Computers, Materials & Continua
DOI:10.32604/cmc.2022.019904
Article

Tech Science Press

Hybrid In-Vehicle Background Noise Reduction for Robust Speech Recognition: The Possibilities of Next Generation 5G Data Networks

Radek Martinek[1], Jan Baros[1], Rene Jaros[1], Lukas Danys[1,*] and Jan Nedoma[2]

[1]VSB–Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, Department of Cybernetics and Biomedical Engineering, 708 00, Ostrava-Poruba, Czechia
[2]VSB–Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, Department of Telecommunications, 708 00, Ostrava-Poruba, Czechia
*Corresponding Author: Lukas Danys. Email: lukas.danys@vsb.cz

Received: 30 April 2021; Accepted: 01 July 2021

Abstract: This pilot study focuses on employment of hybrid LMS-ICA system for in-vehicle background noise reduction. Modern vehicles are nowadays increasingly supporting voice commands, which are one of the pillars of autonomous and SMART vehicles. Robust speaker recognition for context-aware in-vehicle applications is limited to a certain extent by in-vehicle background noise. This article presents the new concept of a hybrid system, which is implemented as a virtual instrument. The highly modular concept of the virtual car used in combination with real recordings of various driving scenarios enables effective testing of the investigated methods of in-vehicle background noise reduction. The study also presents a unique concept of an adaptive system using intelligent clusters of distributed next generation 5G data networks, which allows the exchange of interference information and/or optimal hybrid algorithm settings between individual vehicles. On average, the unfiltered voice commands were successfully recognized in 29.34% of all scenarios, while the LMS reached up to 71.81%, and LMS-ICA hybrid improved the performance further to 73.03%.

Keywords: 5G noise reduction; hybrid algorithms; speech recognition; 5G data networks; in-vehicle background noise

1 Introduction

Speech recognition systems are one of the fundamental parts of future smart vehicles. Voice-activated technology is slowly introduced in almost every manufactured models of various car brands. It is often connected to infotainment system of the vehicle and can be used to control various features, spanning from sat-nav to radio, media or phone. While the technology is still maturing, the reliability of different systems can vary greatly. The simpler systems are only relying on predefined set of commands, while the more advanced are capable of learning the driver's voice over time and understand phrases and words easily. It is basically utilized to boost the safety and convenience of the driver, so that he/she can focus on the road and not interact with various physical buttons and knobs [1].

The car however is a very specific everchanging environment. While the higher-end model tends to be sound insulated really well, the lower tier of cars is built around certain manufacturing price, cutting unnecessary costs. The outside environment and certain sounds can therefore penetrate into the driver's cabin, influencing speech recognition systems. These lower-end and cheaper vehicles also tend to have much slower infotainment hardware, slower or simpler on-board infotainment systems or limited microphone arrays. In addition, the certain sounds caused by varying quality of roads (mainly by interaction between tires/wheels and potholes) also influence the precision of speech recognition systems. While the cabin might at first seem like an ideal place for voice recognition system, it is one of the toughest places for its implementation. While it is possible, it is difficult to pull out speech from noisy environment, especially in the lower-tier vehicles, which are the most susceptible to higher ambient noise levels.

Voice recognition and fluent understanding of human speech and voice command is computationally demanding. The vehicular electronics is often built around harsh environmental conditions and automotive grade processors are often outdated and build for specific tasks, offering only limited performance. That's why the systems with certain vocabularies were introduced – the system only has to partially recognize the command, picking from one of the predefined words. These systems are often designed for single words, so the driver must go through multiple steps to achieve the desired outcome [1].

Everything is slowly changing with introduction of modern digital voice assistants, which are well known from mobile devices. Google Assistant [2], Apple's Siri or Amazon Alexa are nowadays relying on powerful cloud solutions for analysis and recognition of complex voice commands. While these systems are certainly useful, they rely on internet connection and are often used via Android Auto or Apple CarPlay [3]. They are also influenced by ambient noise, which has to be filtered out for proper command recognition. The mobile devices and on-board wireless modems are nowadays connected via LTE and will slowly transfer into the 5G era.

The quality and performance of individual car brands is slowly approaching comparable levels, thus making it difficult for individual manufacturers to offer something new and interesting. The user experience and quality of on-board system is one of the only remaining ways to differentiate between each brand. It is certain, that with the ongoing development of smart and even autonomous vehicles, the on-board voice assistants will be an inseparable part of modern cars.

As was mentioned, the conditions in driver's cabin are varying greatly. Apart from ideal conditions, the voice recognition system requires the best possible input source. While these conditions are difficult to achieve, it is possible to leverage the powerful adaptive systems to filter out unrequired noise, effectively extracting the most important information for evaluation [4].

There are multiple scenarios, which can be improved by deployment of adaptive systems. We can introduce a concept of vehicle 4.0, which would employ an advanced array of onboard microphones in combination with either powerful infotainment or reliable 5G link. Let's say there is a set of potholes on a road and multiple vehicles passes through them. Drivers on board of these vehicles are either calling or using voice commands, so they need to filter out any unnecessary noise from their voice signal [5]. When the first car pass through the mentioned potholes, the on-board adaptive system would react straight away, filtering at least part of the noise. However, it is likely, that the installed system is not capable of real-time denoising. As the adaptive algorithms often needs a bit of time for their training the first vehicle in a row would send the small dataset with filter parameters either to cloud or directly to the other passing vehicles via link with lower latency, such as 5G network [6]. The next vehicle can start to process the problem straight away, slowly preparing for the encounter with potholes. When another vehicle passes through the potholes, it could be already prepared for real-time filtration or at least have better filter parameters for the next passing vehicle. This system would heavily rely on highspeed, and low-latency link offered by 5G, as the speed and distance of individual vehicles can vary greatly. The precision of the processing algorithm can be refined even further by introducing other parameters, such as real-time telemetry data, tire size, speed etc. In addition, certain older vehicles could leverage the power of newer models, gathering their optimized on-road data and input information from more complex microphone arrays. As the newer 5G standards are capable of rollback to earlier releases or even 4G, the newer vehicles can act as an important source of information for older, partially outdated models. The whole system can be seen in Fig. 1.

