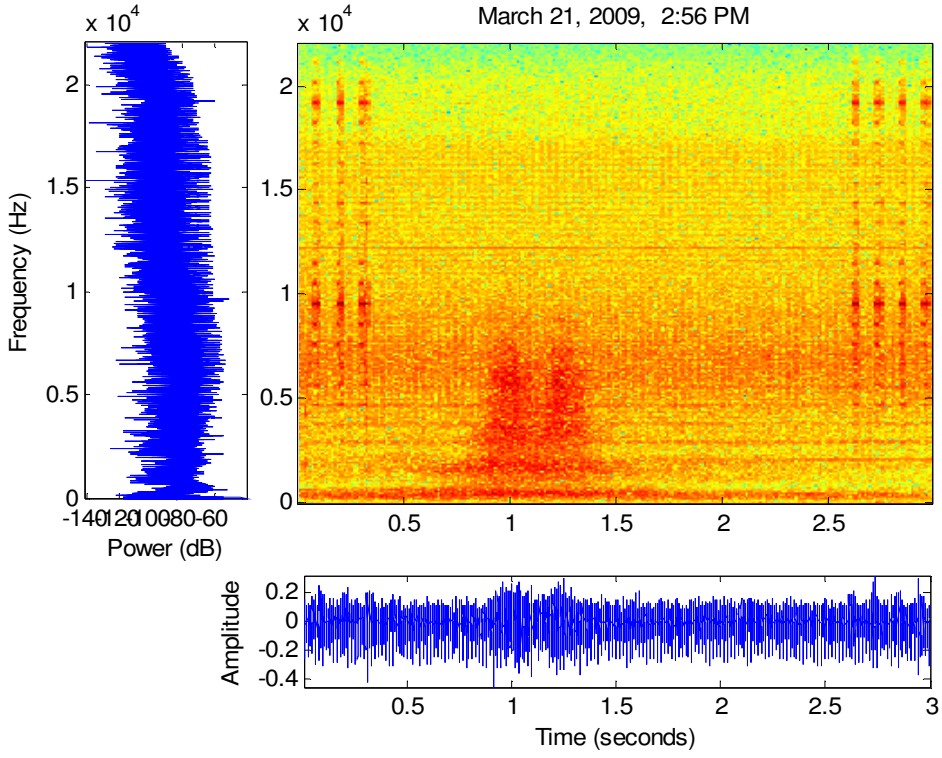

Figure 24: Sedan, near lane, 12 ft. 3 in. from road, 90° with respect to road.

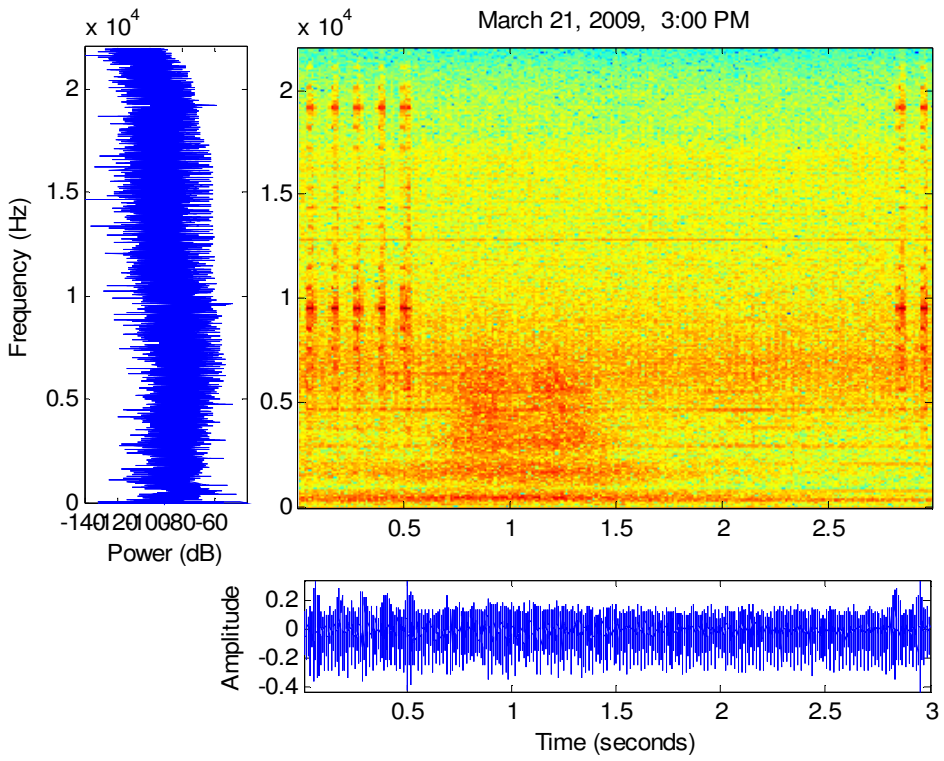

Figure 25: Truck, far lane, 12 ft. 3 in. from road, 90° with respect to road.

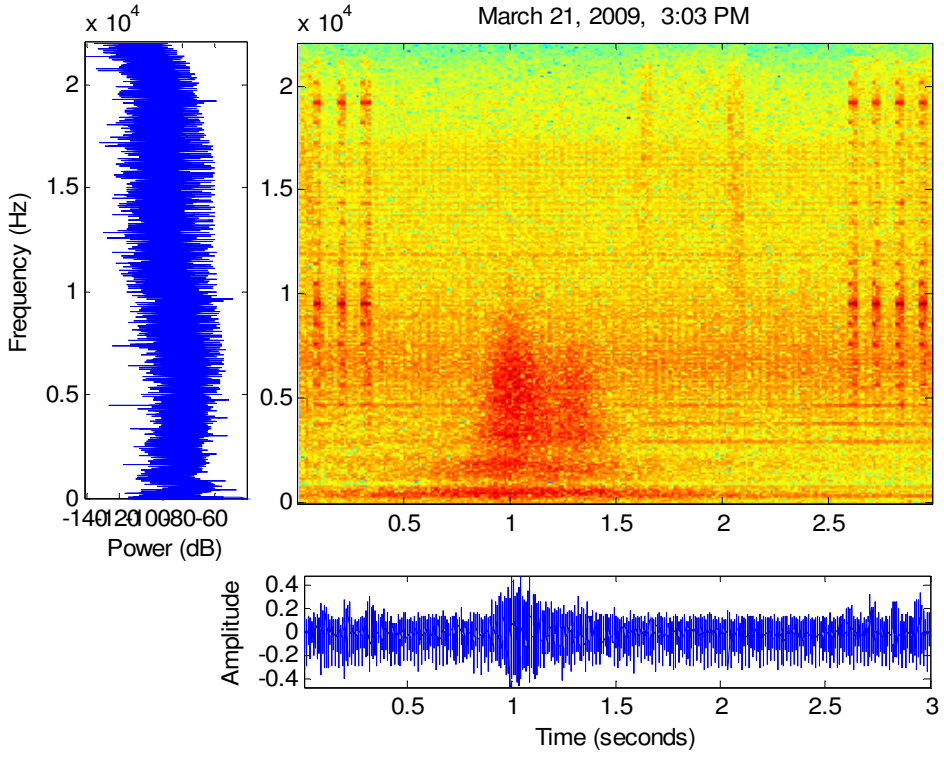

Figure 26: Truck, near lane, 12 ft. 3 in. from road, 90° with respect to road.

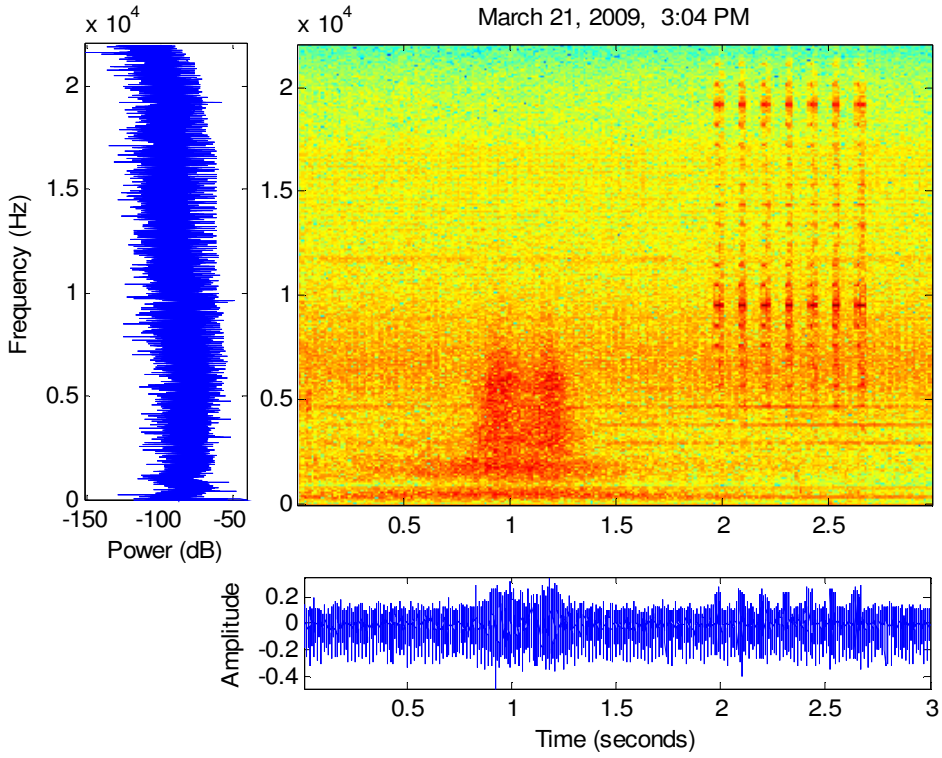

Figure 27: SUV, near lane, 12 ft. 3 in. from road, 90° with respect to road.

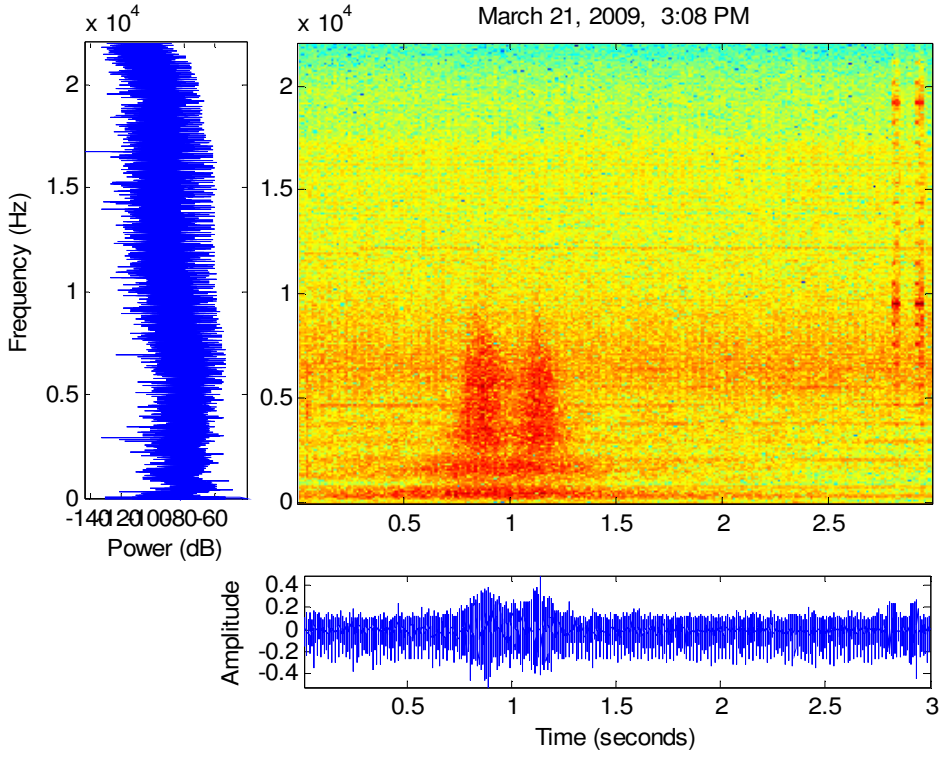

Figure 28: Truck, near lane, 12 ft. 3 in. from road, 90° with respect to road.

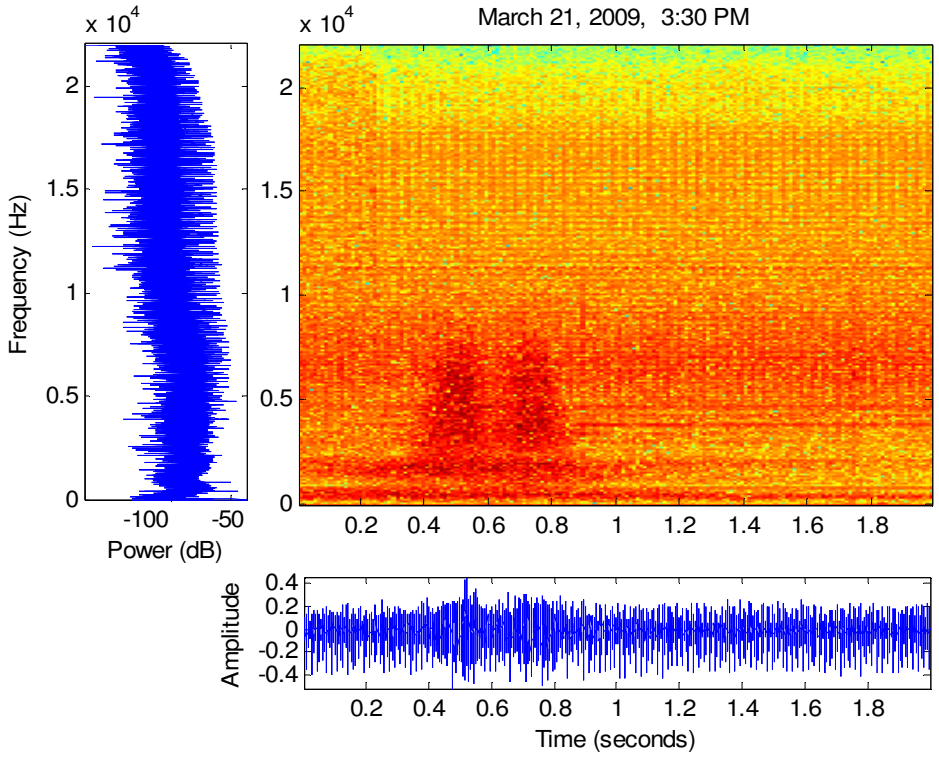

Figure 29: Sedan, near lane, 9 ft. 9 in. from road, 90° with respect to road.

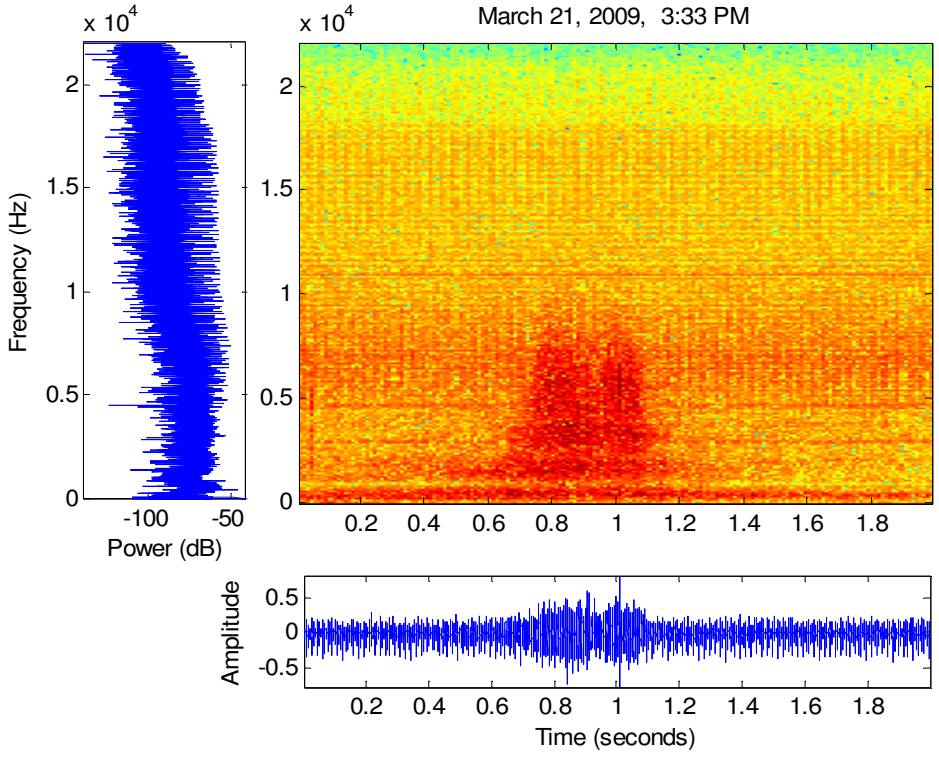

Figure 30: SUV, near lane, 9 ft. 9 in. from road, 90° with respect to road.

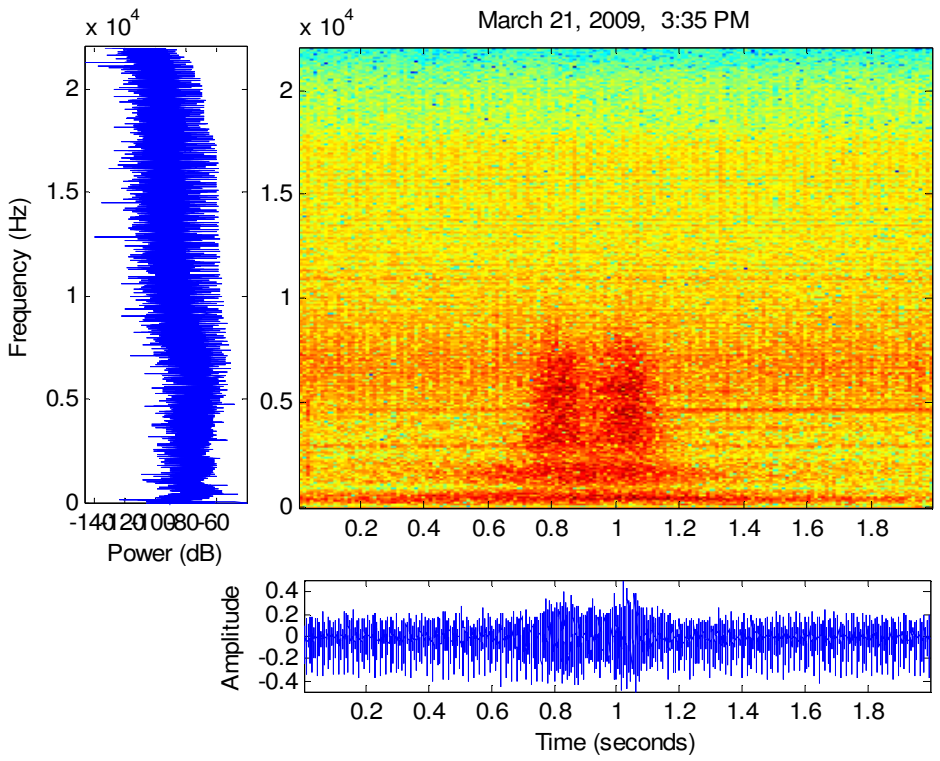

Figure 31: Sedan, near lane, 9 ft. 9 in. from road, 90° with respect to road.

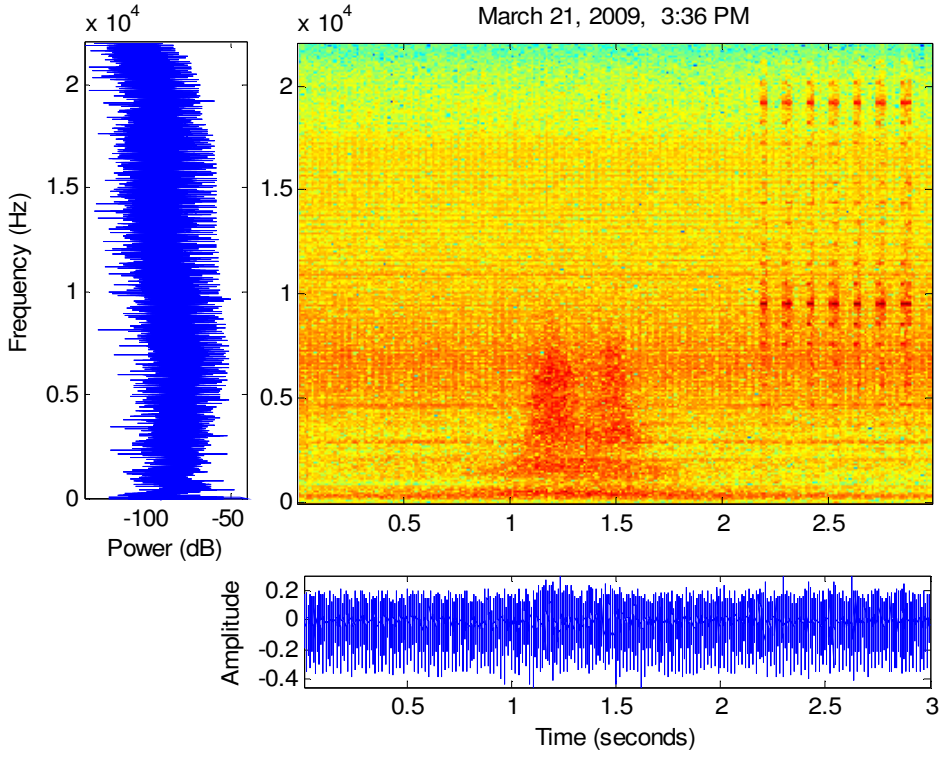

Figure 32: Sedan, near lane, 9 ft. 9 in. from road, 90° with respect to road.

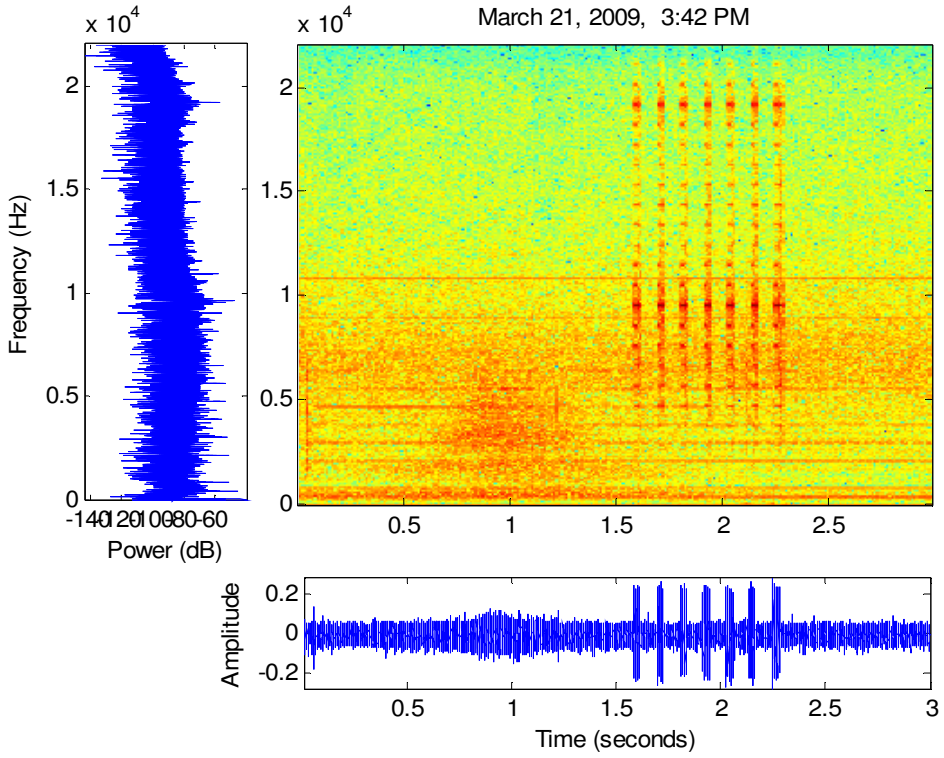

Figure 33: Sedan, far lane, 9 ft. 9 in. from road, 90° with respect to road.

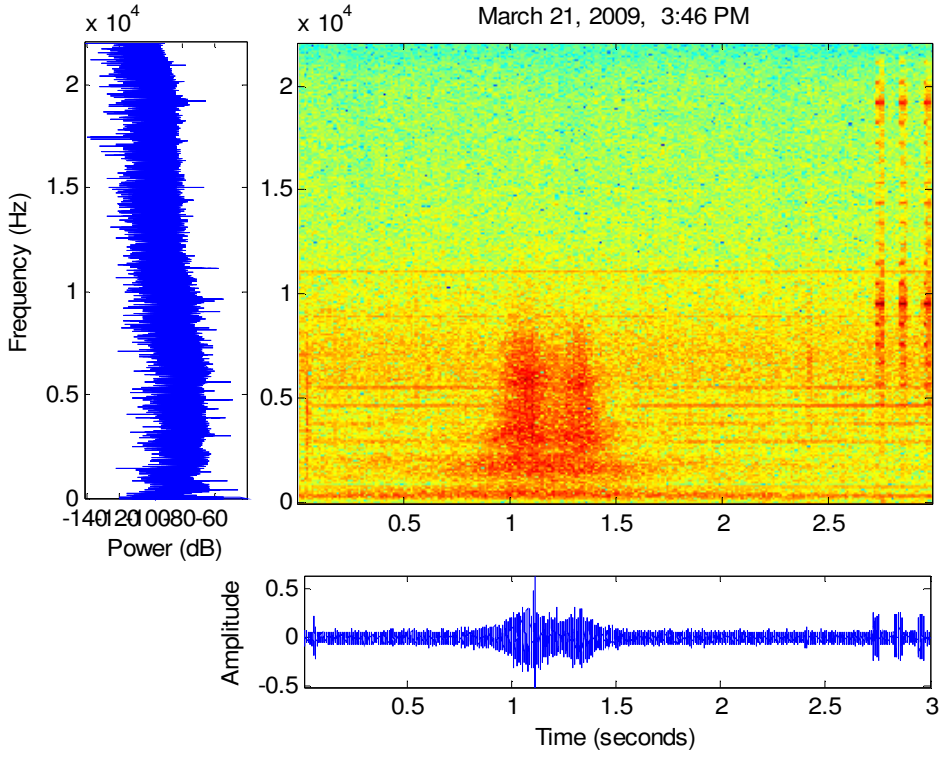

Figure 34: Sedan, near lane, 9 ft. 9 in. from road, 90° with respect to road.

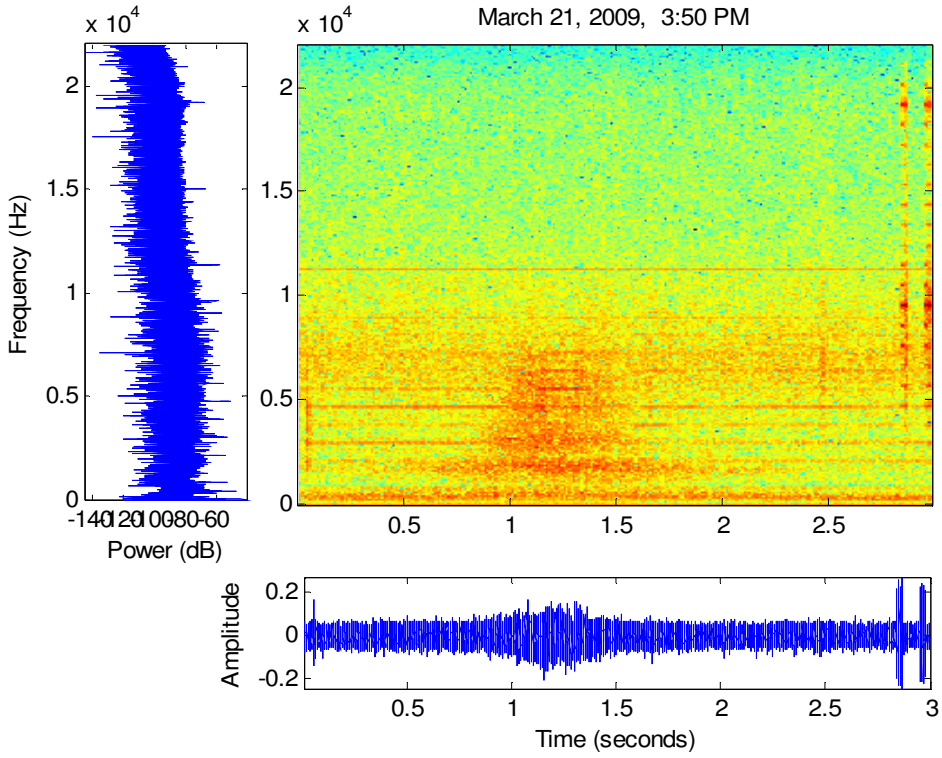

Figure 35: Sedan, far lane, 9 ft. 9 in. from road, 90° with respect to road.

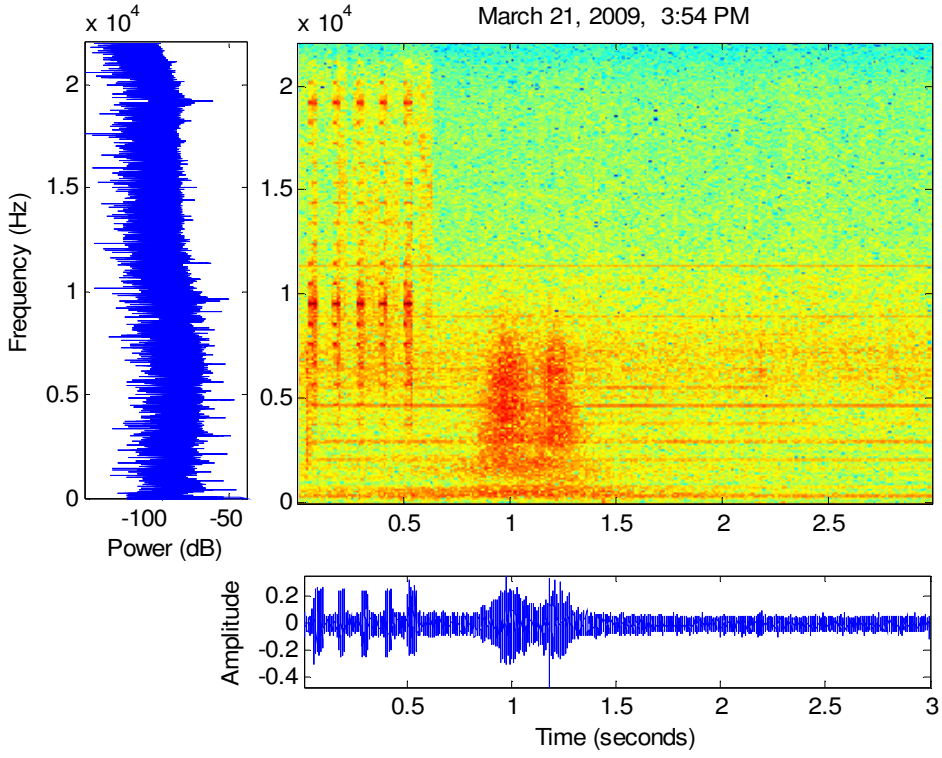

Figure 36: Sedan, near lane, 9 ft. 9 in. from road, 90° with respect to road.

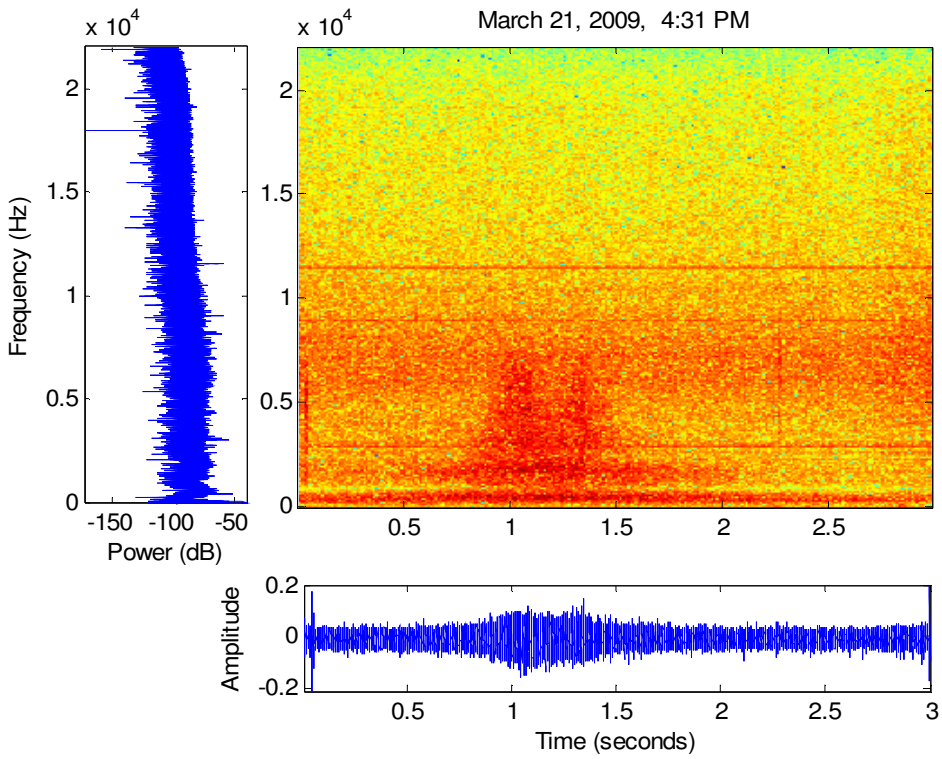

Figure 37: Sedan, far lane, 6 ft. from road, 90° with respect to road.

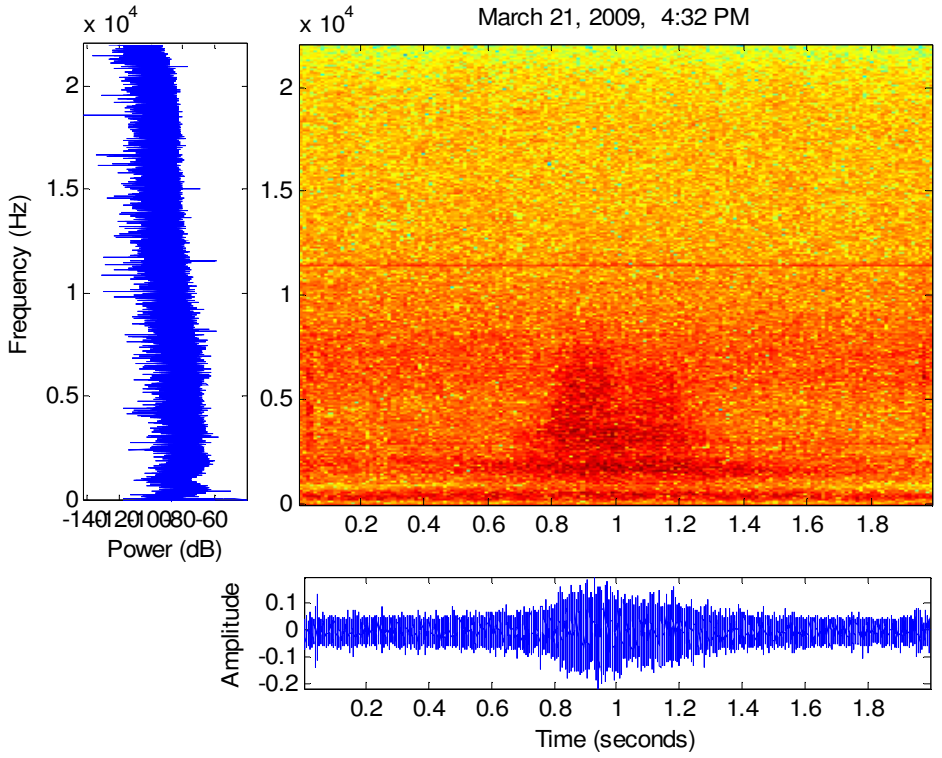

Figure 38: Station wagon, near lane, 6 ft. from road, 90° with respect to road.

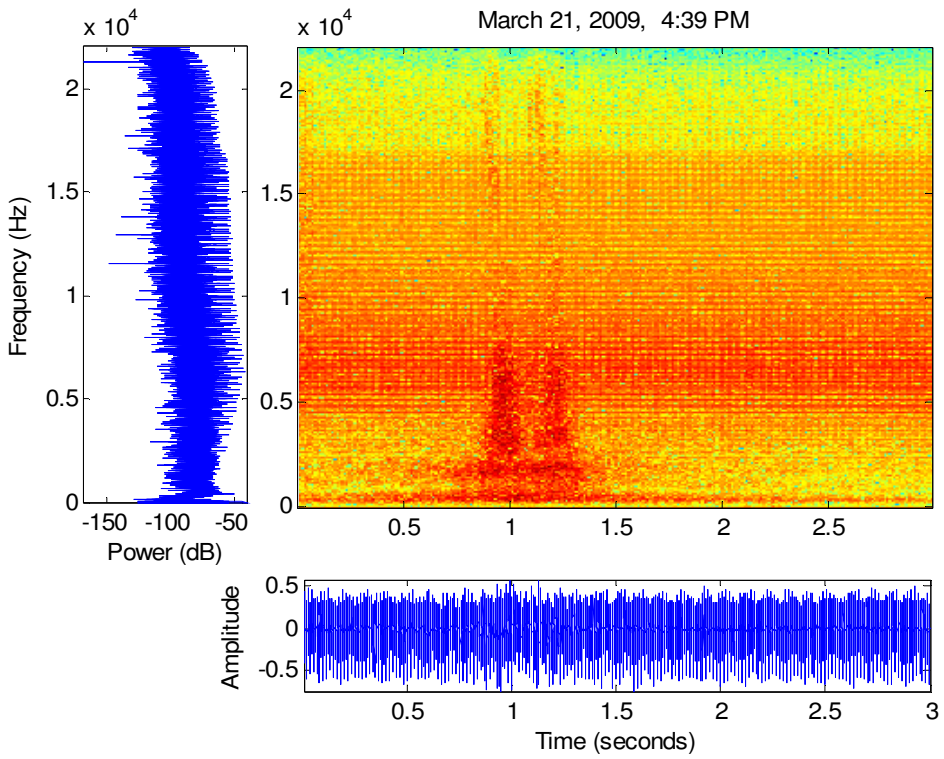

Figure 39: Sedan, near lane, 6 ft. from road, 90° with respect to road.

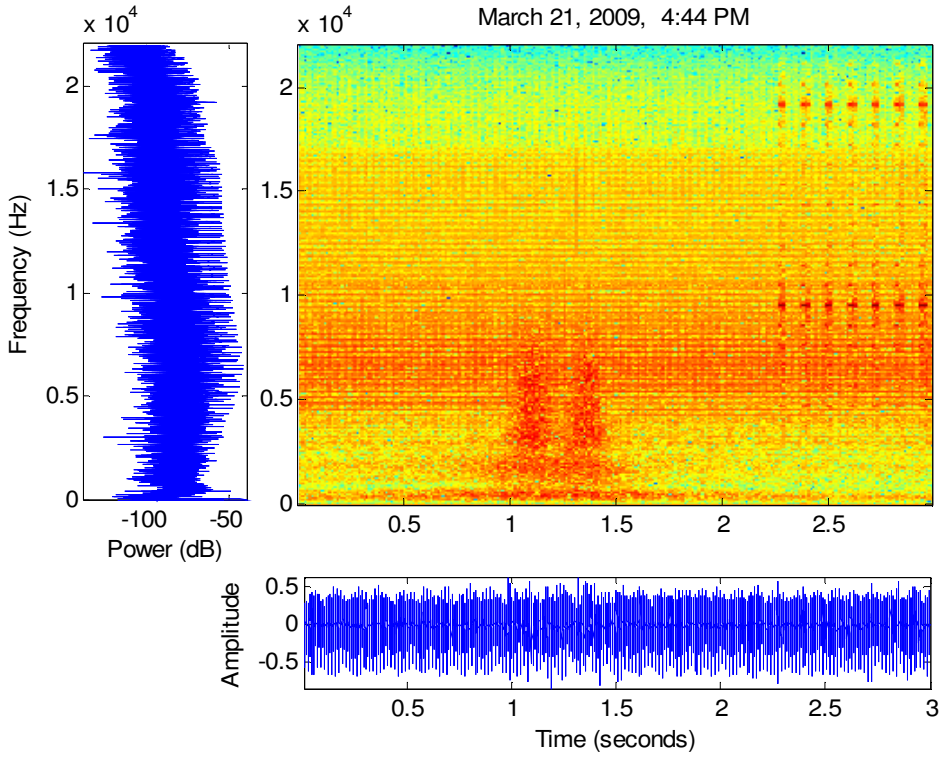

Figure 40: Sedan, near lane, 6 ft. from road, 90° with respect to road.

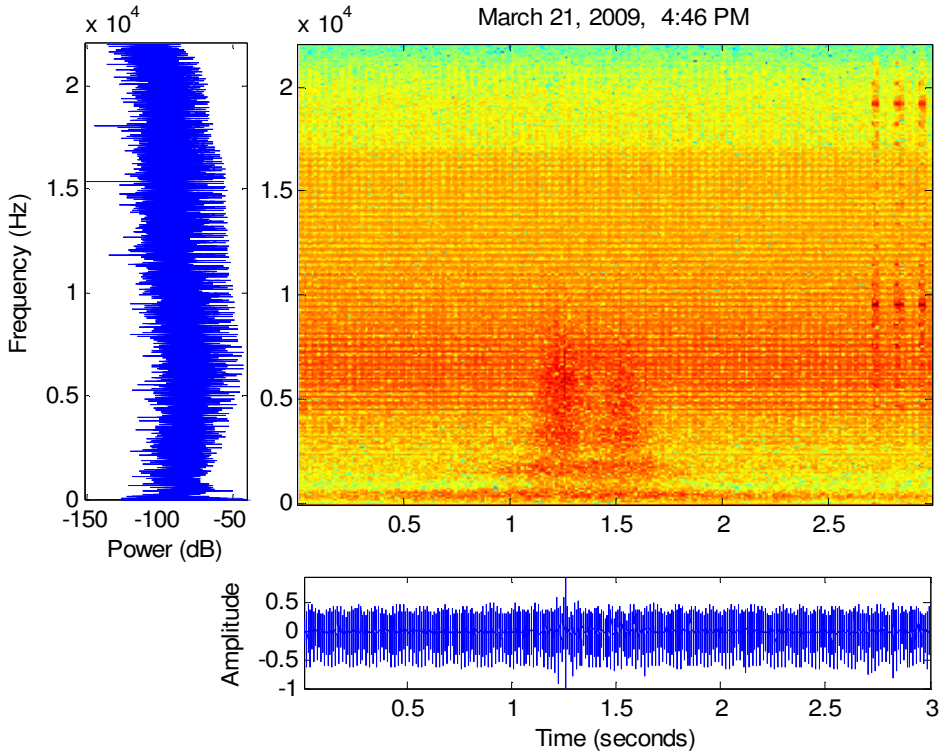

Figure 41: Sedan, near lane, 6 ft. from road, 90° with respect to road.