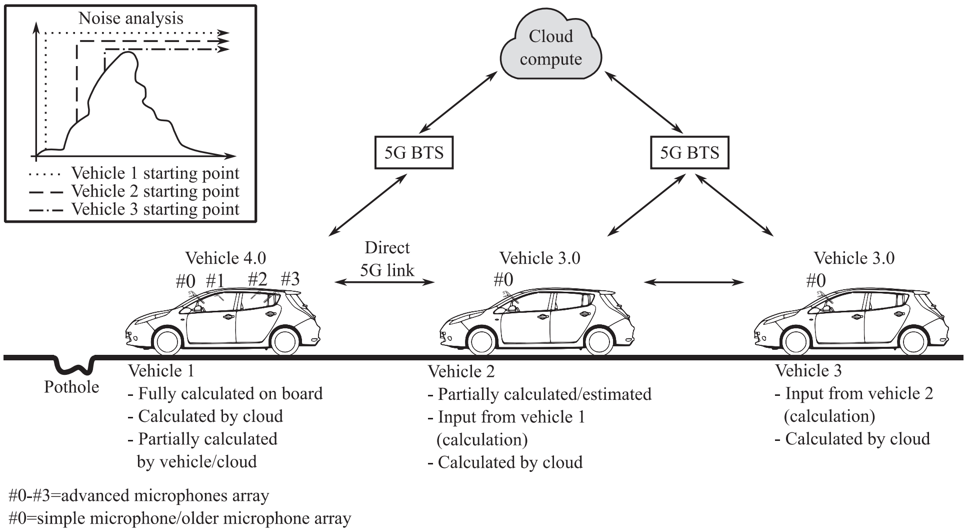

Figure 1: The noise analysis of pothole-vehicle interaction by newer and older vehicles

As presented by Bisio et al. [7] the audio processing technologies are a key feature of modern vehicles. They can be employed by a vast array of commercial applications. Speech is nowadays not only limited to simple commands but can also be used for security services (such as speaker verification and authentication) or accessibility solutions (speech-to-text, text-to-speech, hands-free). Moreover, the modern vehicles are often relying on touchscreen controls. While its certainly useful, some basic functionalities should still stick to robust control, or the system should at least offer an alternative way of controlling these important subsystems. One example is the recently released Skoda Octavia gen 4. Some systems, such as air conditioning are used via touchscreen controls. Some users have reported that the infotainment system can sometimes freeze and must be restarted. While the manufacturer will offer bugfixes to overcome this issue, it can cause discomfort and problems for the driver. The onboard voice assistant Laura offers an alternative way of controlling the previously mentioned system. However, it relies on cloud calculation of advanced voice commands, therefore the vehicle must be connected to mobile network. The offline functionality offers only basic commands, which are present in pretrained vocabularies. According to Bisio et al., the next generation of human-vehicle interfaces will incorporate biometric person recognition for customized on-board entertainment or driver monitoring and profiling applications. The speaker identification, mood of users or number of users are important information, which can be only extracted, when the voice is properly filtered out.

2 Speech Signal Processing

Automatic speech recognition systems are very sensitive to different types of noise. For example, ambient noise makes speech recognition very difficult. This is the reason, why recorded signals are processed by some advanced processing method before speech recognition is performed [8]. Advanced signal processing methods have a great importance for elimination of unwanted signal parts. Basically, there are two fundamental types of methods: adaptive and non-adaptive.

Adaptive methods are characterized by the ability to adapt to a given system. Basically, these methods are based on learning system, which can adapt its own properties to changing working environment. This means that adaptive methods can automatically set the coefficients according to the changing values of the system. During speech recognition, these methods use the primary signal, which contains speech signal with noise, and the reference signal, which contains only noise. While the linear filtering can be used for narrowband interference, it is unsuitable for broadband interference. Adaptive methods can be divided into nonlinear and linear adaptive techniques. Nonlinear adaptive techniques include, for example, artificial neural networks (ANN), methods using hybrid neural networks (HNN), adaptive neuro-fuzzy inference systems (ANFIS) or genetic algorithms (GA). Linear adaptive methods include algorithms based on the principles of Kalman filtering (KF), least mean squares filter (LMS), recursive least squares filter (RLS) or methods based on the principle of adaptive linear neuron (ADALINE) [9–12].

Non-adaptive methods do not use any learning system and work with selected parameters and coefficients. These methods can be divided into single channel and multichannel methods. Single channel non-adaptive methods include Fourier transform (FT), wavelet transform (WT) and empirical mode decomposition (EMD). Multichannel non-adaptive methods include mainly blind source separation methods (BSS), which include independent component analysis (ICA), principal component analysis (PCA) and singular value decomposition (SVD) [13–16].

In this article, LMS algorithm and ICA method were used for creation of automatic speech recognition system. These methods were chosen based on compromise between accuracy, computation cost and calculation speed. Subsections below deals with mathematical apparatus and limitation of used methods.

2.1 Least Mean Squares Filter

LMS algorithm is based on a gradient optimization for determining the coefficients. This algorithm is based on the Wiener filtering theory, stochastic averaging, and the least squares method. This method (same as another adaptive algorithms) is basically attempting to minimize output error $\vec{e}(n)$ calculated by Eq. (1), where $\vec{d}(n)$ is desired signal and $\vec{y}(n)$ is real output signal. Desired signal $\vec{d}(n)$ is known and real output signal $\vec{y}(n)$ is calculated in every iteration of LMS filter by Eq. (2). Adjustment of LMS weights is given at the end of every iteration by update Eq. (3), where μ is the convergence parameter (step size), $\vec{x}(n)$ is the input signal and $\vec{w}(n)$ is the vector of filter coefficients. Convergence parameter μ determines how fast and how well the algorithm converges. A great influence on the computational complexity has the order of the filter N [17–20].

$$\vec{e}(n) = \vec{d}(n) - \vec{y}(n), \tag{1}$$

$$\vec{y}(n) = \vec{w}^T(n)\vec{x}(n), \tag{2}$$

$$\vec{w}(n+1) = \vec{w}(n) + 2\mu\vec{e}(n)\vec{x}(n). \tag{3}$$

During elimination of noisy part of speech signal, primary signal and reference signal are the inputs of LMS algorithm. After application of LMS algorithm, reference signal is adjusted with respect to the primary signal and prepared for subtraction. Then the adjusted reference signal is subtracted from primary signal. After this procedure, a clean speech signal and separated error signal are obtained.

2.2 Independent Component Analysis

This method belongs into group of BSS methods and is based on higher order statistics. The aim of this method is finding linear representation of non-Gaussian data. These data need to contain statistically independent components. During separation of speech signal, ICA method requires at least two microphones. Each microphone $\vec{x}_n(t)$ has to be placed in different location and at a different distance from the speaking person. Every microphone then records every source $s_i(t)$ signals that must be separated. In this article, this method is used to extract component containing noise and component containing required speech signal. Eq. (4) describes composition of the signals $\vec{x}_n(t)$, where $\mathbf{A}_{mix}$ is a mixing matrix. To resolve an issue with an unknown parameter $\mathbf{A}_{mix}$, Eq. (5) is used to estimate independent components from mixed speech signals, where $\mathbf{W}$ is the inverse matrix from the $\mathbf{A}_{mix}$ matrix [21–23].