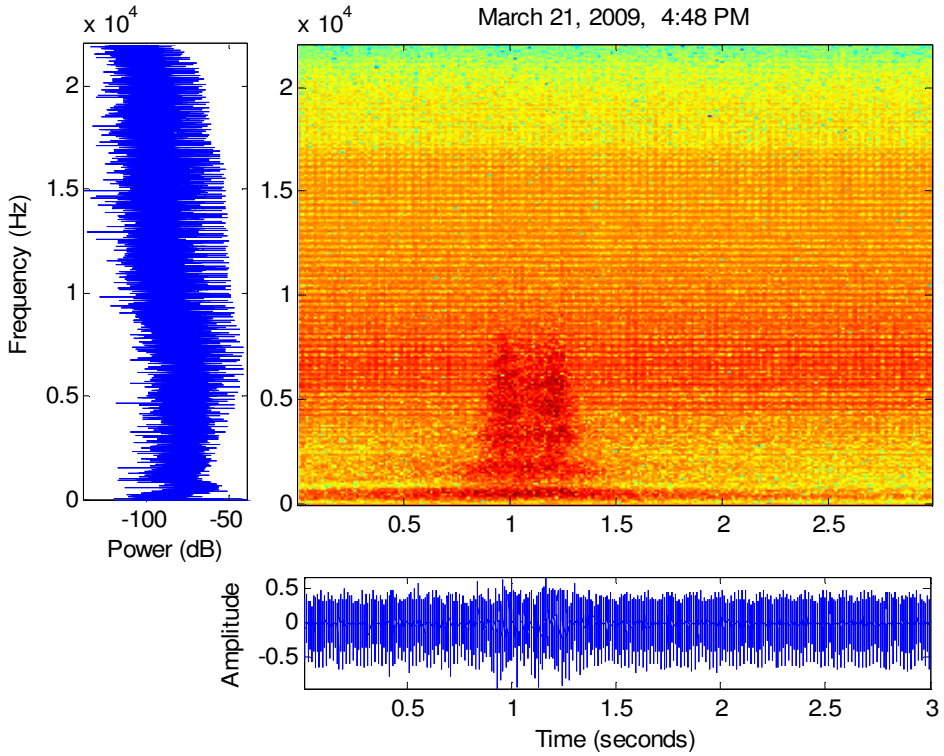

Figure 42: SUV, near lane, 6 ft. from road, 90° with respect to road.

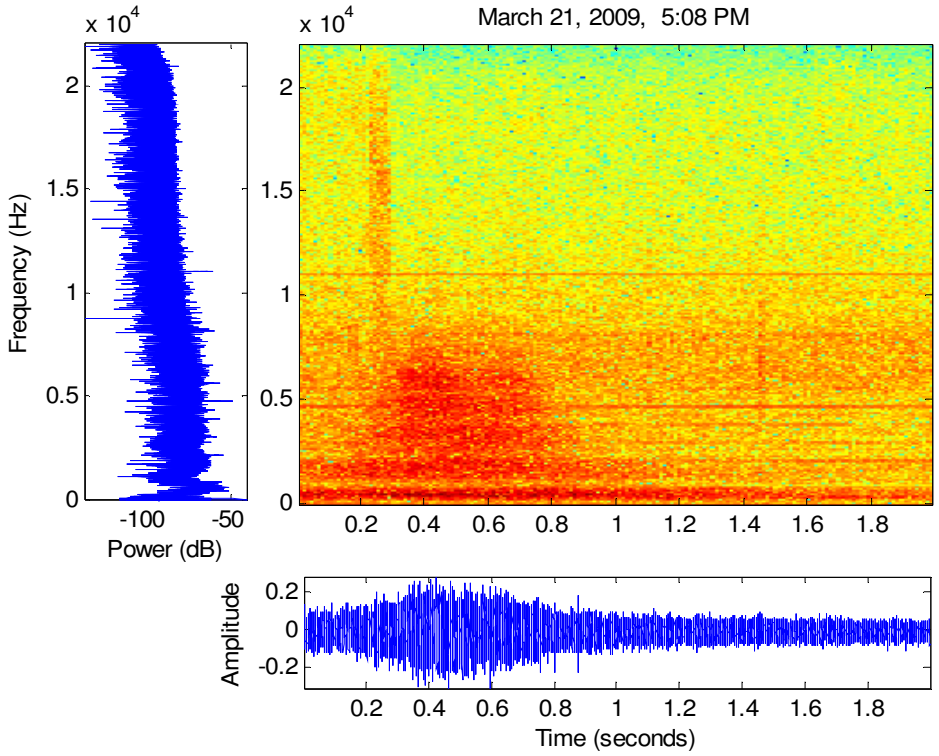

Figure 43: SUV, far lane, 14 ft. 6 in. from road, 90° with respect to road.

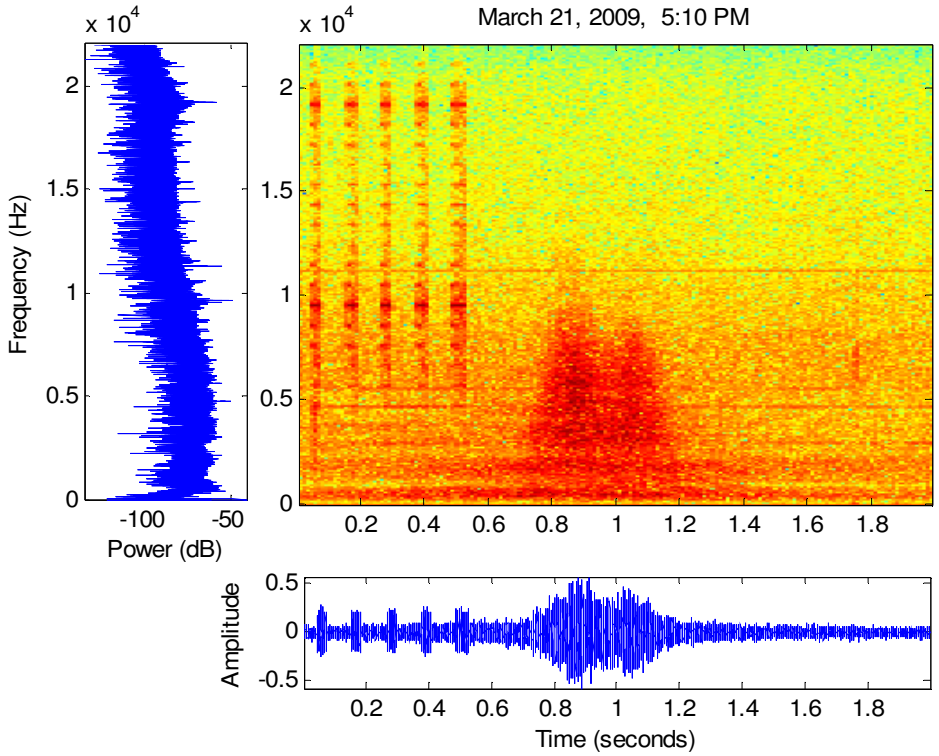

Figure 44: Sedan, near lane, 14 ft. 6 in. from road, 90° with respect to road.

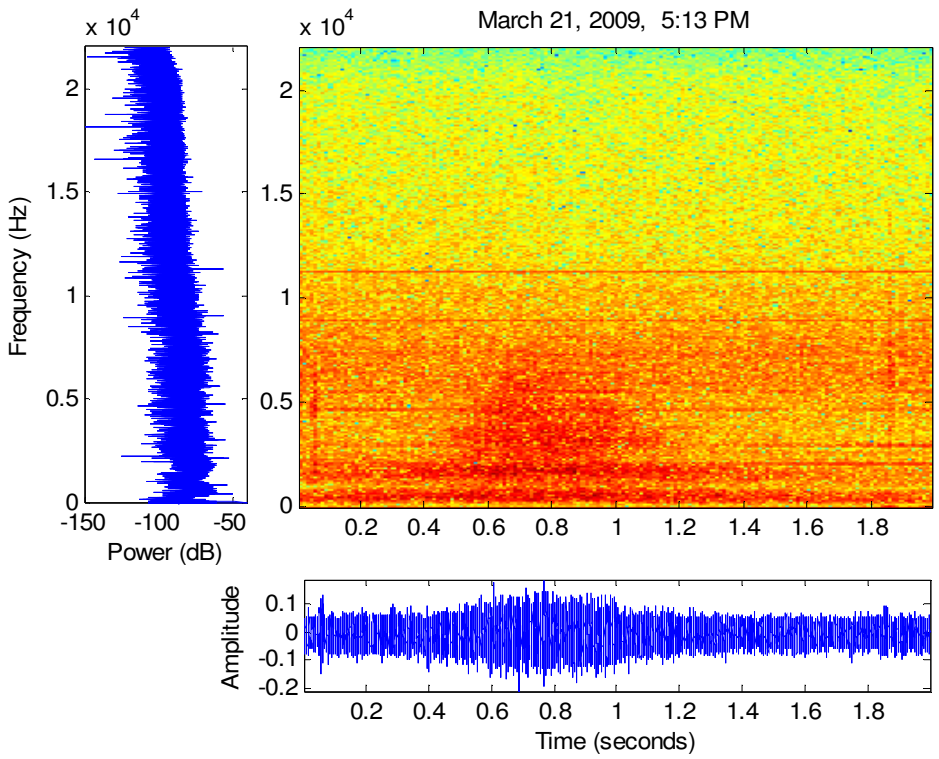

Figure 45: Minivan, far lane, 14 ft. 6 in. from road, 90° with respect to road.

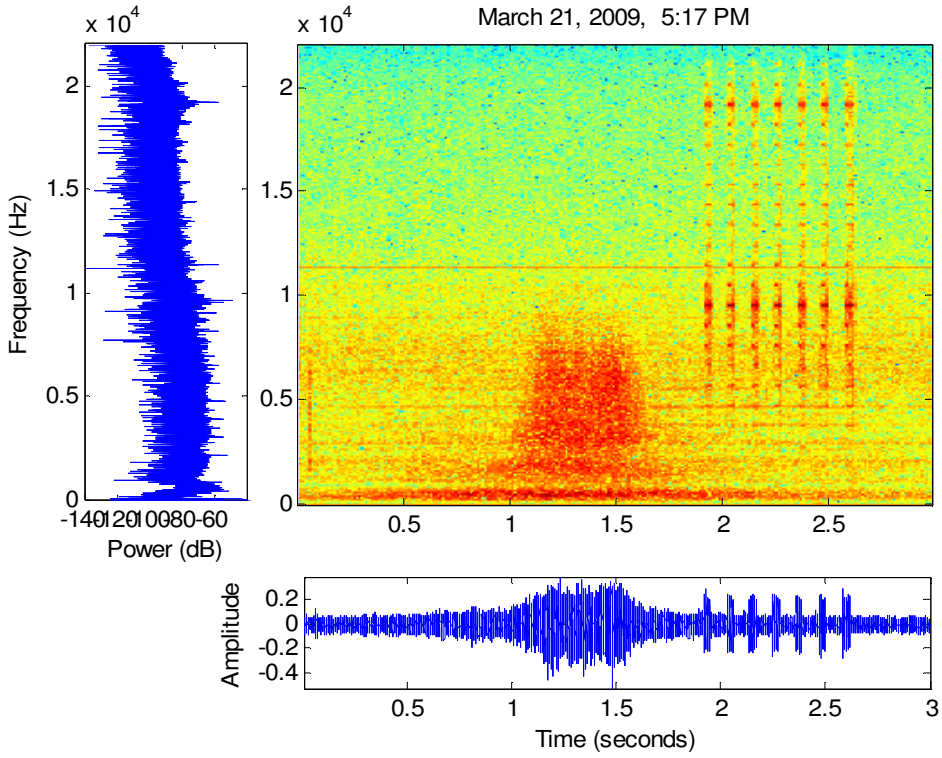

Figure 46: SUV, near lane, 14 ft. 6 in. from road, 90° with respect to road.

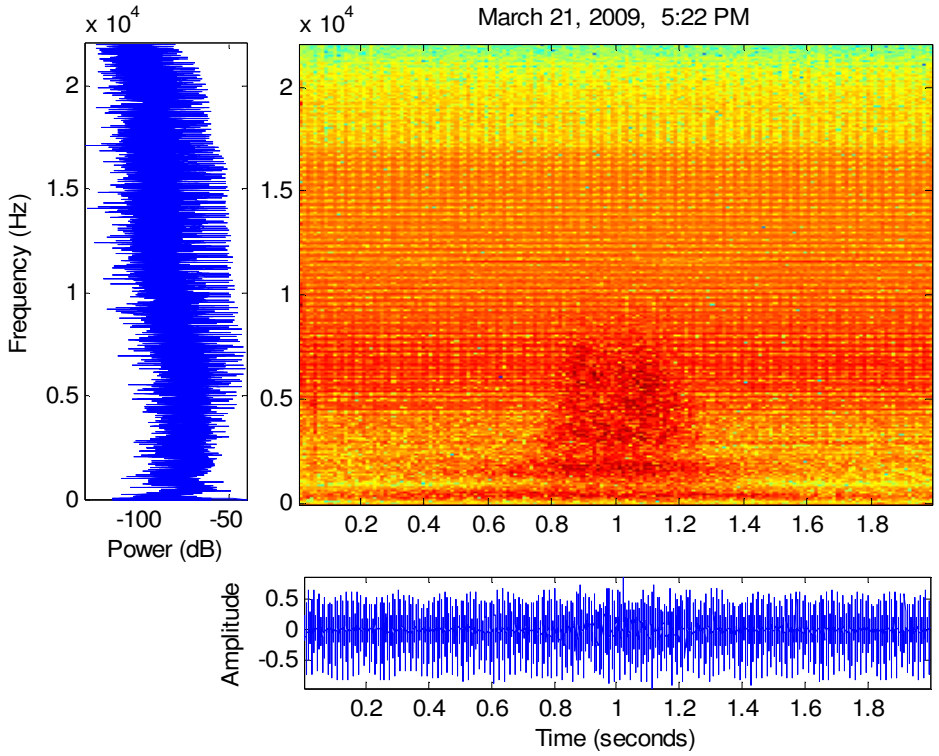

Figure 47: Station wagon, near lane, 14 ft. 6 in. from road, 90° with respect to road.

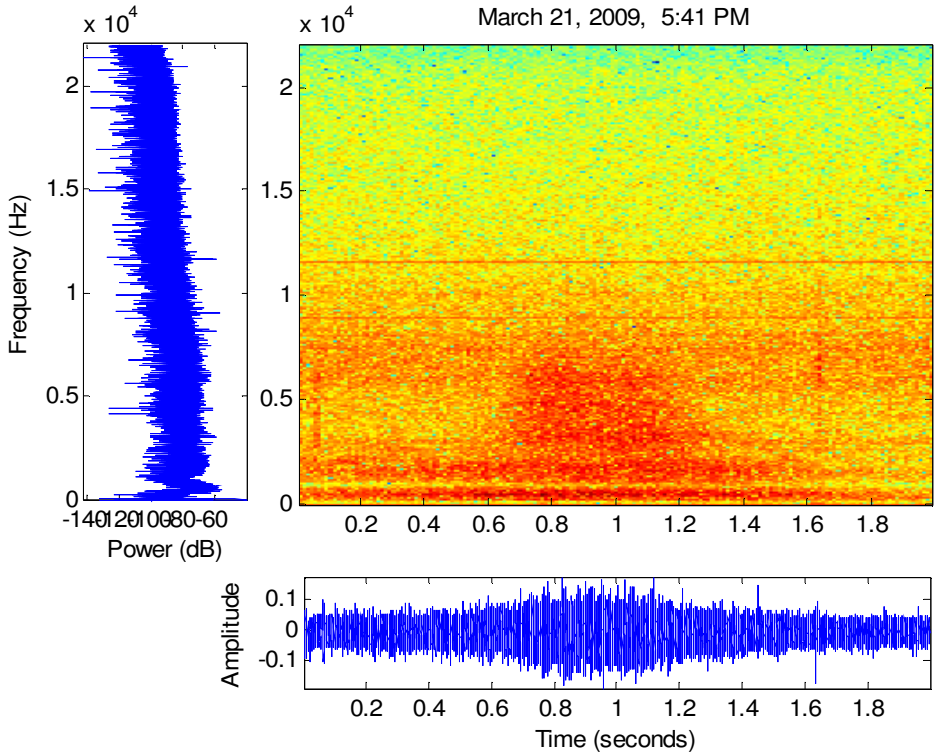

Figure 48: Truck, far lane, 24 ft. from road, 90° with respect to road.

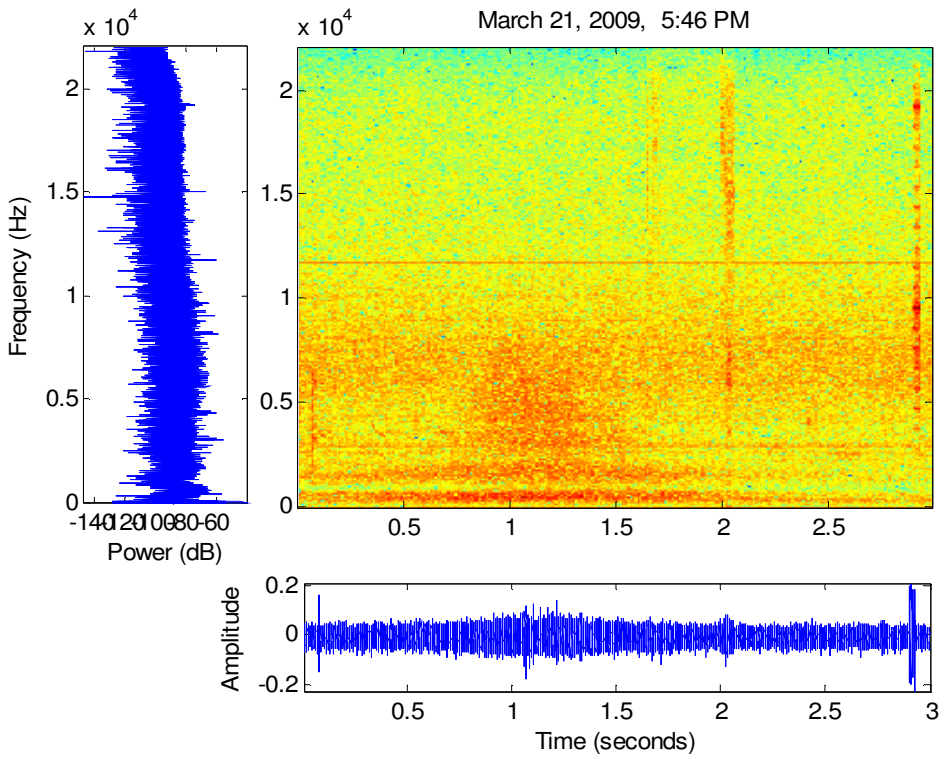

Figure 49: SUV, far lane, 24 ft. from road, 90° with respect to road.

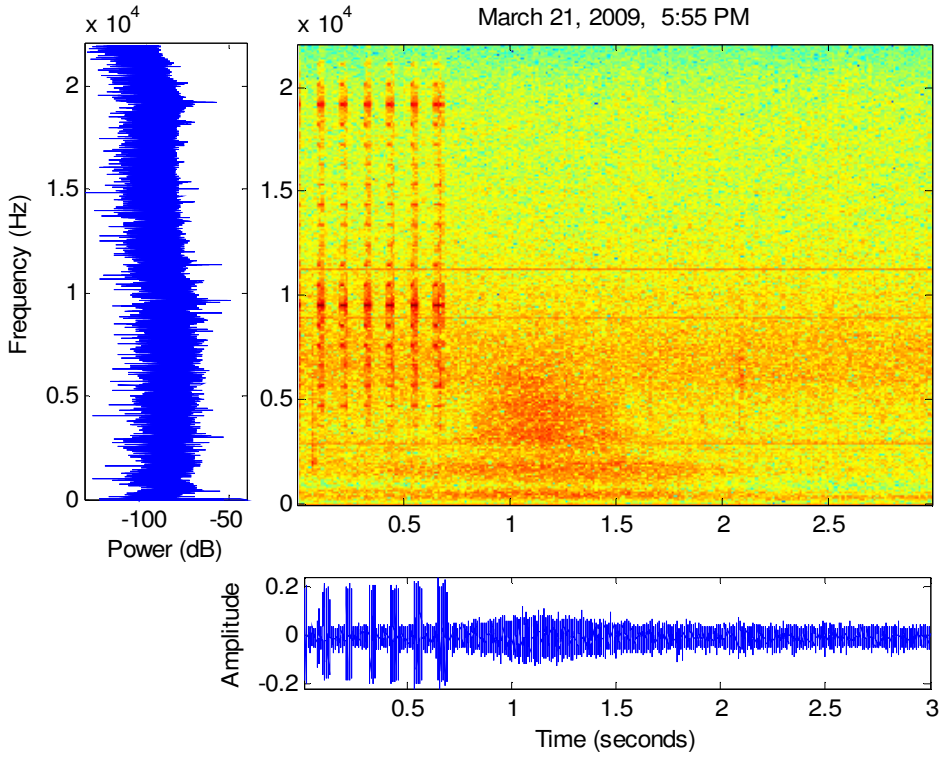

Figure 50: SUV, far lane, 24 ft. from road, 90° with respect to road.

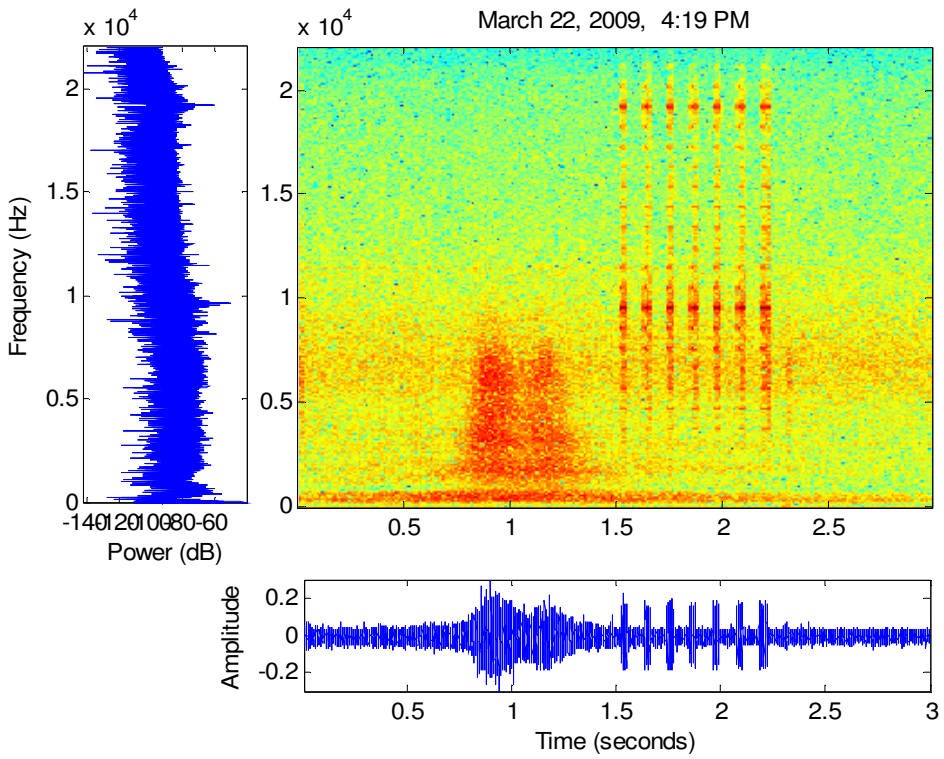

Figure 51: Sedan, far lane, 6 ft. 3 in. from road, 69.41° with respect to road.

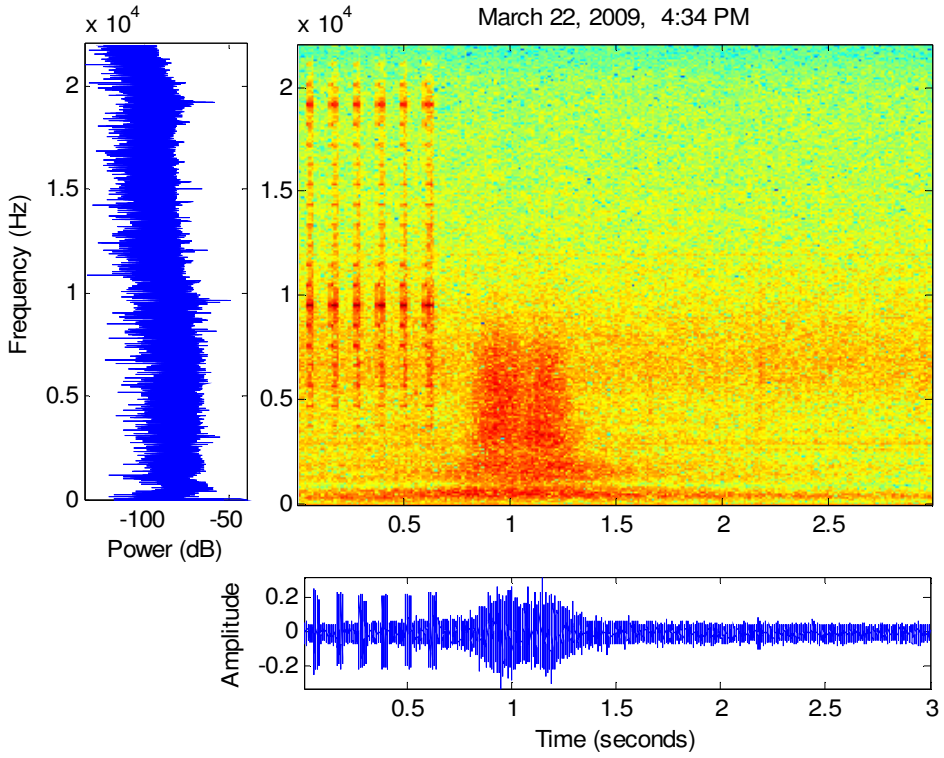

Figure 52: SUV, near lane, 6 ft. 3 in. from road, 69.41° with respect to road.

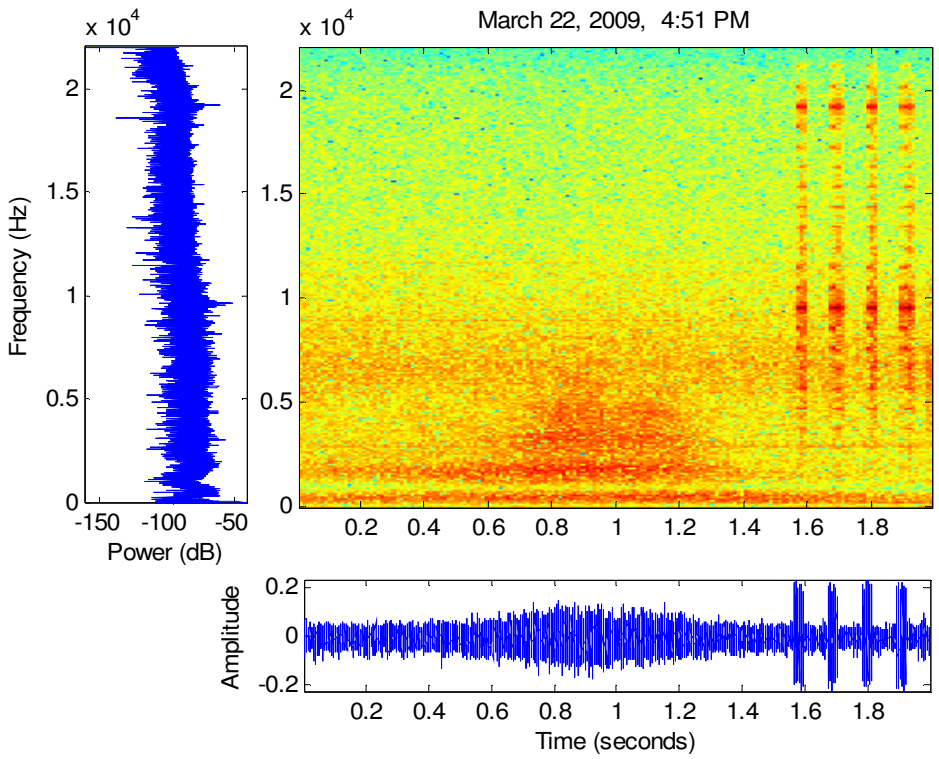

Figure 53: SUV, far lane, 6 ft. 3 in. from road, 69.41° with respect to road.

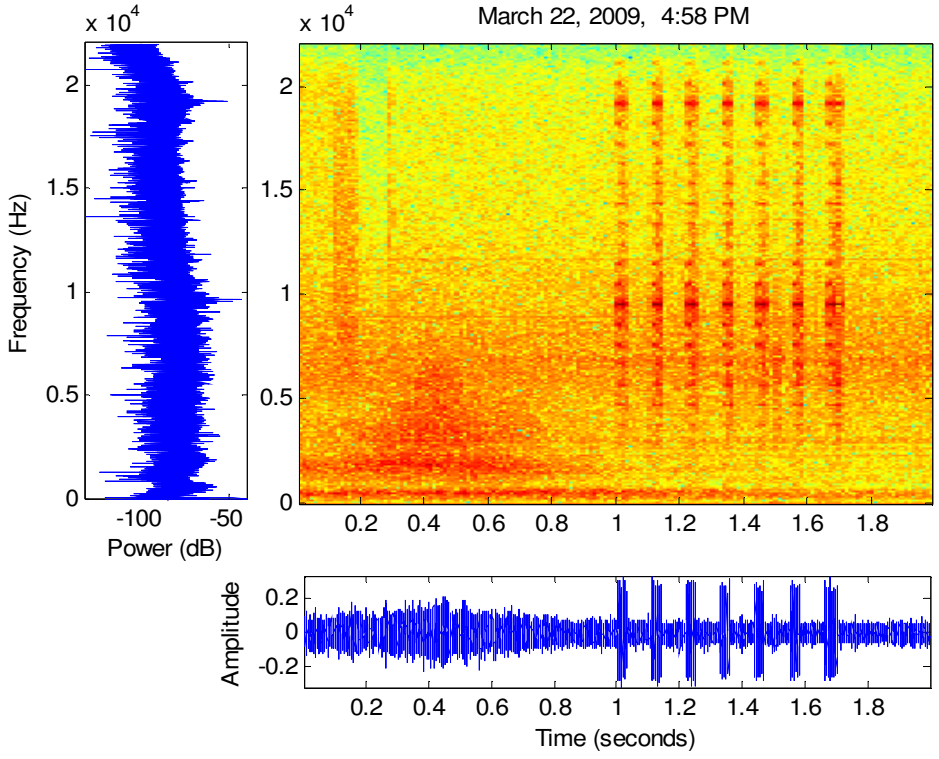

Figure 54: SUV, far lane, 6 ft. 3 in. from road, 69.41° with respect to road.

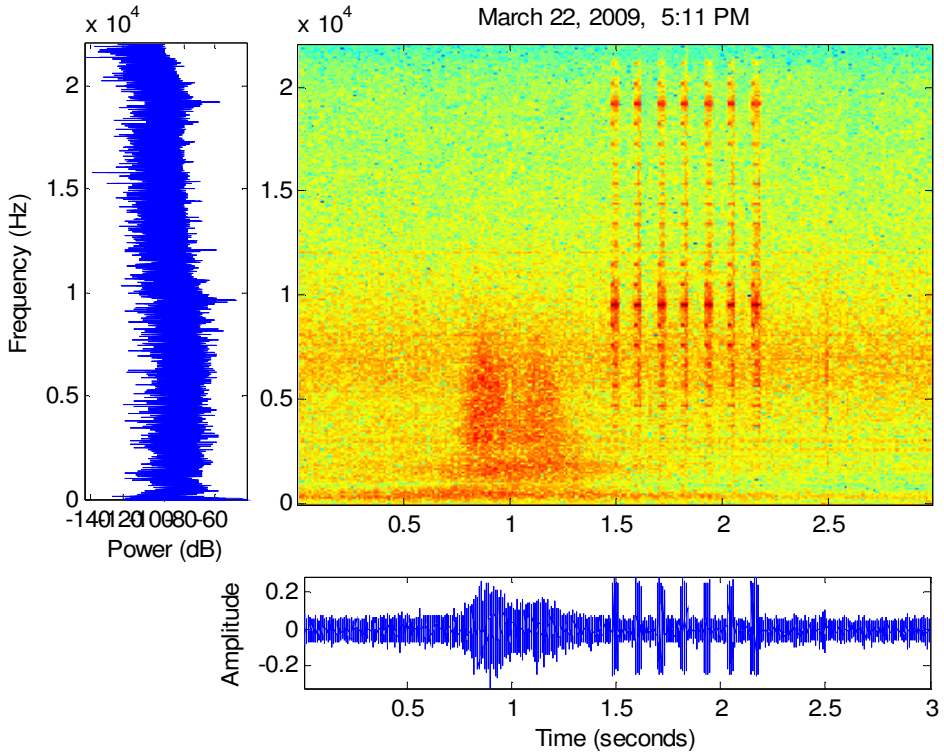

Figure 55: SUV, near lane, 6 ft. 3 in. from road, 69.41° with respect to road.

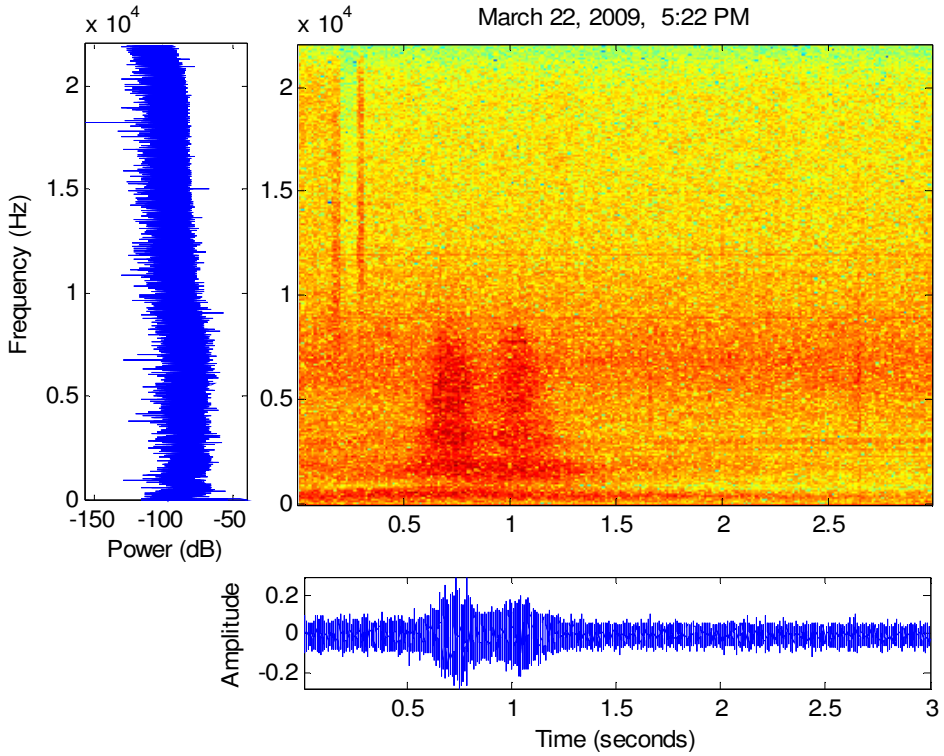

Figure 56: Sedan, near lane, 6 ft. 3 in. from road, 69.41° with respect to road.

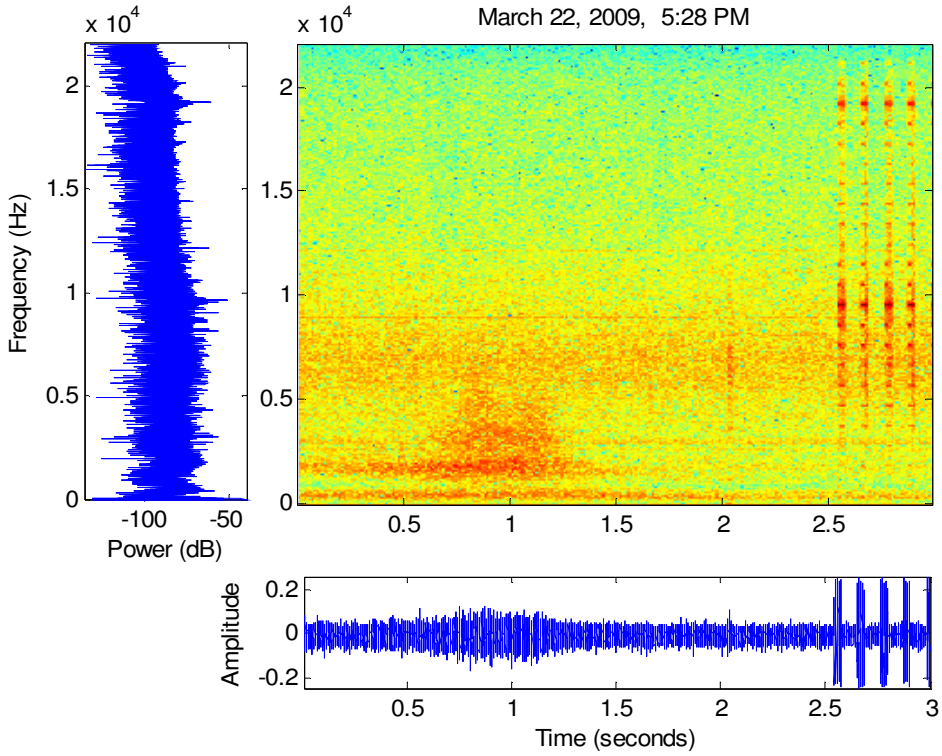

Figure 57: Sedan, far lane, 6 ft. 3 in. from road, 69.41° with respect to road.

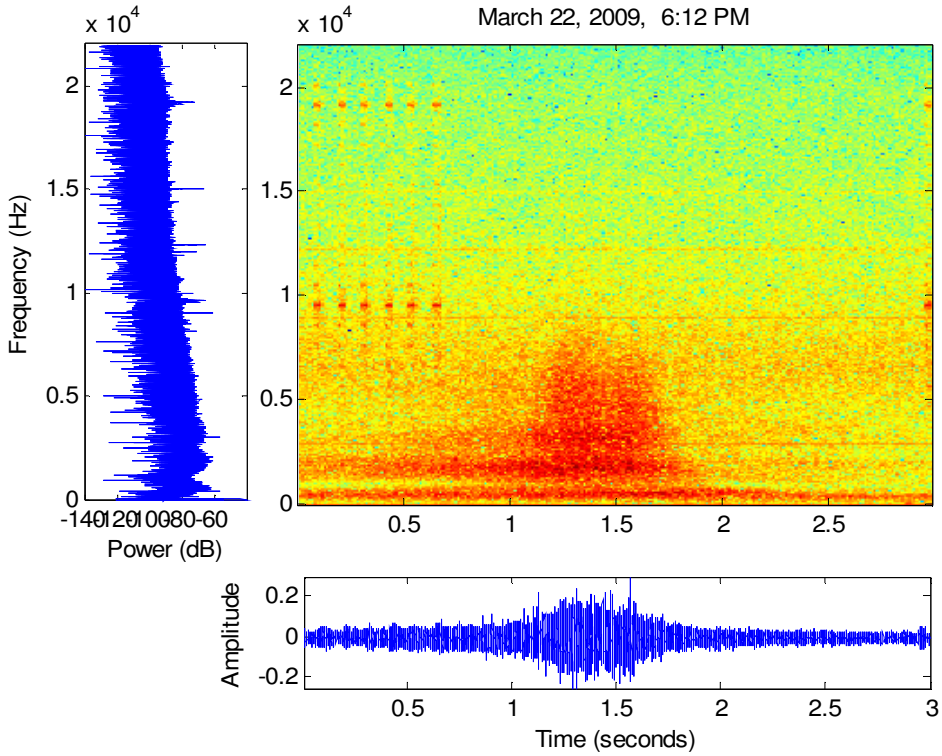

Figure 58: SUV, far lane, 0 ft. from road, 52.90° with respect to road.

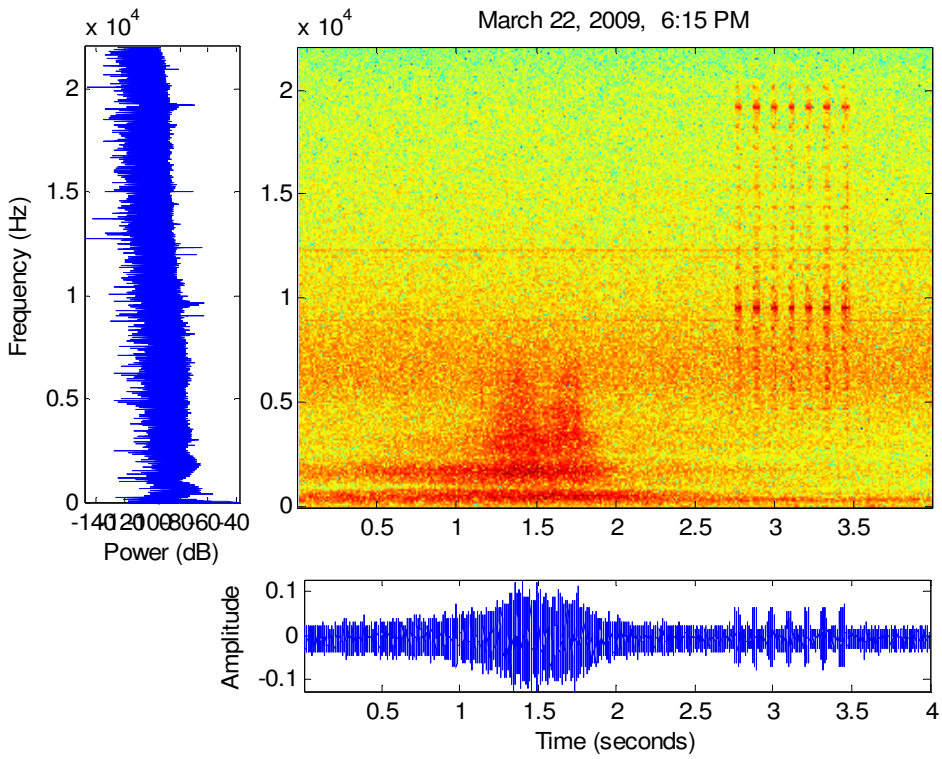

Figure 59: Sedan, far lane, 0 ft. from road, 52.90° with respect to road.