There is a significant number of ICA based algorithms. Among them are FastICA, JADE, SOBI, Infomax, FlexiICA, kICA, RADICAL ICA, AMUSE etc [22,24–26]. All these algorithms require performing ICA preprocessing in form of centering (creation of zero mean vector) and whitening (creation of vector with unit scattering). The most widely used and very promising algorithm is FastICA, which is also used in this study. First, maximum number of iterations k and criterium of convergence δ must be selected. FastICA algorithm is then based on following steps [21–23]:

1) Random normalized vector $\vec{w}^+$ is created.
2) Vector $\vec{w}^+$ is stored in $\vec{w}$ and calculation of kurtosis is performed, see Eq. (6).
3) Normalization of recalculated vector $\vec{w}^+$ is performed.
4) Checking if scalar multiplying between $\vec{w}^+$ and $\vec{w}$ is smaller than the selected convergence criterion δ, and if cycle run more times than selected maximum number of iterations k.
5) If condition in previous step is false, then repeat steps 2)–4).

$$\vec{x} = \mathbf{A}_{mix} \cdot \vec{s}, \tag{4}$$

$$\vec{s} = \mathbf{W} \cdot \vec{x}. \tag{5}$$

$$\vec{w}^+ = E\{\vec{x}g(\vec{w}^T\vec{x})\} - E\{g'(\vec{w}^T\vec{x})\}\vec{w}. \tag{6}$$

2.3 Hybrid Speech Recognition System

In this article, a hybrid system based on LMS algorithm and ICA method was used for automatic speech recognition system. First, primary signal, which contains speech signal with noise, and the reference signal, which contains only noise, are preprocessed by bandpass finite impulse response (FIR) filter with 300 Hz lower limit frequency and with 3400 Hz upper limit frequency. Then, primary signal $\vec{d}(n)$ and reference signal $\vec{x}(n)$ are used as input into LMS algorithm. Output signal $\vec{y}(n)$ and error signal $\vec{e}(n)$ from LMS algorithm are used as input into ICA method to estimate two components $\vec{y}_1(n)$ and $\vec{y}_2(n)$. One component $\vec{y}_1(n)$ which represents clean speech signal used for speech recognition and another component $\vec{y}_2(n)$ which represents clean noise signal. Fig. 2 shows block diagram of described hybrid system.

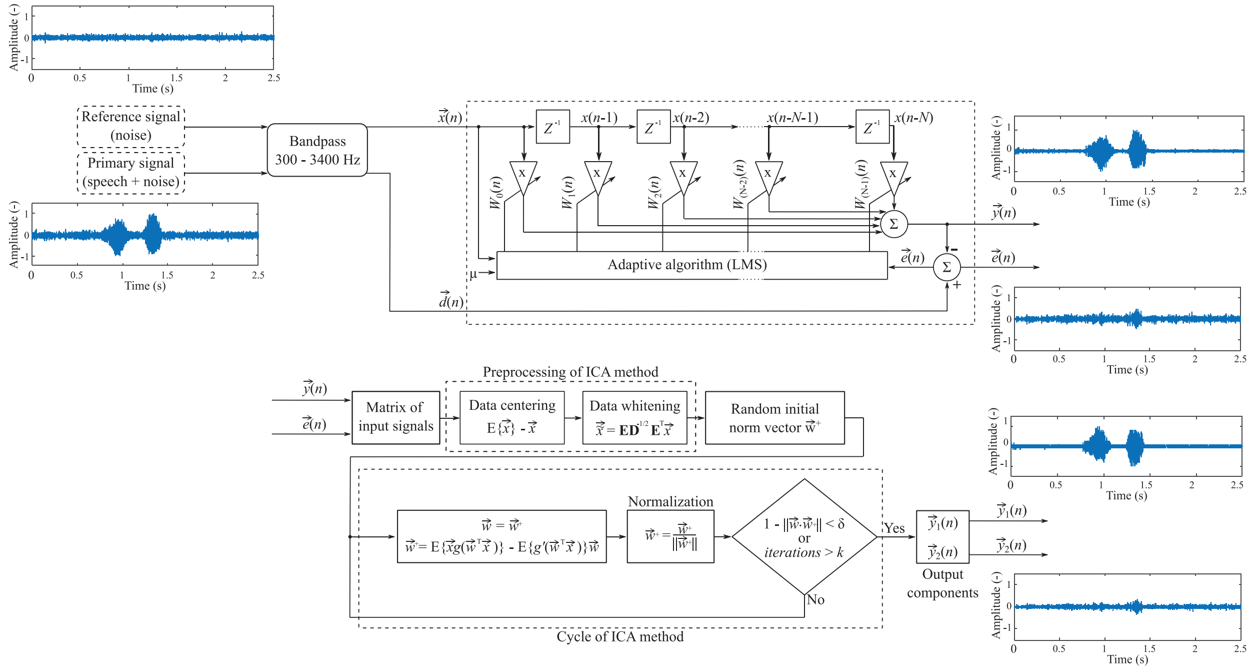
Amplitude (-)
Time (s)
Reference signal (noise)
Primary signal (speech + noise)
Bandpass 300 - 3400 Hz
$\vec{x}(n)$
Z^{-1}
$x(n-1)$
$x(n-2)$
$x(n-N-1)$
$x(n-N)$
$W_0(n)$
$W_1(n)$
$W_2(n)$
$W_{(N-2)}(n)$
$W_{(N-1)}(n)$
Σ
$\vec{y}(n)$
Adaptive algorithm (LMS)
μ
$\vec{e}(n)$
$\vec{d}(n)$
Preprocessing of ICA method
Matrix of input signals
Data centering $E\{\vec{x}\} - \vec{x}$
Data whitening $\tilde{\vec{x}} = \mathbf{ED}^{-1/2}\mathbf{E}^T\vec{x}$
Random initial norm vector $\vec{w}^+$
$\vec{w} = \vec{w}^+$
$\vec{w}^+ = E\{\vec{x}g(\vec{w}^T\vec{x})\} - E\{g'(\vec{w}^T\vec{x})\}\vec{w}$
Normalization
$\vec{w}^+ = \frac{\vec{w}^+}{\|\vec{w}^+\|}$
$1 - \|\vec{w}^+\vec{w}\| < \delta$ or iterations > k
Yes
No
$\vec{y}_1(n)$
$\vec{y}_2(n)$
Output components
Cycle of ICA method

Figure 2: Hybrid system based on LMS algorithm and ICA method

The consecutive estimation of ideal LMS parameters can be seen in Fig. 3. The trajectory and estimation of LMS algorithm is highly dependent on the performance of onboard system-on-chip (SoC). The low-end vehicles can either rely on cloud computing or other vehicles located in the vicinity, which offers untapped higher performance. When the vehicles are calculating the ideal parameters, they could basically rely on each other to specify the parameters and pinpoint the ideal algorithm parameters.