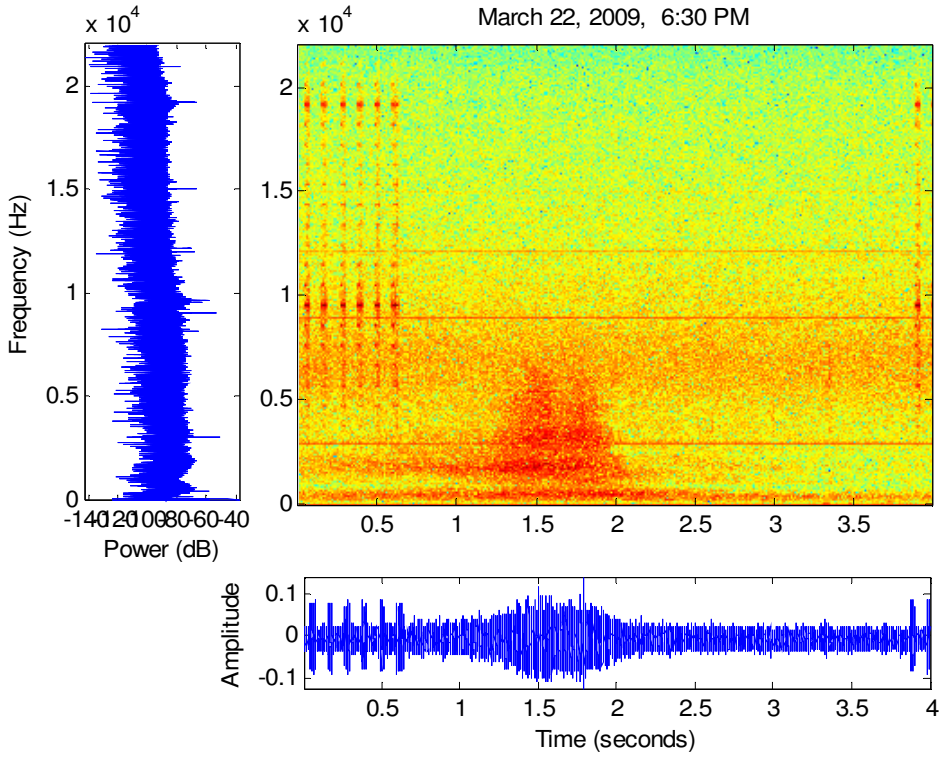

Figure 60: Sedan, far lane, 0 ft. from road, 52.90° with respect to road.

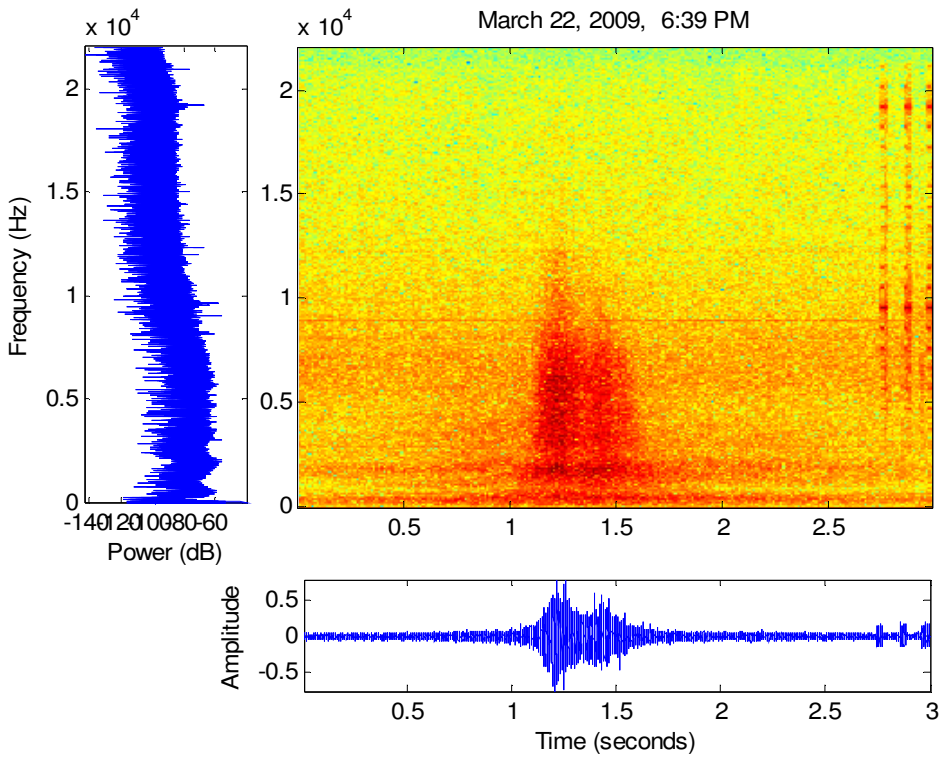

Figure 61: Sedan, near lane, 0 ft. from road, 52.90° with respect to road.

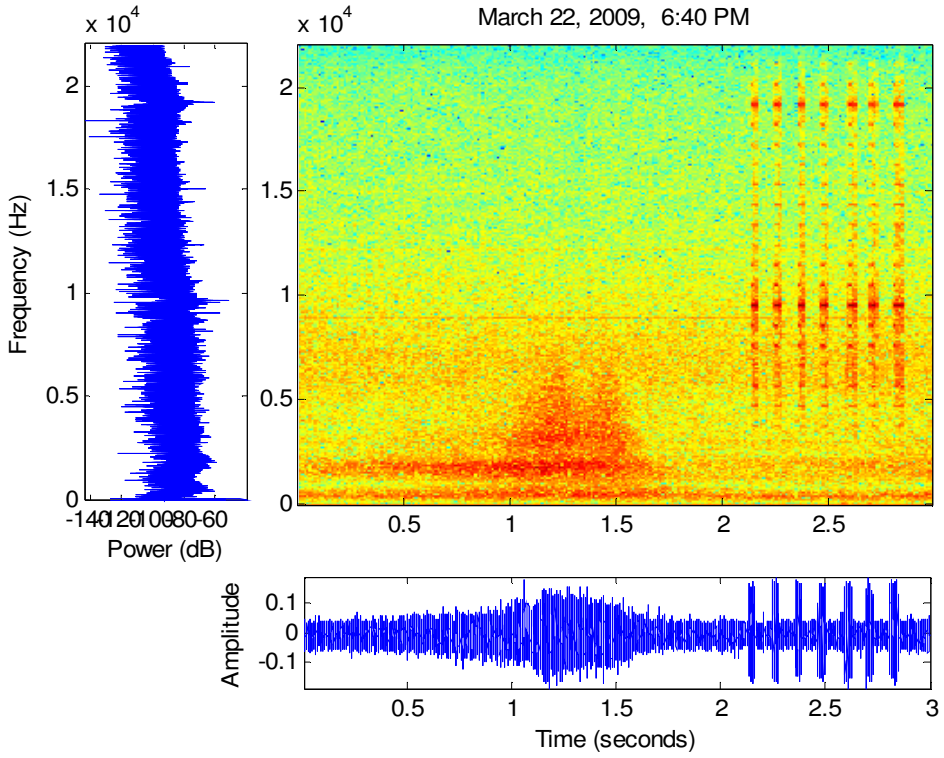

Figure 62: Sedan, far lane, 0 ft. from road, 52.90° with respect to road.

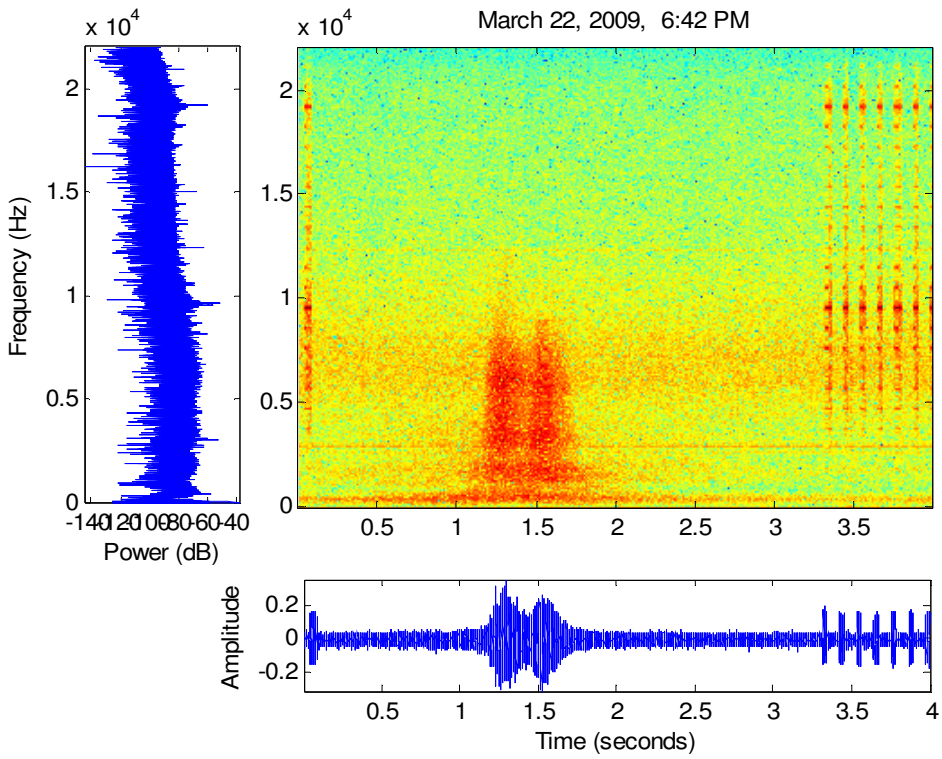

Figure 63: Sedan, near lane, 0 ft. from road, 52.90° with respect to road.

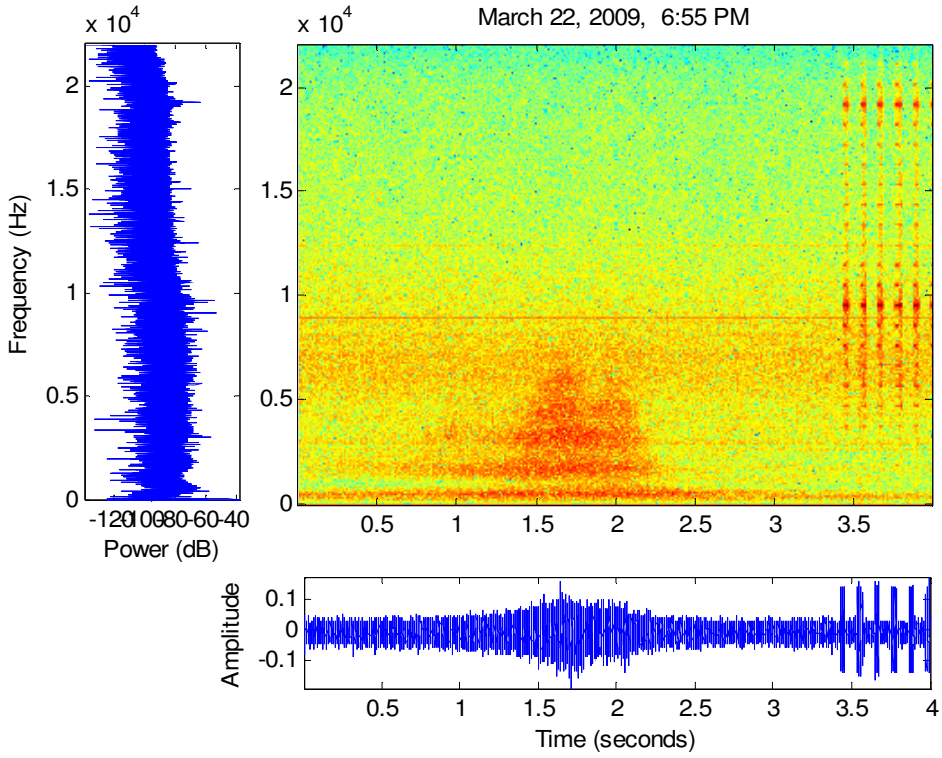

Figure 64: Sedan, far lane, 0 ft. from road, 52.90° with respect to road.

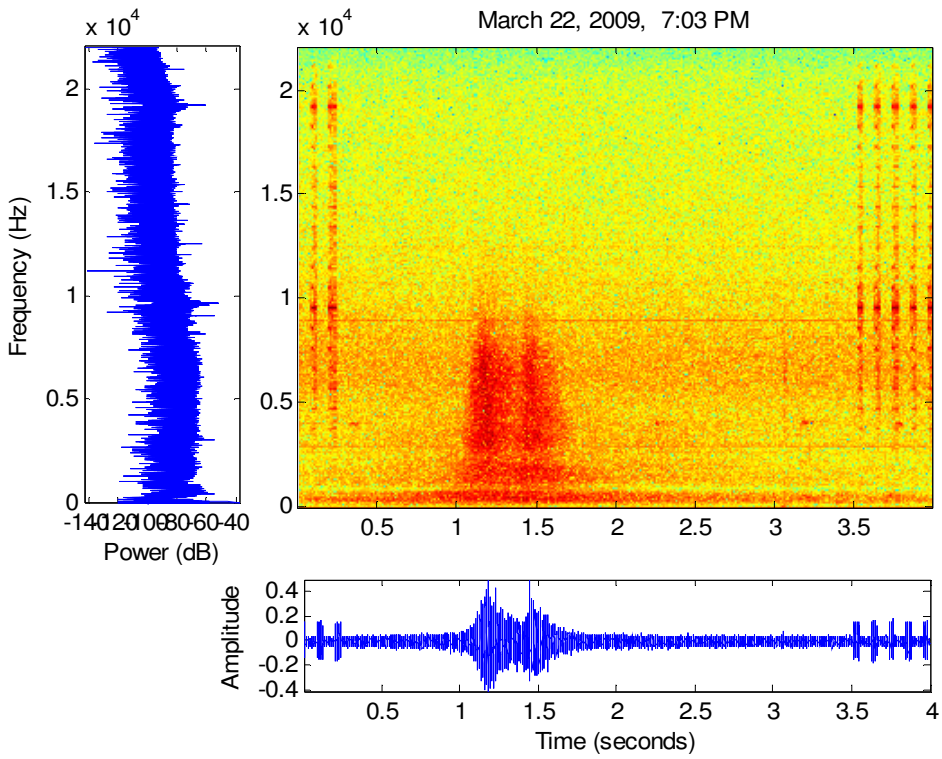

Figure 65: SUV, near lane, 0 ft. from road, 52.90° with respect to road.

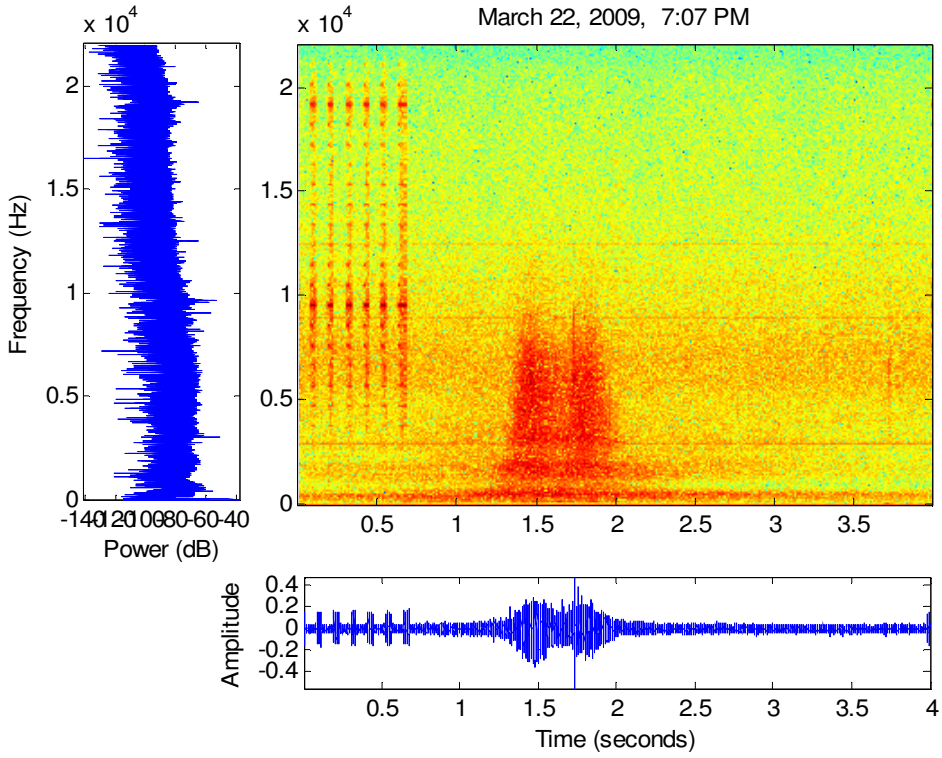

Figure 66: Minivan, near lane, 0 ft. from road, 52.90° with respect to road.

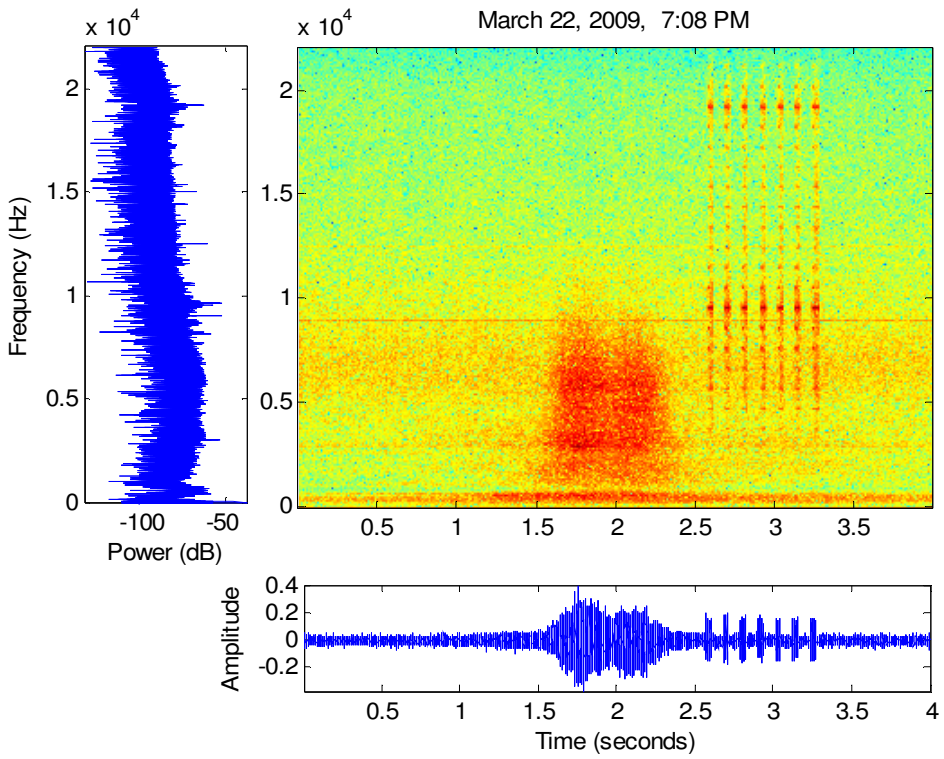

Figure 67: SUV, near lane, 0 ft. from road, 56.75° with respect to road.

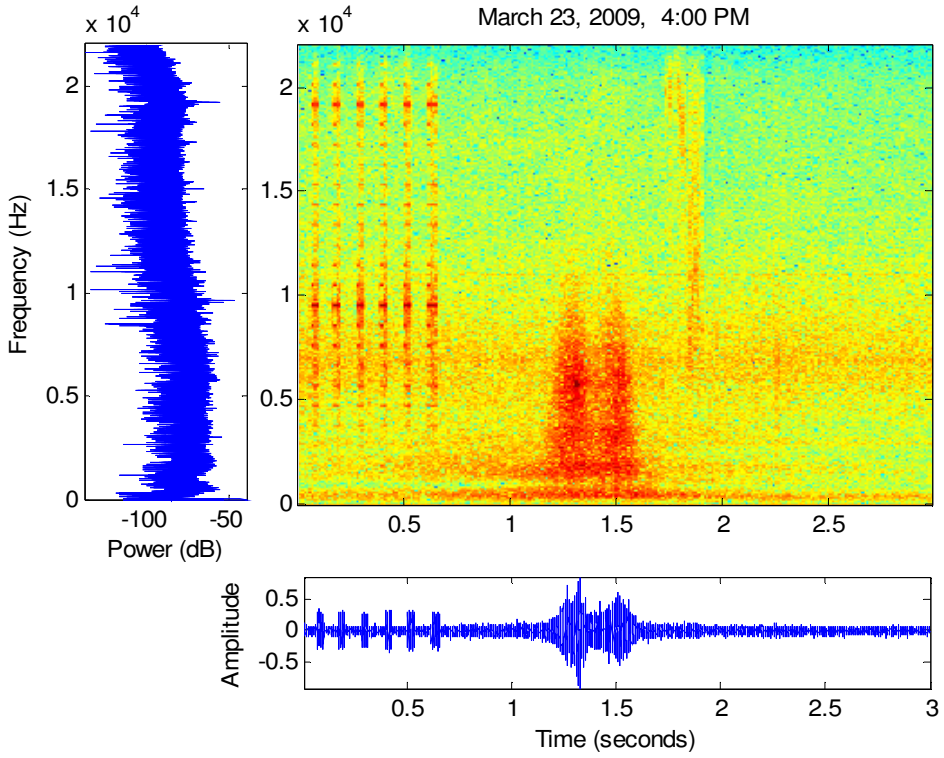

Figure 68: Sedan, near lane, 0 ft. from road, 73.57° with respect to road.

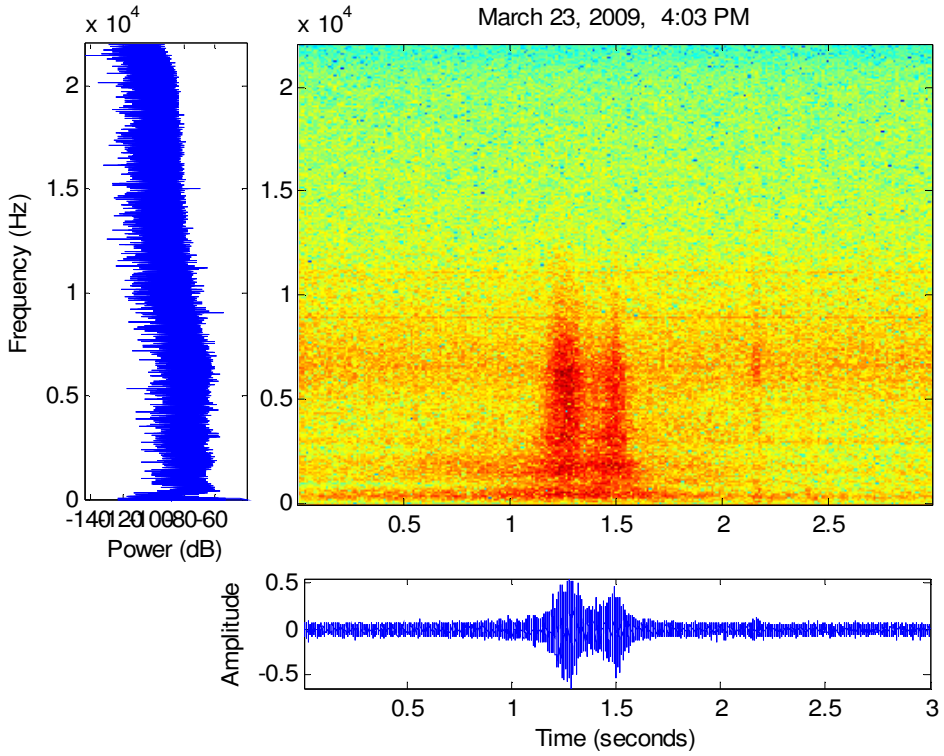

Figure 69: Sedan, near lane, 0 ft. from road, 73.57° with respect to road.

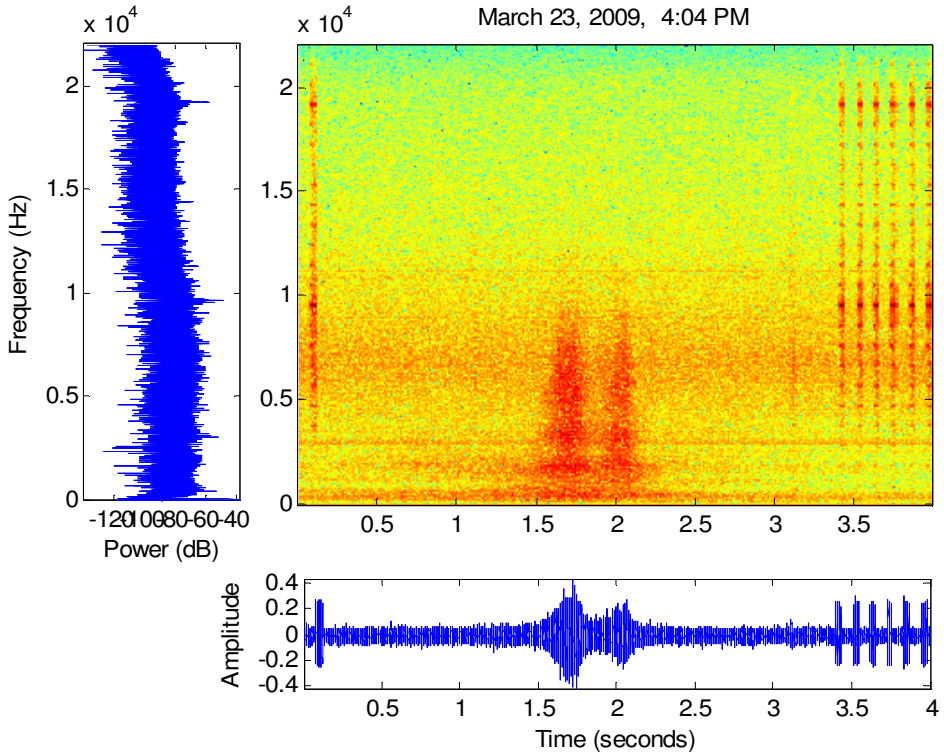

Figure 70: Sedan, near lane, 0 ft. from road, 73.57° with respect to road.

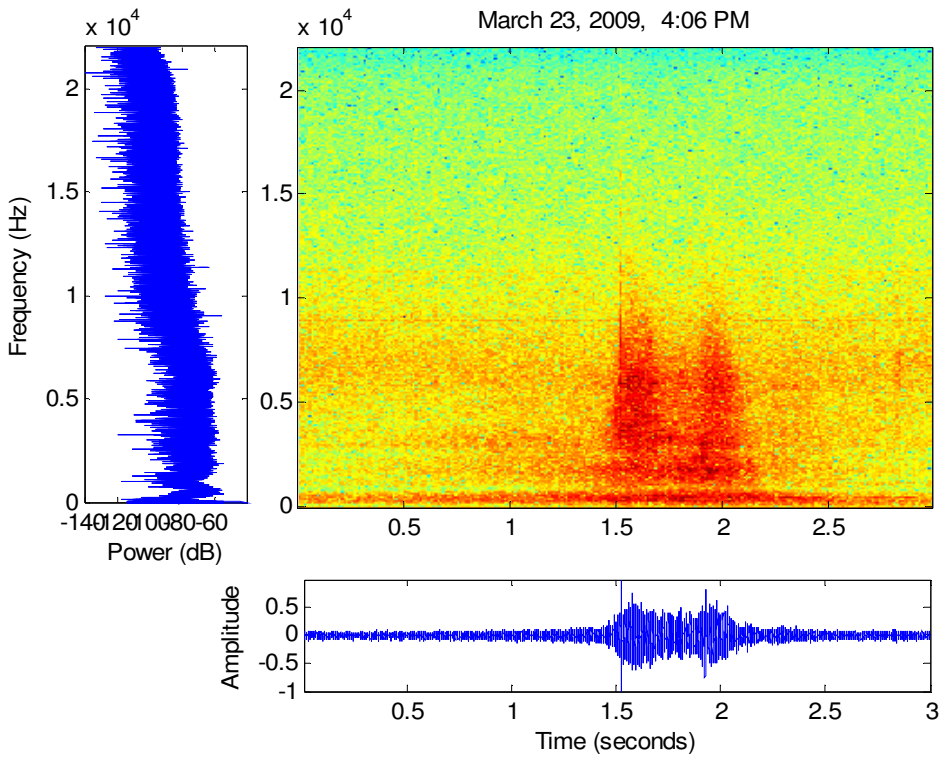

Figure 71: Truck, near lane, 0 ft. from road, 73.57° with respect to road.

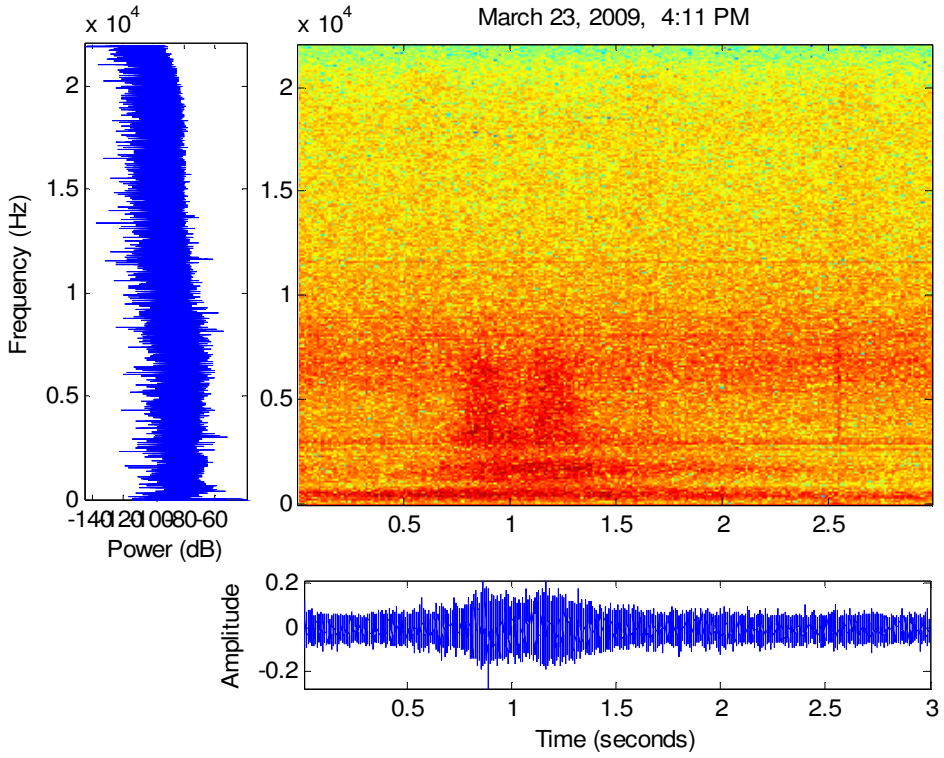

Figure 72: Sedan, far lane, 0 ft. from road, 73.57° with respect to road.

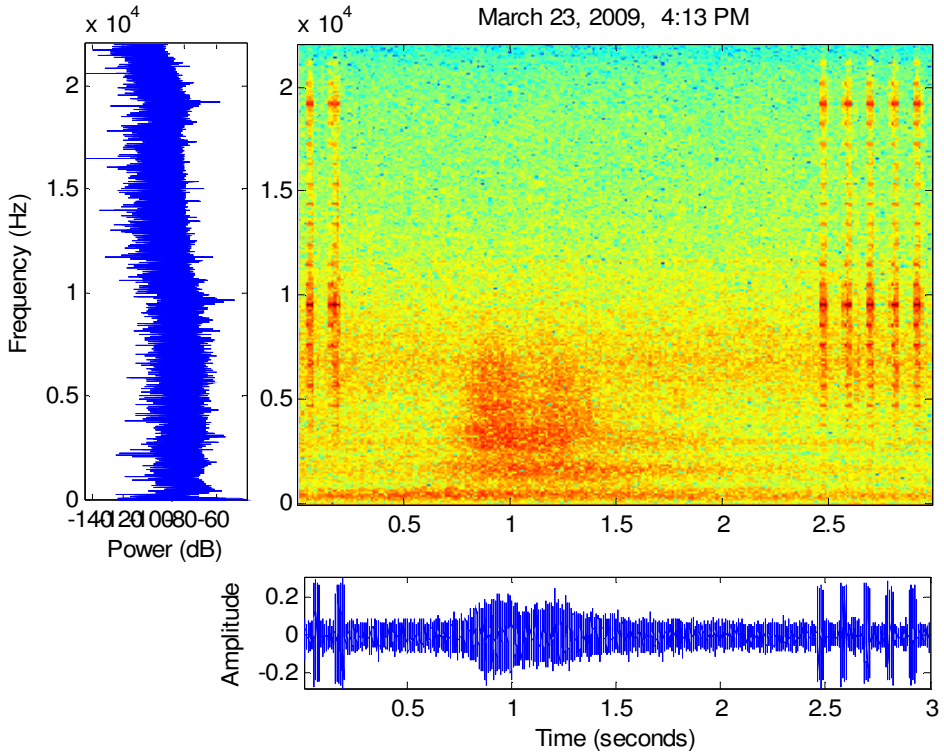

Figure 73: Sedan, far lane, 0 ft. from road, 73.57° with respect to road.

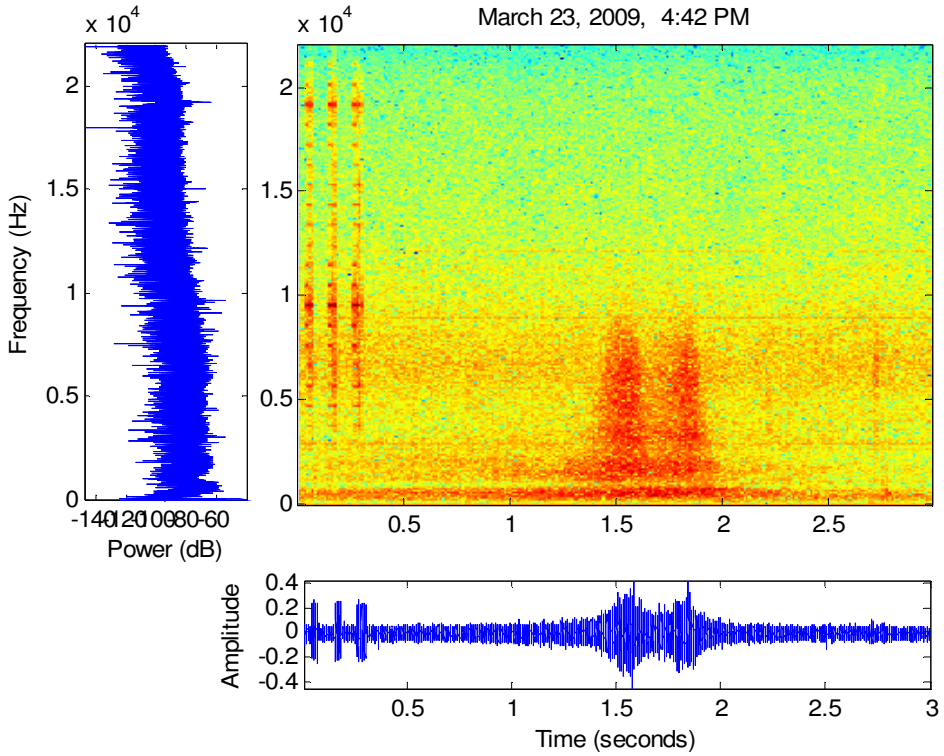

Figure 74: SUV, near lane, 0 ft. from road, 73.57° with respect to road.

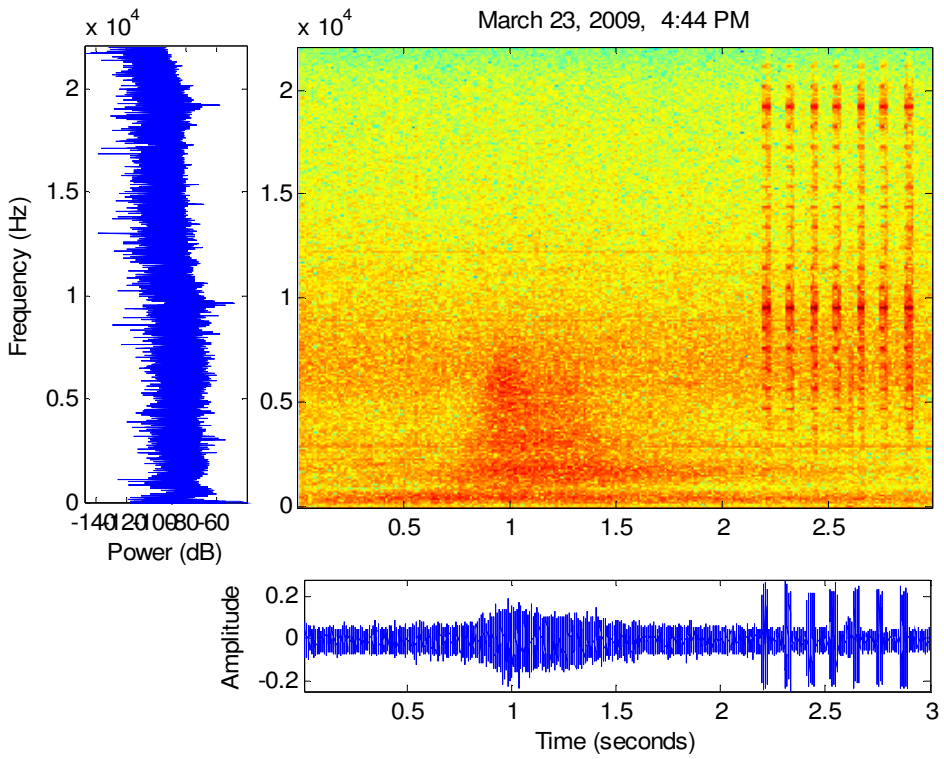

Figure 75: Station, far lane, 0 ft. from road, 73.57° with respect to road.

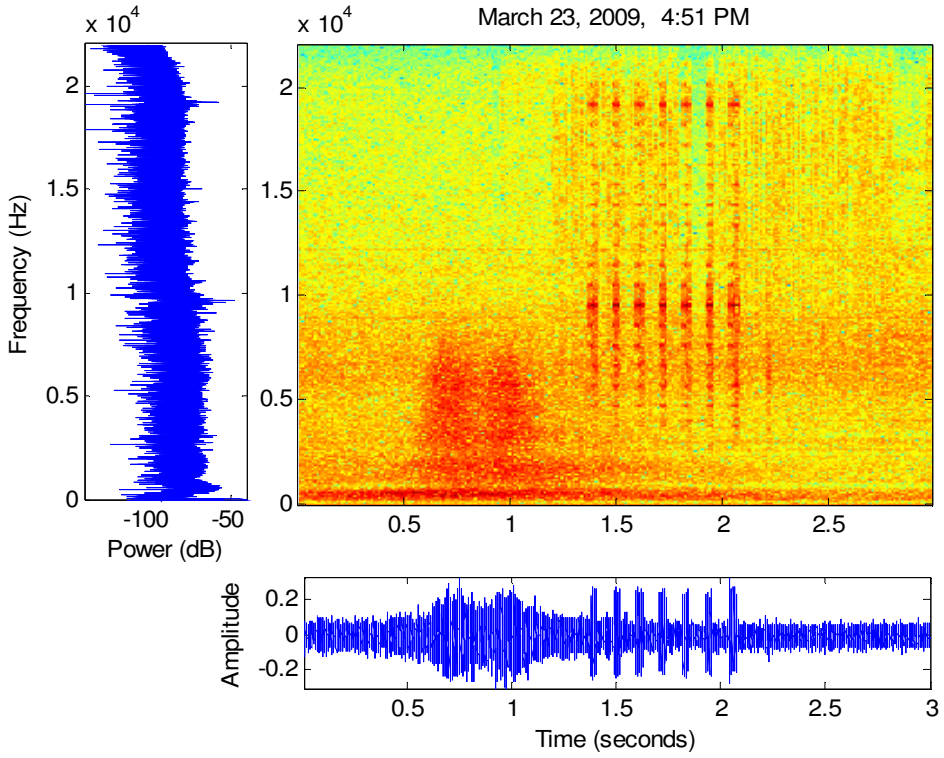

Figure 76: Truck, far lane, 0 ft. from road, 73.57° with respect to road.

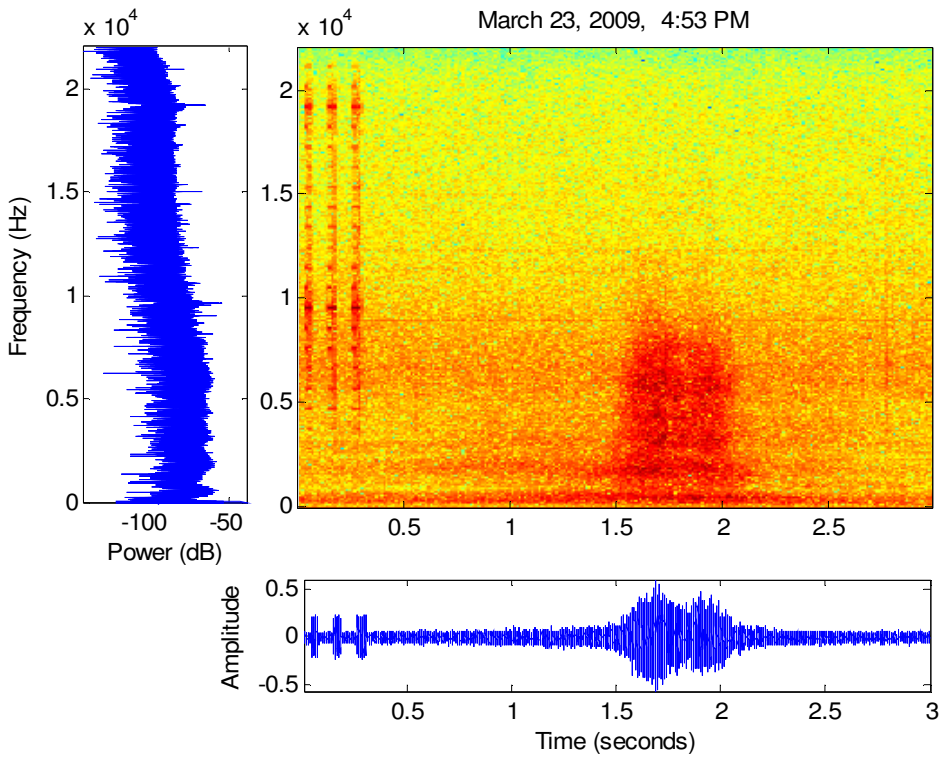

Figure 77: Sedan, near lane, 0 ft. from road, 73.57° with respect to road.