3 Conducted Experiments

Speech signal filtering methods were verified by a set of conducted experiments in two separate vehicles. The first scenario was designed to represent the worst-case scenario. A Skoda Felicia (1994–2001) vehicle was selected as a suitable candidate. Its combustion engine has only 50 kW and it can reach up to 152 kmph. This archaic vehicle has limited sound insulation and the in-vehicle environment is highly influenced by background and environmental noise. The second vehicle was much more recent. A battery electric vehicle (BEV) first generation 80 kW (top speed –144 kmph) Nissan Leaf was selected to represent the newer models. Since this vehicle is powered by electricity, the background noise caused by the engine is minimal. This vehicle can be therefore used in a scenario, when only the environmental noise is important, representing the future all electric vehicles.

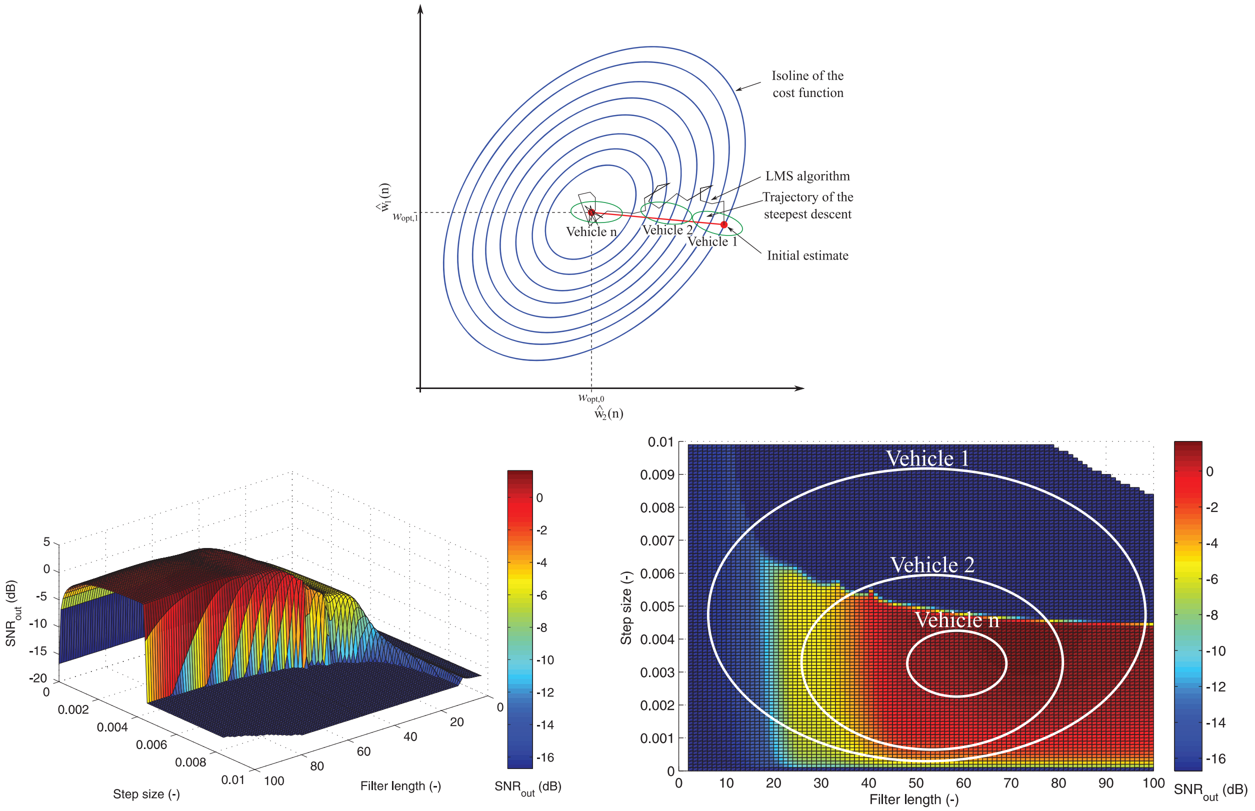

Figure 3: Estimation of ideal LMS parameters in 2D and 3D representation

Four measuring microphones were used in each scenario. The primary microphone (index #0) situated near the rearview mirror was used for both speech and interference recording. Remaining reference microphones (indexes #1, #2, #3) were mounted in each window compartments and recorded acoustic interferences caused by the vehicle itself. The precise diagram with microphone locations can be seen in Fig. 4.

Samples were gathered at different driving speeds (20 kmph, 50 kmph, 100 kmph and 130 kmph) with scenarios with differently opened windows. In the beginning all windows were closed, then they were all opened and, in the end, only one of them was opened, while the rest was closed.

3.1 Hardware

The measuring system consisted of a professional Steinberg UR44 sound card and four Rode NT5 microphones. The system was managed through a PC with software based on virtual instrumentation. The UR44 sound card has a total of 4 analog inputs, which can be used to connect either a microphone array or a musical instrument. It supports various standardized communication protocols such as ASIO, WDM or Core Audio. The resolution of the AD conversion is up to 24 bits with different standardized sampling frequency values (from 44.1 to 192 kHz). The sound card also provides phantom power for connected microphones (from +24 VDC to +48 VDC).