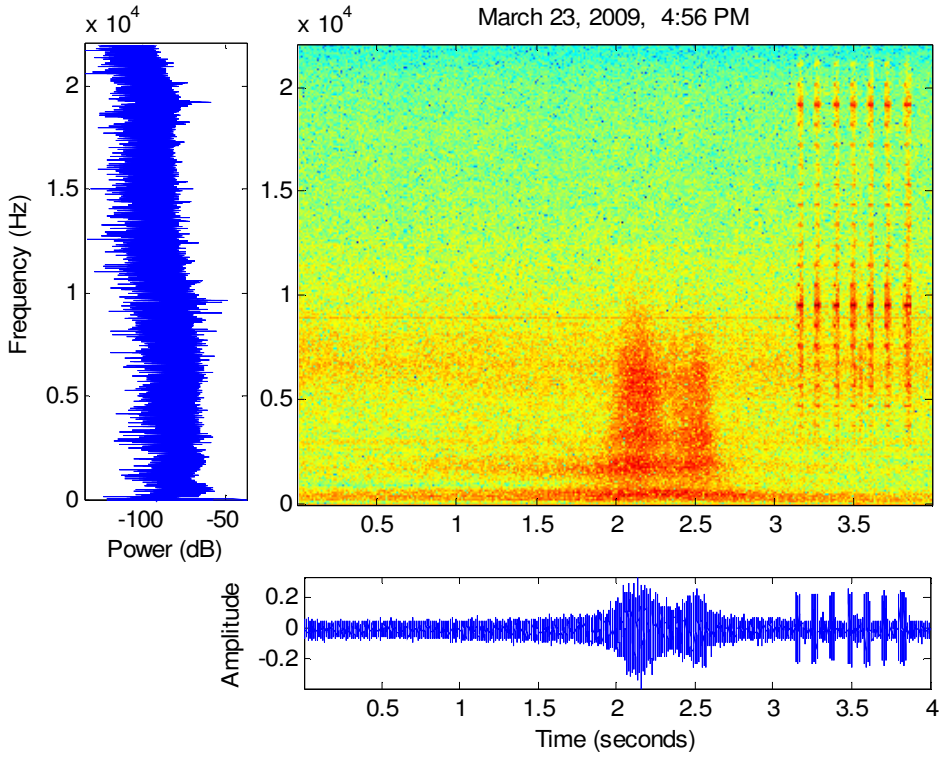

Figure 78: Sedan, near lane, 0 ft. from road, 73.57° with respect to road.

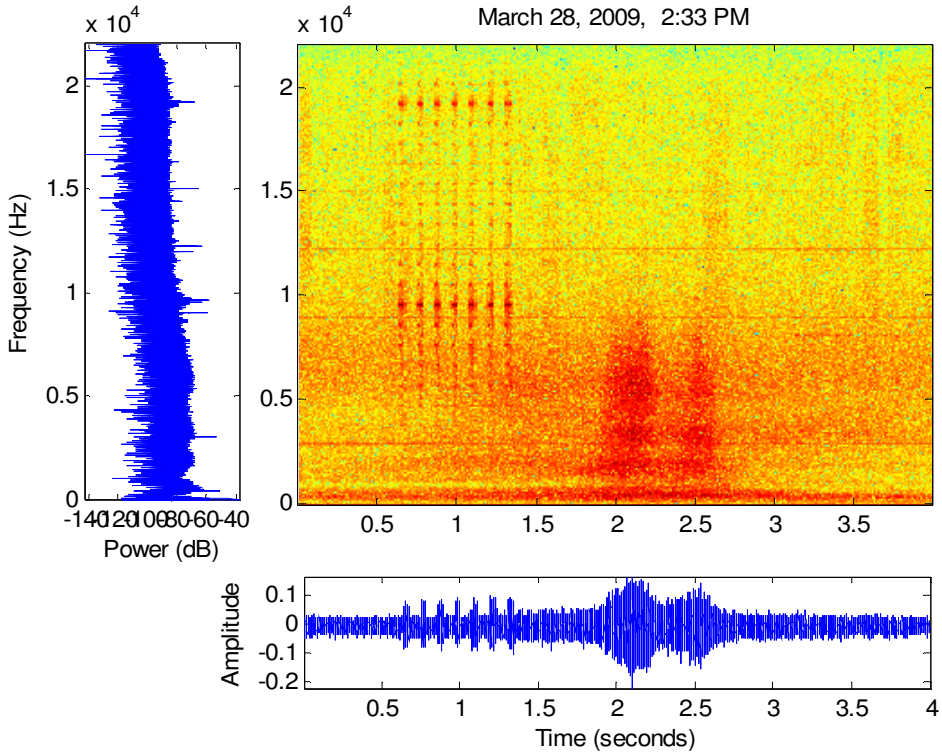

Figure 79: Sedan, near lane, 1 ft. from road, 75.96° with respect to road.

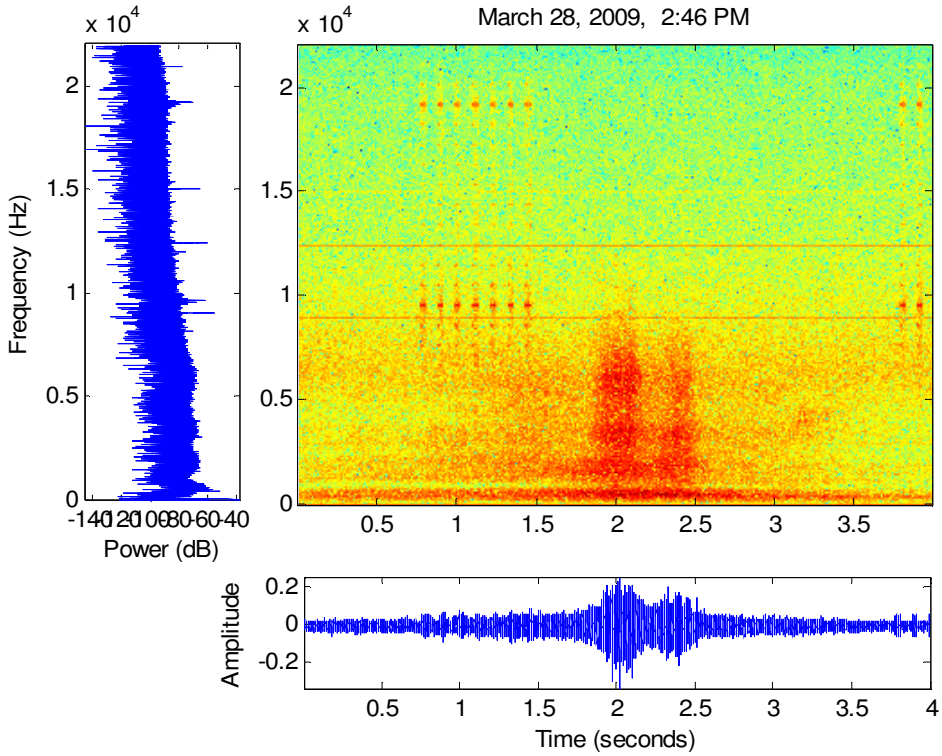

Figure 80: Minivan, near lane, 1 ft. from road, 75.96° with respect to road.

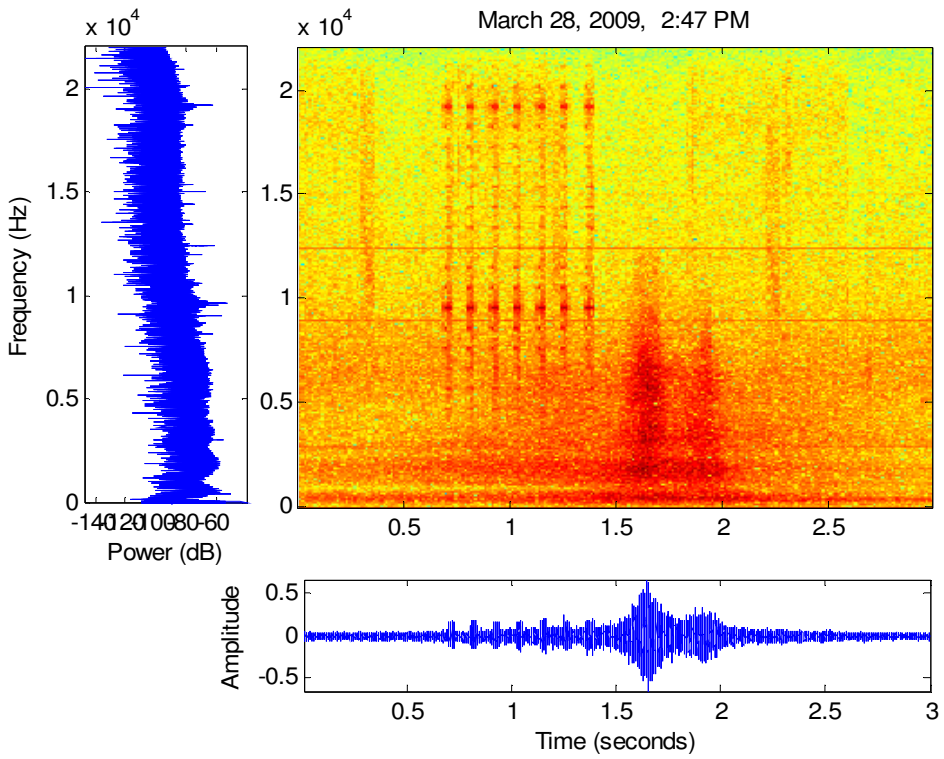

Figure 81: Minivan, near lane, 1 ft. from road, 75.96° with respect to road.

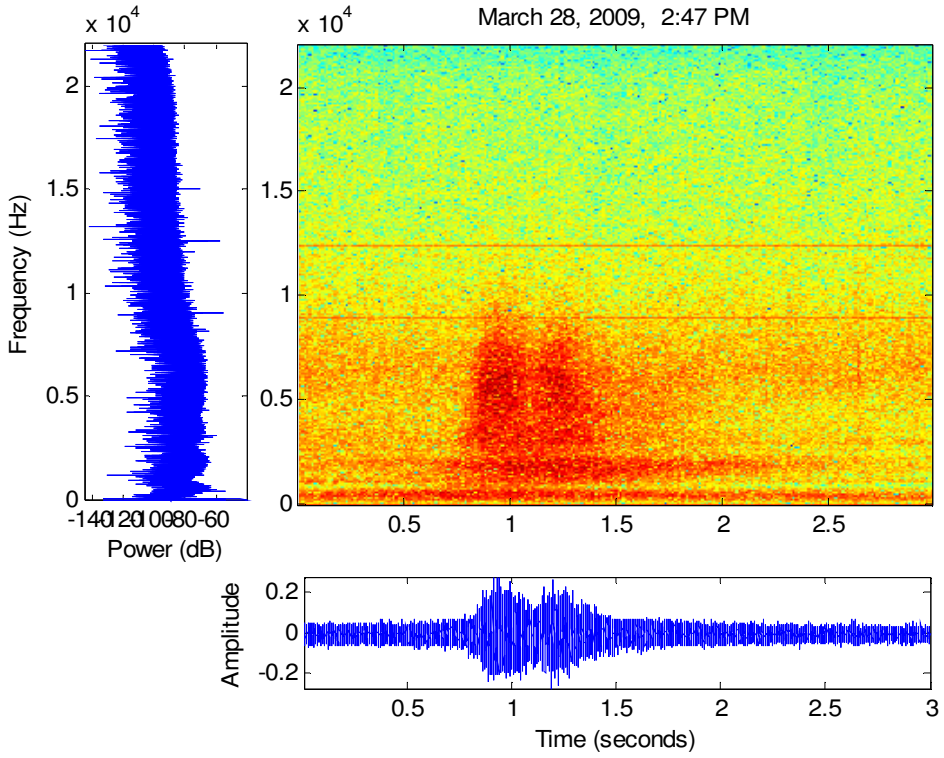

Figure 82: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

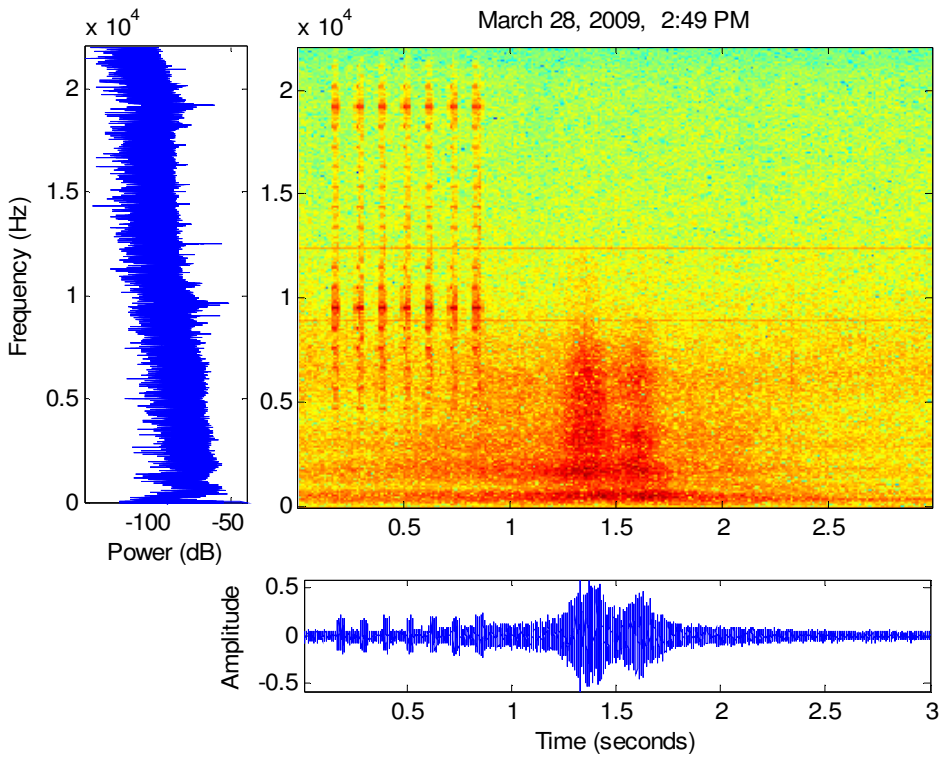

Figure 83: SUV, near lane, 1 ft. from road, 75.96° with respect to road.

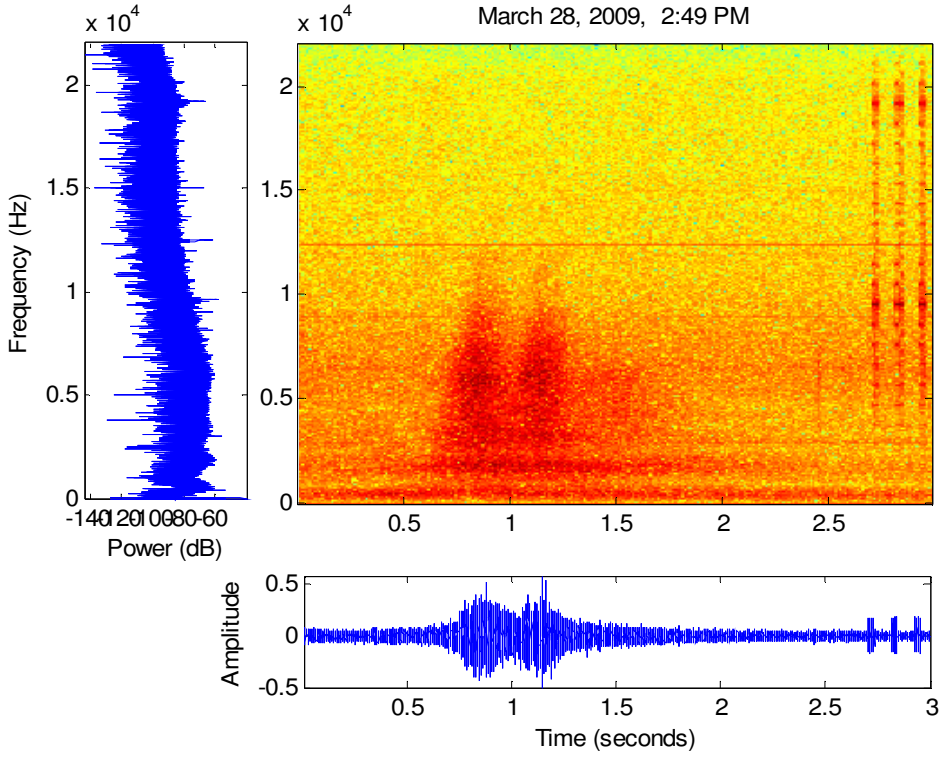

Figure 84: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

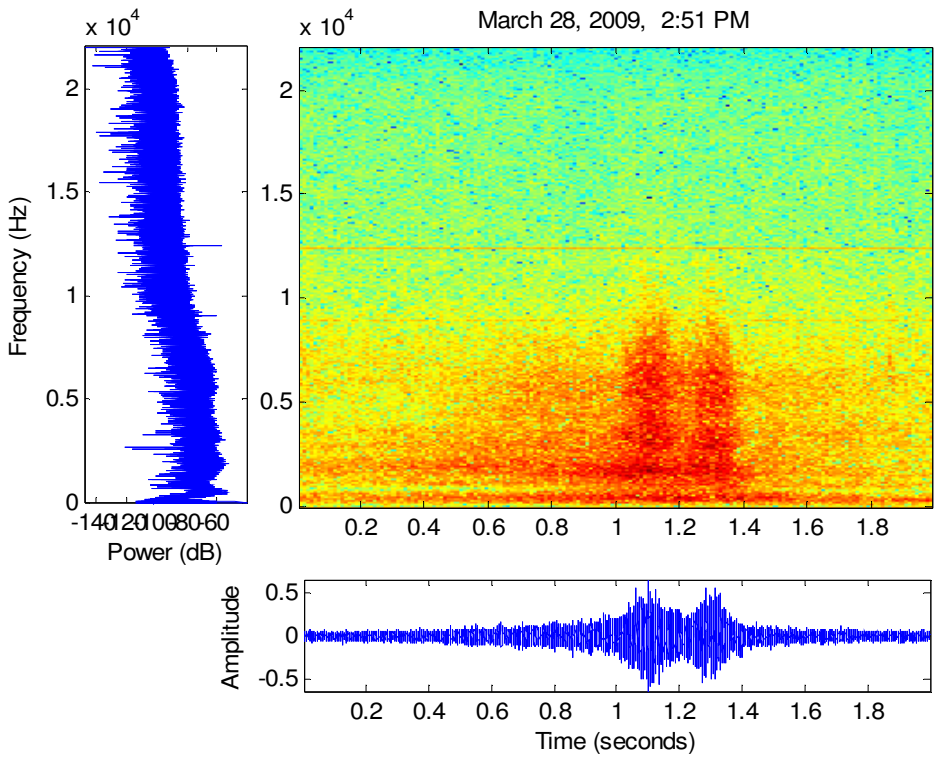

Figure 85: Sedan, near lane, 1 ft. from road, 75.96° with respect to road.

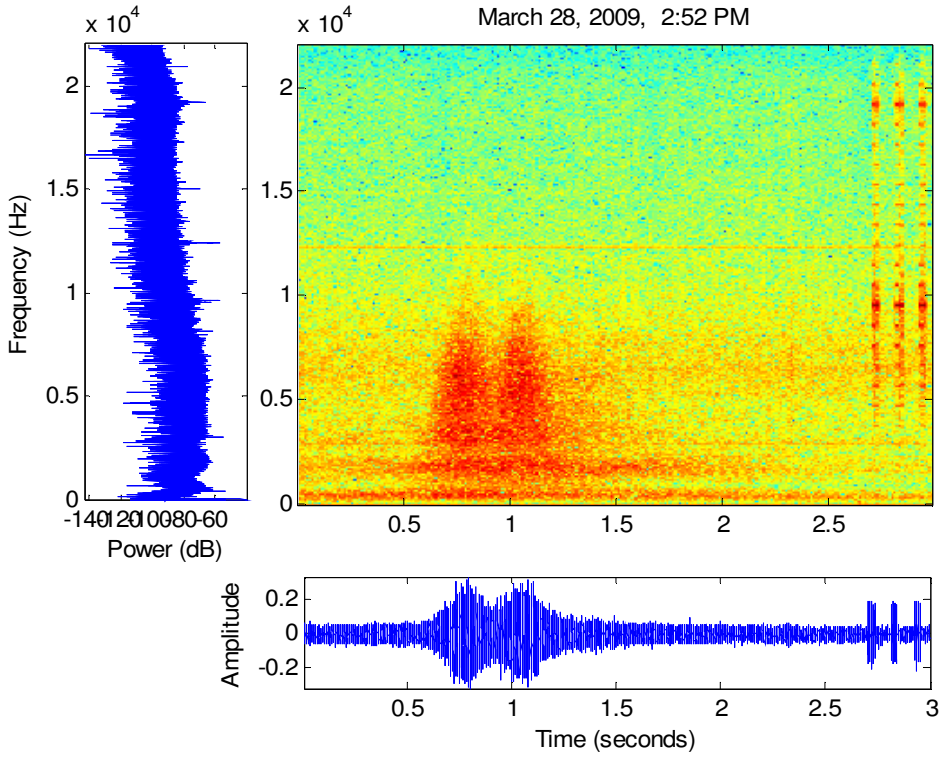

Figure 86: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

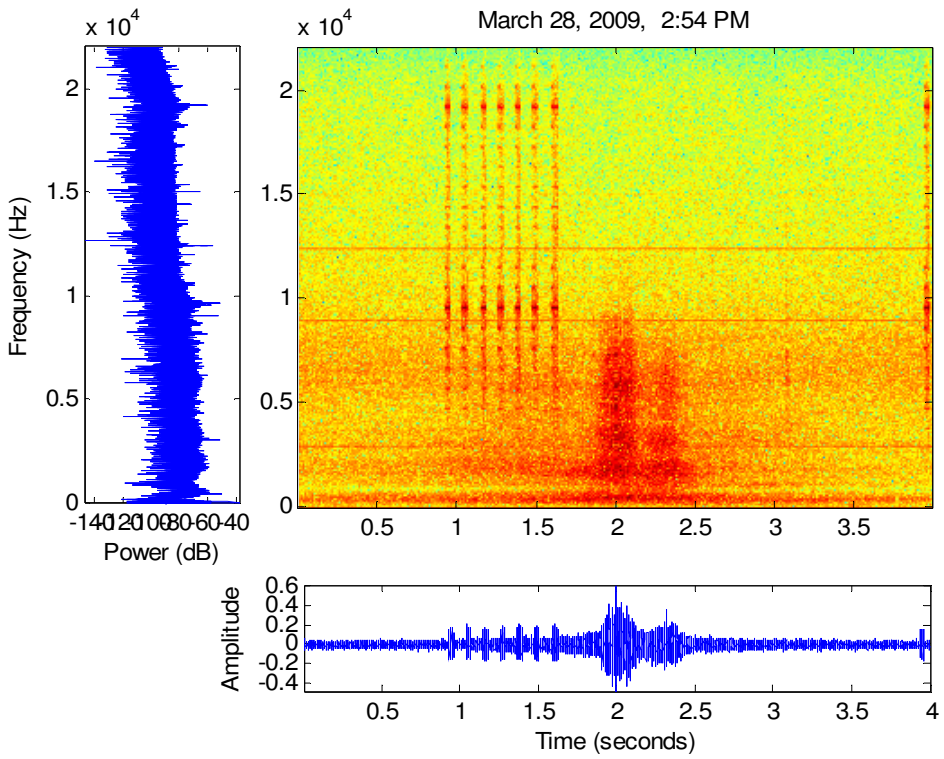

Figure 87: Sedan, near lane, 1 ft. from road, 75.96° with respect to road.

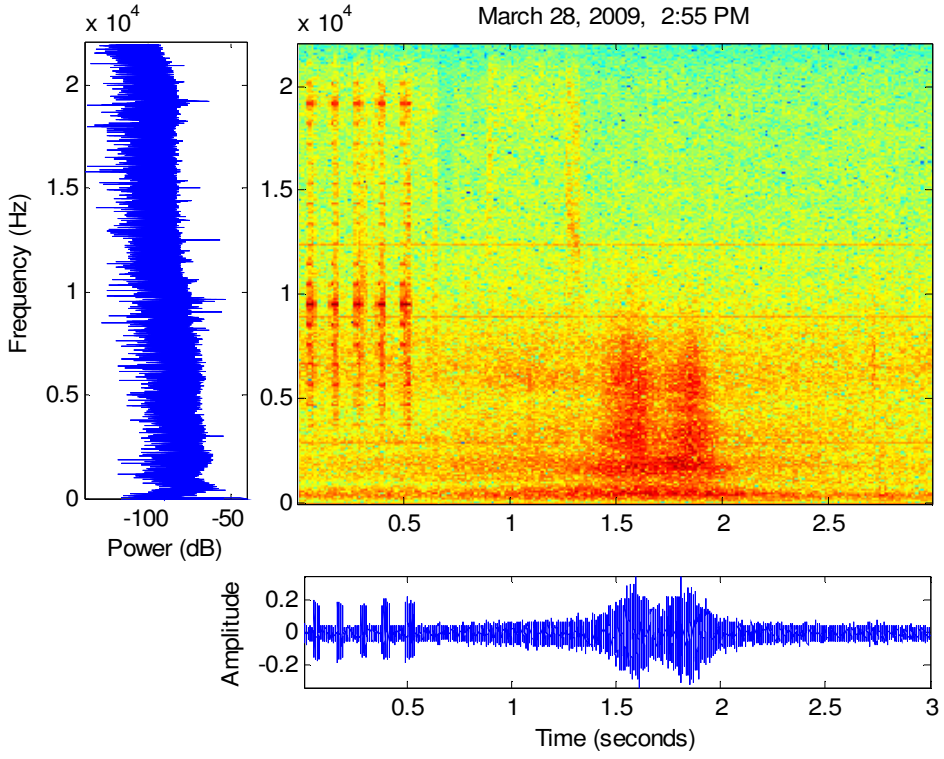

Figure 88: Sedan, near lane, 1 ft. from road, 75.96° with respect to road.

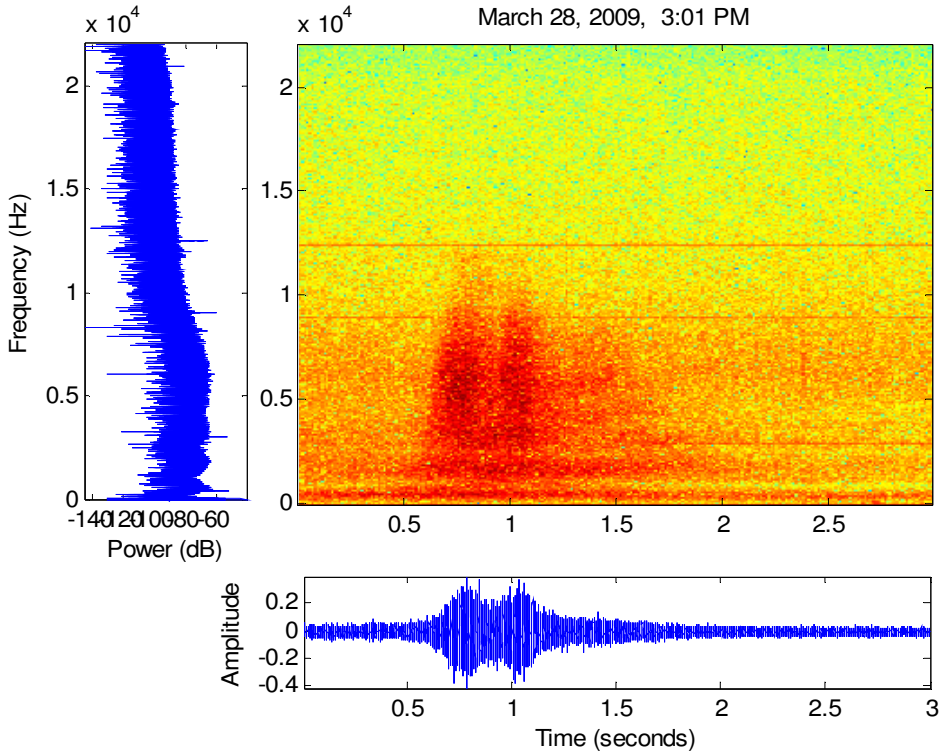

Figure 89: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

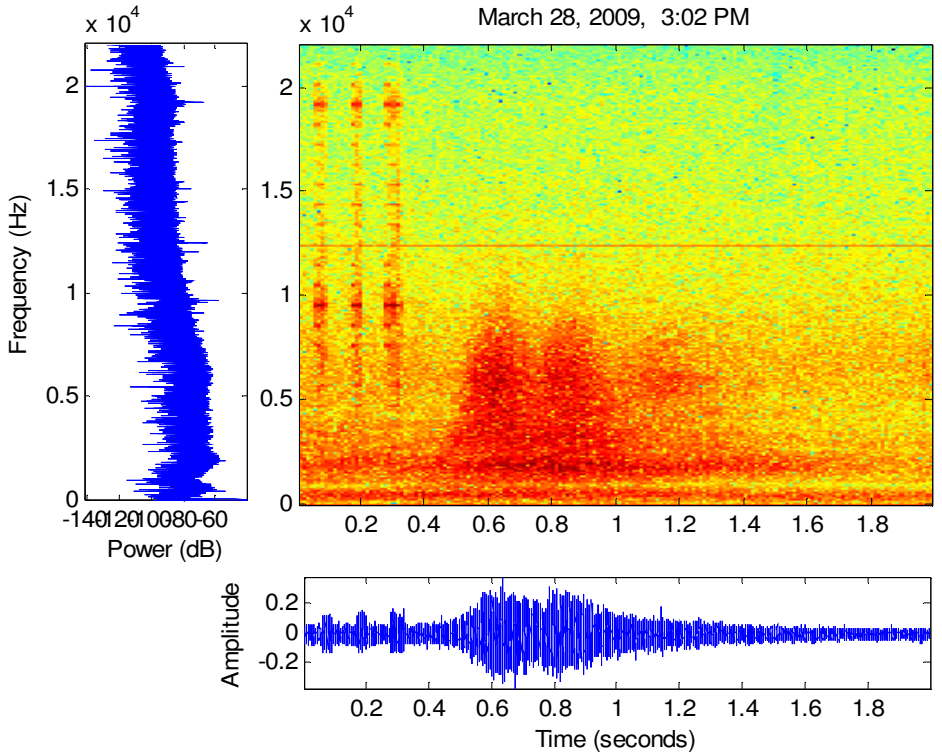

Figure 90: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

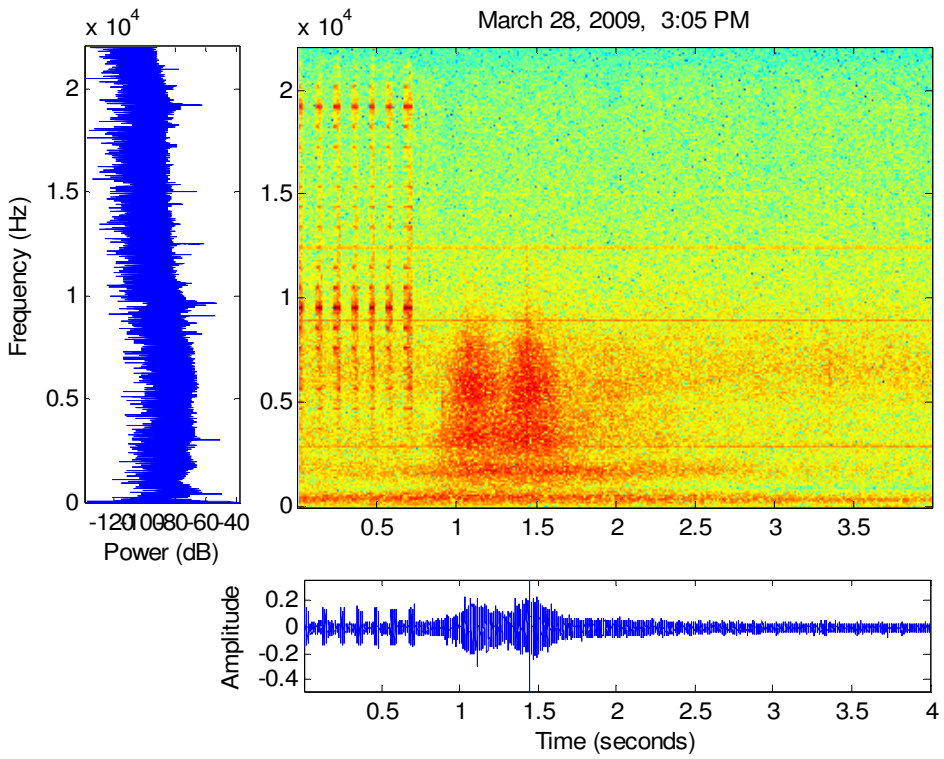

Figure 91: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

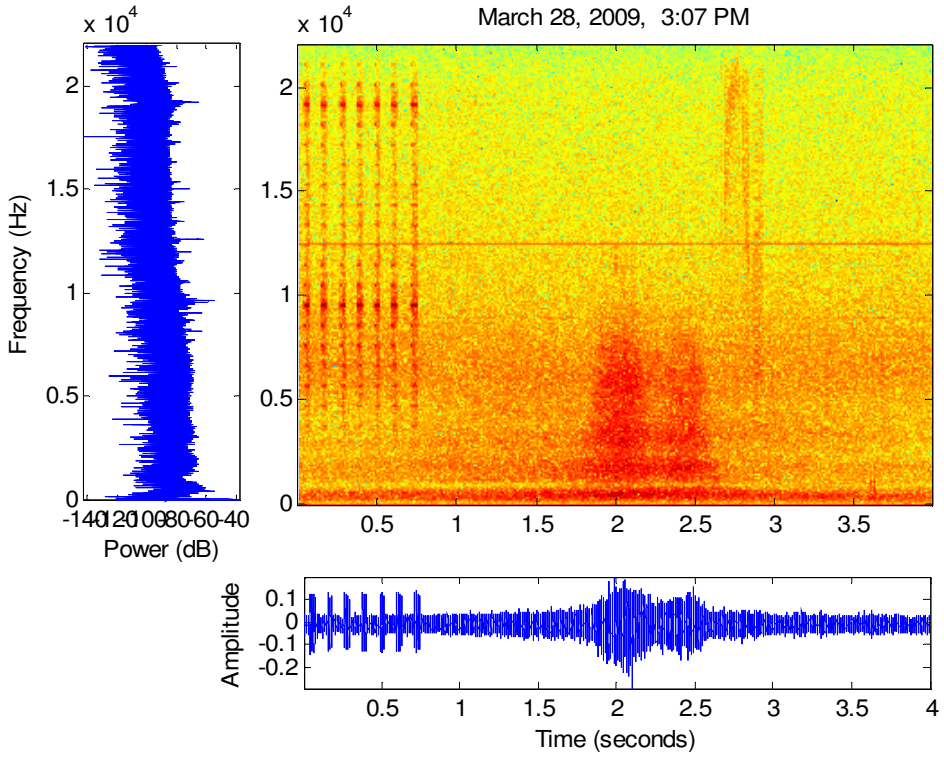

Figure 92: Minivan, near lane, 1 ft. from road, 75.96° with respect to road.

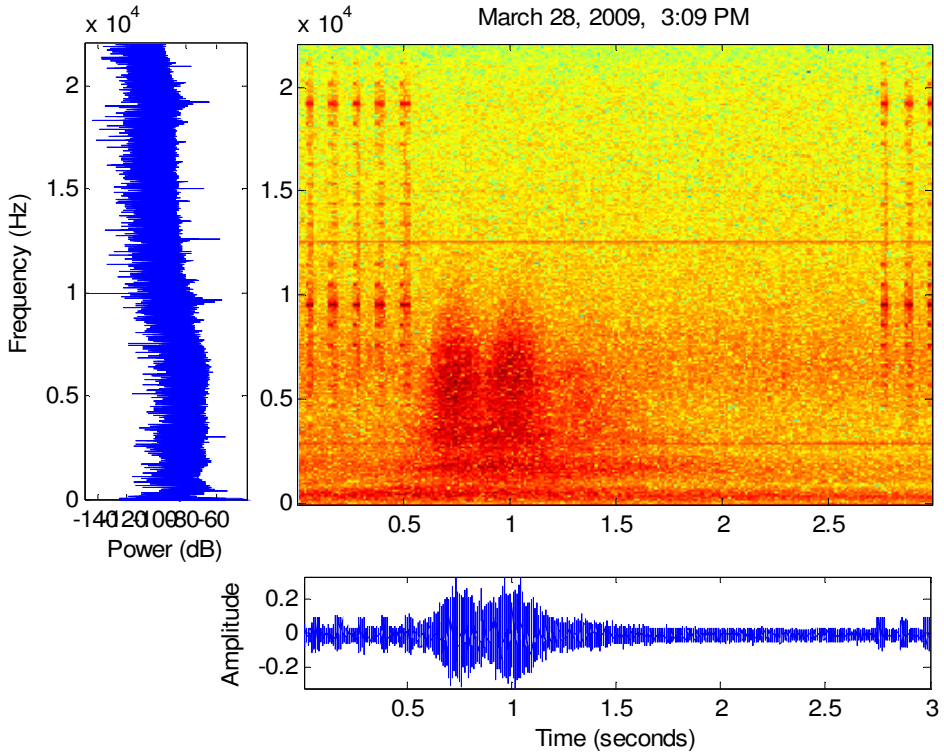

Figure 93: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

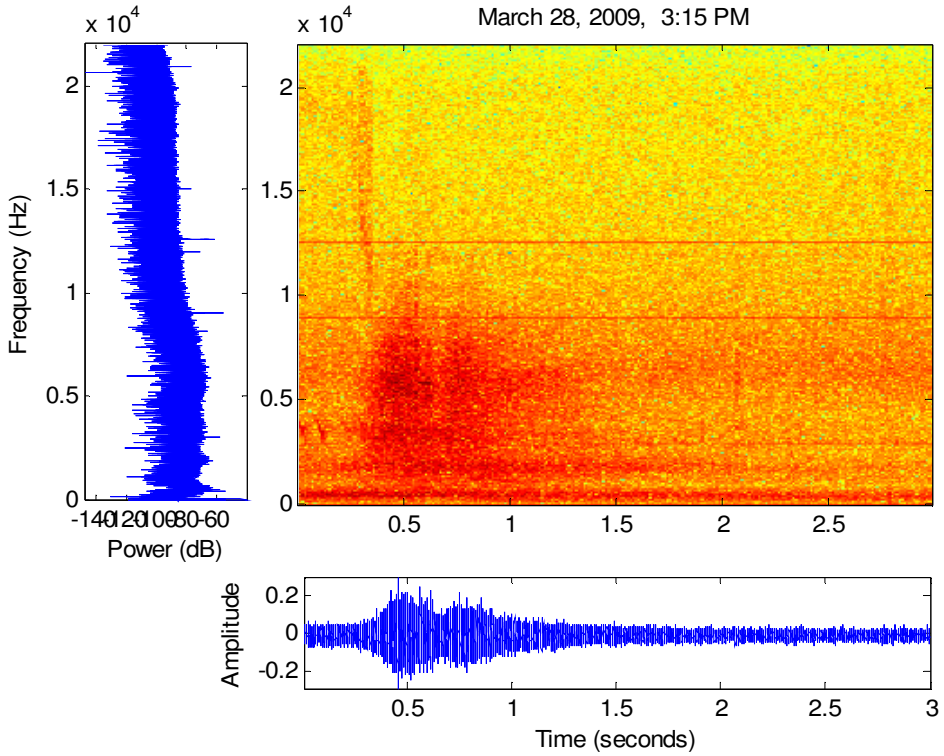

Figure 94: Sedan, far lane, 1 ft. from road, 75.96° with respect to road.

Appendix B

B.1. Recordsig.m File

This file takes in a sinewave input of a given sample rate, plays the sound, and then records the returned data in a matrix. It saves both an audio .wav file, and a MATLAB .mat file, with the current date and time as the filename. It was created for this research project to facilitate the data collection of signal reflections of cars.

```
function recordsig(length,signal)
Fs=11025/4;
len=100;
len=length;
output=440;
output=signal;
soundsig=ToneBurst(output,len,Fs);
audio=audiorecorder(44100,8,1);
input('Press Enter key to start recording.');
wavplay(soundsig,Fs,'async');
record(audio);
input('Press Enter key to stop recording.');
stop(audio);
curtime=fix(clock);
name=strcat(num2str(curtime(1)),'-',num2str(curtime(2)),'-
',num2str(curtime(3)),'_',num2str(curtime(4)),'-
',num2str(curtime(5)),'-',num2str(curtime(6)));
audiograph=getaudiodata(audio);
wavwrite(audiograph,44100,8,strcat('data\',name));
save(strcat('data\',name),'audiograph');
fprintf(strcat('Filename:',name));
```

B.2. ToneBurst.m File

This file creates a sinewave that can be used in the Recorder.m program to play an audio signal. It was previously created for use in Rudd's work [9].

```
function [ tb ] = ToneBurst( f, num, samplerate )
% TONEBURST
% returns a <num> cycle toneburst of frequency <f>
% at samplerate <samplerate>
tb = sin((0:(2*pi/samplerate):(2*pi)*num/f)*f);
```

References

1. P. J. Westervelt. “Parametric Acoustic Array”, *The Journal of the Acoustical Society of America.* **35** (4), 535 (1963).
2. B. Felhman. “Classification of Non-Heat Generating Outdoor Objects in Thermal Scenes for Autonomous Robots”, in *Dept. of Applied Science.* 2008, College of William and Mary: Williamsburg, VA.
3. W. Gao and M. Hinders. “Mobile Robot Sonar Backscatter Algorithm for Automatically Distinguishing Walls, Fences, and Hedges”, *The International Journal of Robotics Research*. **25** (2), 135-45 (2006).
4. M. Hinders, W. Gao, and W. Fehlman. “Sonar Sensor Interpretation and Infrared Image Fusion for Mobile Robotics”, *Advanced Robotic Sysems International*
5. M. Hinders and W. Gao. “Mobile Robot Sonar Interpretation Algorithm for Distinguishing Tress from Poles”, *Review of Quantitative Nondestructive Evaluation*. **24** 752 (2005).
6. M. B. Bennett and D. T. Blackstock. “Parametric Array in Air”, *The Journal of the Acoustical Society of America*. **57** (3) 562 (1975).
7. Sennheiser Electronic Corporation. “Sennheiser USA – AudioBeam – Product Details” <http://www.sennheiserusa.com/newsite/productdetail.asp?transid=009859>, Accessed April, 2009.
8. Associated Press. "Troops get high-tech noisemaker." March 3, 2004. <http://www.cnn.com/2004/TECH/ptech/03/03/sonic.weapon.ap/index.html>.

9. K. Rudd, “Parallel 3D Acoustic and Elastic Wave Simulation Methods with Applications in Nondestructive Evaluation”, in *Dept. of Applied Science.* 2007, College of William and Mary: Williamsburg, VA.