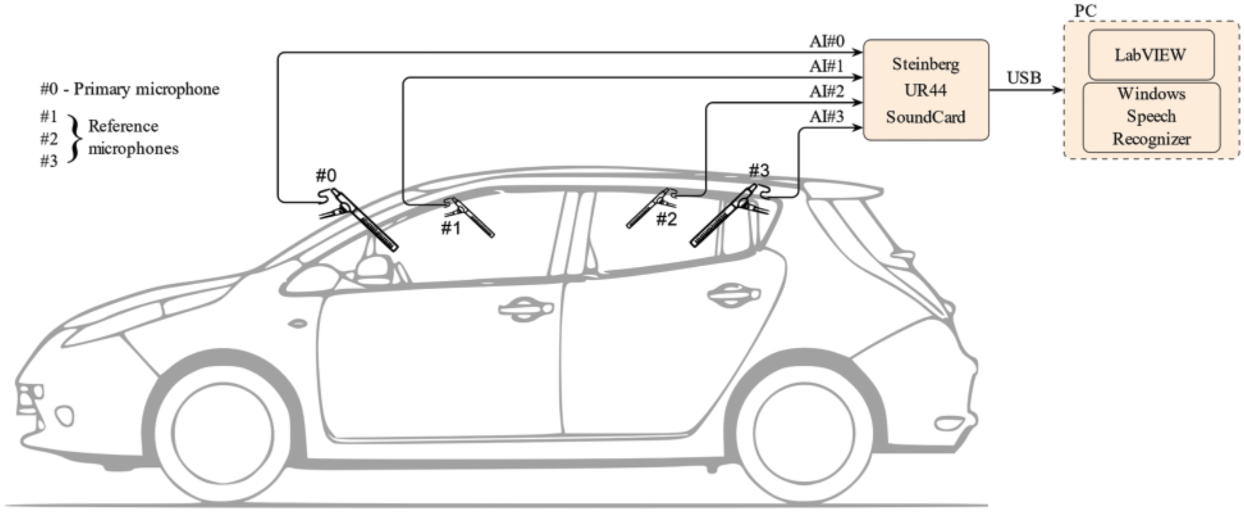

Figure 4: Locations of measuring microphones and the whole measuring platform

The Rode NT5 microphone is a small compact microphone with an XLR connector. The diaphragm is of 1/2” size and consists of an externally deflected capacitor. The membrane is gold-plated, which improves its properties. The microphone is directional with cardioid directional characteristics, the frequency range of the microphone is between 20 Hz and 20 kHz (corresponds to the range of human hearing). In order to use the microphone, it must be connected to the input of a sound card supporting phantom power.

3.2 Software

LabVIEW was chosen as a suitable programming environment, since it offers an extensive library of signal processing functions and is capable of fast development of multi-threaded appliacations. Available ASIO API libraries provides another undeniable advantage since they offer a complete WaveIO library.

The application was designed to be highly modular to make any future modifications as fast as possible. QMH (Queued Message Handler) was chosen as a core application architecture. Each microphone can be therefore considered as a separate unit or input source.

A commercially available recognizer integrated into the Windows OS was used as a speech recognizer. The Speech SDK 5.1 must be installed to maintain a reliable connection to LabVIEW. The recognizer converts the speech into text, which is then analyzed to estimate the correct command.

In order for the signal to be modified or filtered by any adaptive filtering method, it is necessary to adjust the signal path. As the speech recognizer runs in the background of the OS as a service, i tis not possible to select any other than the default audio inputs – i.e., it is not possible to select LabVIEW output. To solve this issue, the signal path was adjusted by a VB-Cable software, which emulates both the inputs and outputs of the sound card.

The front panel of the application can be seen in Fig. 6, which faithfully replicates the standard dashboard of Nissan Leaf. There are 4 alarm indicators on the front panel: revs, speed, temperature and fuel level. After the initial start of the application, is necessary to say the “Start engine” command, which will start the vehicle and the simulation itself. The recording of car idle status will be maintained and the system is therefore ready for input commands. Subsequently i tis possible to control the application according to a predefined vocabulary set. To switch the simulated vehicle off, it is first necessary to stop the vehicle by manuály setting the speed value to 0 kmph and then say “Stop engine” command. This will turn off all indicators and the simulated engine will shut down as well. The application is then waiting for a restart (“Start engine” command). A simplified diagram of the whole application can be seen in Fig. 5.

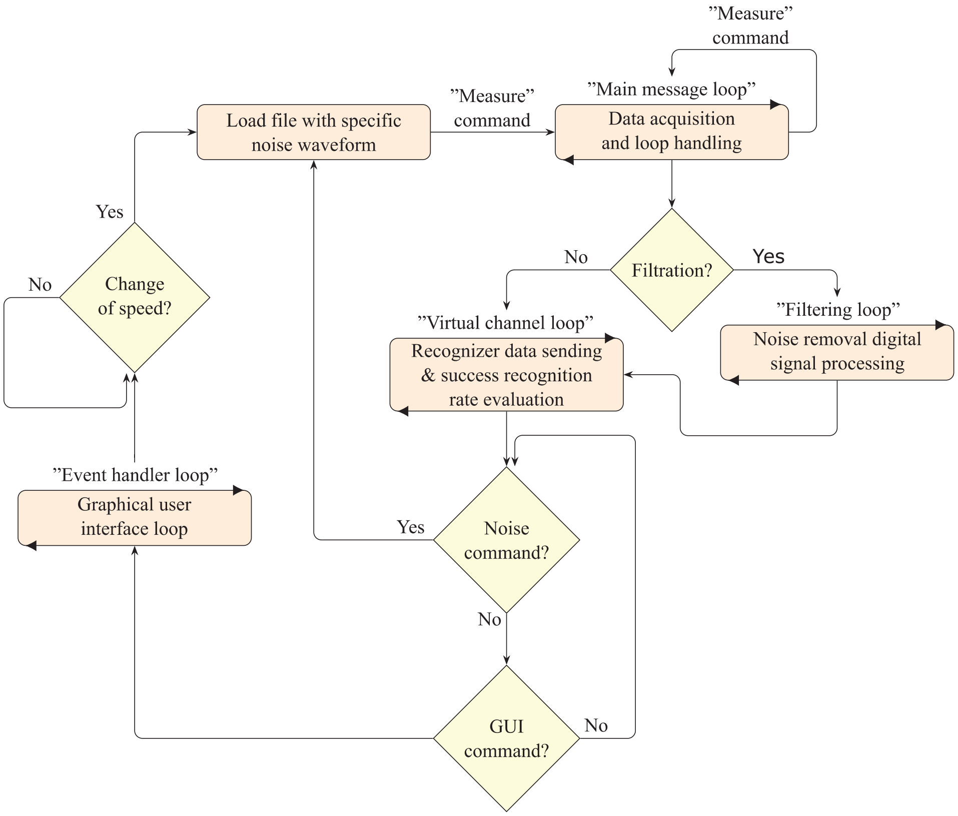
”Measure” command
Load file with specific noise waveform
”Measure” command
”Main message loop”
Data acquisition and loop handling
Yes
No
Change of speed?
No
Filtration?
Yes
”Virtual channel loop”
Recognizer data sending & success recognition rate evaluation
”Filtering loop”
Noise removal digital signal processing
”Event handler loop”
Graphical user interface loop
Yes
Noise command?
No
GUI command?
No

Figure 5: Simplified diagram of the controlling algorithm

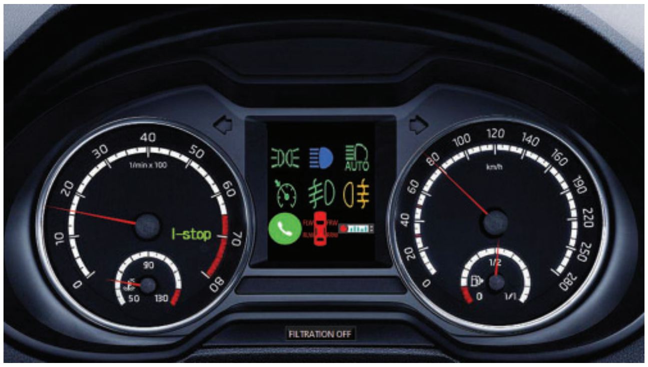

Figure 6: The application front panel with icons for individual commands

The application-supported vocabulary can be seen in Tab. 1. The vocabulary consists of two parts – first part is focused on the front panel (i.e., the vehicle) while the second one can be used to activate various interference sources.

Table 1: Vocabulary for voice control of the car interior simulation application

Command	Meaning
Front panel-car commands	
"Start engine"	Starts the engine
"Stop engine"	Stops the engine
"Winker left"	Left winker switch
"Winker right";	Right winker switch
"Daytime lights"	Daytime lights switch
"High beam lights"	High beam lights switch
"Auto lights"	Automatic high beam lights switch
"Front fog lights"	Front fog lights switch
"Rear fog lights"	Rear fog lights switch
"Accept call"	Accepts the incoming call
"Decline call"	Rejects the incoming call
"Radio ON"	Turn on the radio
"Radio OFF"	Turn off the radio
"Istop ON"	Turn on the I-stop
"Istop OFF"	Turn off the I-stop
Noise simulation commands	
"Cruise control VALUE"	Sets the cruise control to a specific VALUE in km/h
"Open windows"	Opens all windows
"Close windows"	Close all windows
"Open the LEFT/RIGHT FRONT/REAR window"	Opens the specified window
"Close the LEFT/RIGHT FRONT/REAR window"	Closes the specified window

4 Results of Experiments

The recognition results were estimated based on the recognized/unrecognized status. To verify the whole experiment a 100 repetitions were performed. Tabs. 2–Tab. 4 represent various scenarios measured with experimental vehicle and their individual recognition rates. A significant improvement of sucessful recognition can be seen in Fig. 7. When the driver’s front window was opened, the original success rate was only 39% on average. After applying the LMS algorithm, the average success rate was improved to up to 95%. The “Accept call” command offered the lowest recognition rate from all the analyzed commands while running the 80 kmph scenario – 57%. A combination of LMS and ICA offered average recognition rate of 98% and the “Accept call” command reached even 100%. It is important to mention that the LMS and ICA combination can have a negative effect on some specific commands such as “Radio Off”. While the standalone LMS offered a 100% recognition rate, the LMS+ICA combination had only 78%. On the other side, when the worst-case scenario was measured (all windows opened) a LMS+ICA combination offered significantly better results than the standalone LMS. Exact results of the whole vocabulary measured at 80 kmph can be seen in Tab. 2.

Table 2: Recognition success rate for experimental vehicle at 80 km/h

LMS & ICA	Front left window opened			All windows opened		
Command	No filtration (%)	LMS (%)	LMS + ICA (%)	No filtration (%)	LMS (%)	LMS + ICA (%)
“Winker left”	33	100	100	0	85	78
“Winker right”	42	100	100	0	91	80
“Daytime lights”	54	100	100	15	100	100
“High beam lights”	42	100	100	60	100	100
“Auto lights”	36	94	100	18	100	100
“Cruise control 20”	60	98	100	9	63	92
“Cruise control 50”	45	100	100	9	81	100
“Cruise control 100”	54	97	100	6	100	98
“Cruise control 130”	39	100	100	5	91	96
“Front fog lights”	12	100	100	12	100	100
“Rear fog lights”	36	100	100	79	98	100
“Accept call”	27	57	100	42	44	100
“Decline call”	75	90	100	33	57	100
“Radio ON”	33	100	100	15	42	94
“Radio OFF”	12	100	78	33	85	65
“Istop ON”	5	84	100	6	63	100
“Istop OFF”	66	100	85	39	97	93

When the speed was increased to 100 kmph, the results deteriorated even further due to the higher acoustic pressure changes, which caused background hum. On the average, the ICA method again offers better results (by approx. 5%). There are however two specific cases, in which the LMS + ICA combination reached unsatisfactory results – the “Winker left” and “Winker right” commands. While the LMS managed to recognize the driver in about 80% of all cases, the LMS + ICA maintained only 9% and 3% respectively. Similar results were maintained when the windows were opened. The results were probably caused by the nature of the interference (pressure waves caused by changing gusts of wind). A bar graph presenting the results for 100 kmph can be seen in Fig. 8, while the exact results can be seen in Tab. 3.

Table 3: Recognition success rate for experimental vehicle at 100 km/h

LMS & ICA	Front left window opened			All windows opened		
Command	No filtration (%)	LMS (%)	LMS + ICA (%)	No filtration (%)	LMS (%)	LMS + ICA (%)
"Winker left"	24	84	9	0	72	12
"Winker right"	36	78	3	0	35	8
"Daytime lights"	42	92	88	27	90	78
"High beam lights"	50	93	84	12	100	51
"Auto lights"	21	100	90	18	72	75
"Cruise control 20"	33	60	51	0	36	42
"Cruise control 50"	54	88	72	9	66	36
"Cruise control 80"	24	63	69	0	60	54
"Cruise control 130"	18	42	31	0	35	3
"Front fog lights"	21	69	80	18	72	75
"Rear fog lights"	36	58	90	78	54	87
"Accept call"	18	100	95	39	100	73
"Decline call"	34	100	100	57	100	72
"Radio ON"	33	100	78	27	75	70
"Radio OFF"	8	9	66	42	82	54
"Istop ON"	5	100	87	9	81	66
"Istop OFF"	52	6	90	63	28	87

For the last measurements, the maximal permitted speed in Czech Republic was chosen – a 130 kmph. Compared to the previous results, the table was expanded and also offers values with closed windows, as the noise penetrating from the surroundings into the car was significant. Prior to filtering, the recognition success rate with closed windows was only 58% on average, for example the "Radio On" command has not been recognized even once. After applying the adaptive LMS algorithm, the recognition rate was 89%, while the hybrid LMS + ICA offered even 93%. When the driver's window was opened, the average pre-filter recognition value dropped to 27%. A total of 7 commands were not even recognized. After the adaptive LMS algorithm was introduced, the recognition rate improved to an average of 66%. The LMS + ICA hybrid improved the rate by further 6%. After opening all windows, there was a very significant drop in recognition rate for all scenarios. Before the filtration, the recognition rate was only 7%. After the LMS was used, the recognition rate was improved to 29% and ICA managed to improve it further to 30%. Conditions in interior were already quite extreme and the functionality of the whole platform was borderline unusable. The speech was basically overshadowed by huge pressure waves caused by wind. A bar graph presenting the results for 120 kmph can be seen in Fig. 9 and the exact results are visible in Tab. 4.

Table 4: Recognition success rate for experimental vehicle at 130 km/h

LMS & ICA	All windows closed			Front left window opened			All windows opened		
Command	No filtration (%)	LMS (%)	LMS + ICA (%)	No filtration (%)	LMS (%)	LMS + ICA (%)	No filtration (%)	LMS (%)	LMS + ICA (%)
“Winker left”	9	78	93	60	81	60	0	12	52
“Winker right”	30	90	57	27	63	72	0	15	66
“Daytime lights”	12	90	100	60	78	98	0	75	45
“High beam lights”	57	100	100	5	63	100	0	73	30
“Auto lights”	45	93	100	0	57	90	0	65	27
“Cruise control 20”	100	93	92	0	60	90	0	0	12
“Cruise control 50”	100	96	100	0	33	75	0	0	18
“Cruise control 80”	100	100	100	0	66	72	0	0	26
“Cruise control 130”	100	96	100	6	69	57	0	0	21
“Front fog lights”	57	81	100	0	30	60	0	51	40
“Rear fog lights”	63	100	100	28	69	88	4	40	35
“Accept call”	95	51	100	100	95	100	1	27	15
“Decline call”	100	93	100	60	84	96	0	10	12
“Radio ON”	0	81	100	0	88	30	0	0	5
“Radio OFF”	50	69	60	5	38	36	45	30	10
“Istop ON”	0	100	100	0	51	40	0	0	0
“Istop OFF”	50	94	78	100	100	65	63	100	82

In Fig. 10 the immediate course of the “decline call” command before and after the application of the LMS algorithm can be seen. It can be noticed that the algorithm effectively removes noise and interference, and the words “decline” and “call” remain isolated. With gradually increasing speed and thus more noise pollution, it can be seen that the LMS algorithm manages to isolate speech. However the amplitude of the useful signal decreases, since it is partially filtered as well. The filter order N was set tof 530, while the μ parameter was set to 0.001. When listening to the LMS filtered sound signal, it is possible to clearly recognize the isolated words, but the intonation is sgnificantly distorted by the bandpass 300 Hz – 3400 Hz filter.

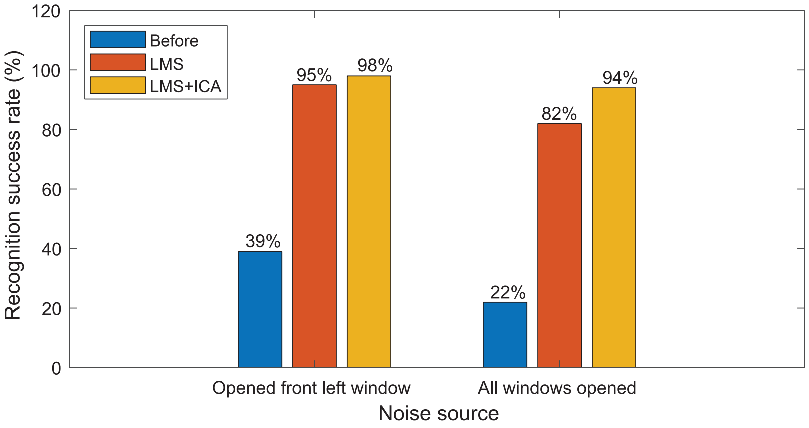

Figure 7: Recognition success rate for experimental vehicle at 80 km/h

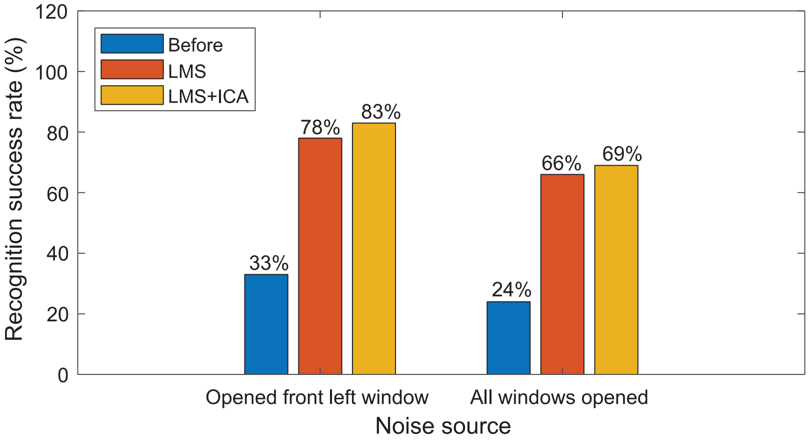

Figure 8: Recognition success rate for experimental vehicle at 100 km/h

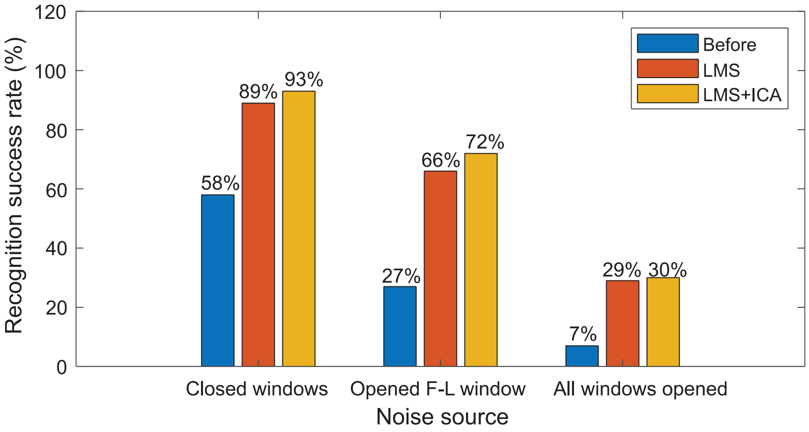

Figure 9: Recognition success rate for experimental vehicle at 130 km/h

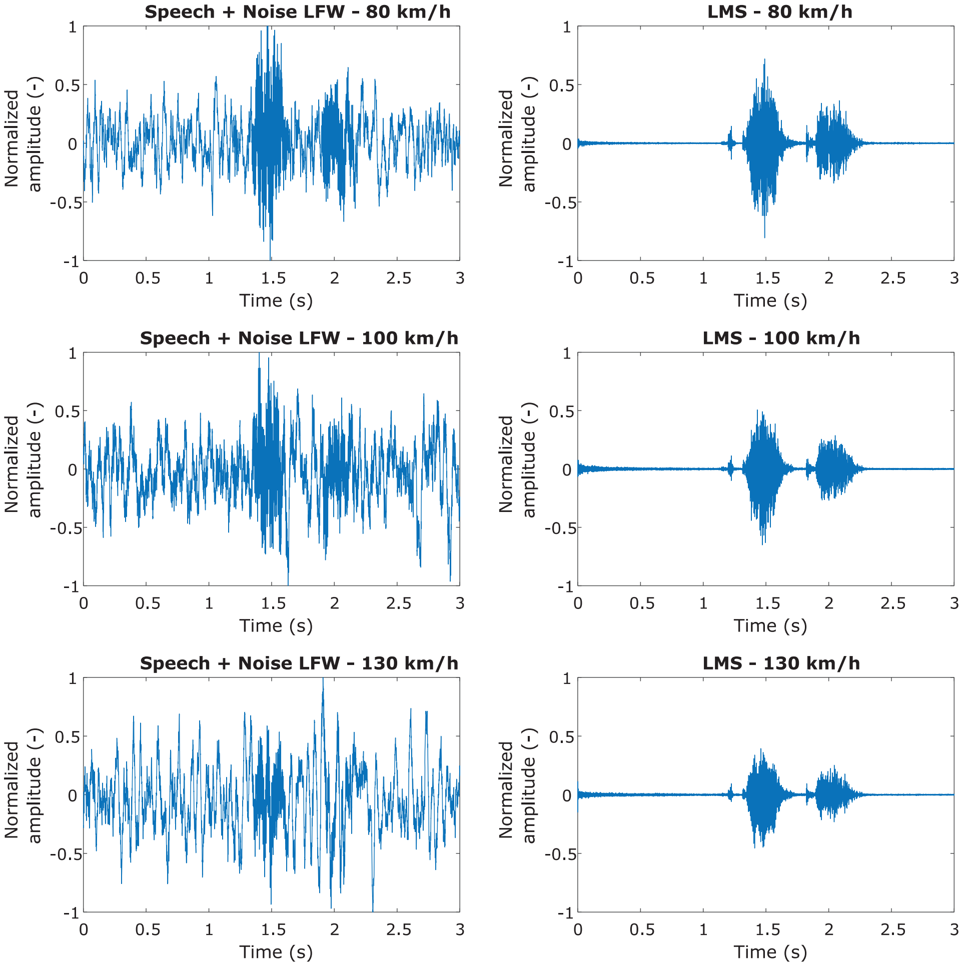

Figure 10: Example of "Decline call" command before and after LMS filtration. The scenario with opened driver's windows

5 Discussion and Conclusion

Based on the presented testing scenarios, both LMS and LMS+ICA combination managed to significantly improve the system reliability. The speech processing is particularly important in the worst-case scenarios. While the non-filtered speech was successfully recognized only in 7% of all cases, the LMS offered up to 29% and LMS-ICA combination up to 30%. In this specific scenario, the difference between LMS and LMS-ICA might be insignificant, and the computational complexity is probably unjustified. The employment of advanced algorithms or their combinations will depend on the hardware equipment of specific vehicles. The signal can be further enhanced by machine learning and neural networks – while these techniques are certainly powerful, they also tend to be much more demanding than conventional methods. The future deployment of AI is currently planned.

Our future research will be focused on testing of different types of hybrid systems for automatic speech recognition. While the LMS+ICA combination offered satisfactory results, other algorithms can be used instead. There are different types of ICA based algorithms, each with different advantages and disadvantages. For example, during the presented initial tests, a fastICA was used. In the future

JADE, flexICA, SOBI, InfoMax, RADICAL, robustICA etc., can be used in place of fastICA. LMS, which was chosen based on its low computational complexity, simplicity and accuracy. Choosing the ideal adaptive algorithm is difficult and this area will be explored further as well. Recursive least squares (RLS) algorithm can offer higher accuracy in certain areas but has a higher computational complexity. There is also a RLS type with lower complexity called fast transversal filter (FTF), which seems like an ideal candidate for further testing.

Technical University of Ostrava (VSB-TUO) has recently acquired two fully customizable Skoda Superb testing vehicles. These vehicles offer the latest Volkswagen hardware, which is partially unlocked for development at university. The conducted tests can now be tested in these modern vehicles and the speech recognition system can be deployed together with Skoda proprietary Laura voice assistant, comparing the performance of the already integrated system to modified scenario with the presented algorithms.

The presented systems can be also deployed in different areas. Based on the previous conducted tests, the system is also capable of speech recognition in production plants – operating even in harsh environments. System with minor adjustments filtered voice commands and adjusted parameters on the fly, while working next to the press machine. The article covering this problematic is currently in processing and will be published shortly. Testing of other scenarios (voice recognition in trains or planes) are currently scheduled, and the results will be compared to current research.

Both research branches will be further explored in newly built VSB-TUO testbed CPIT TL3. This specialized building is focused on three main development areas – smart factory, home care and automotive – offering sophisticated building management systems, energy flow monitoring [27], integrated extensive network of various advanced sensor systems and high speed data transmissions. CPIT TL3 will be opened in 6/2021.

The presented article offered the first insight into our adaptive speech recognition system. The platform was built around professional hardware components (Steinberg and Rode), which was integrated into real vehicle (Skoda). While the platform had its limiting factors, it still managed to significantly improve measured values. When comparing the unfiltered voice commands to the LMS and LMS+ICA combinations, the system reached up to 7 times better performance. The best results were achieved in the worst-case scenarios, when the car was driving at higher speeds with opened windows. When the car was driving at lower speeds (i.e., 100 kmph), the LMS+ICA combinations improved the system reliability by up to 50%.

Acknowledgement: This work was supported by the European Regional Development Fund in the Research Centre of Advanced Mechatronic Systems project, project number CZ.02.1.01/0.0/0.0/16_019 /0000867 within the Operational Programme Research, Development and Education, and in part by the Ministry of Education of the Czech Republic under Project SP2021/32.

Funding Statement: This research was funded by the European Regional Development Fund in the Research Centre of Advanced Mechatronic Systems project, project number CZ.02.1.01/0.0/0.0/16_019 /0000867 and by the Ministry of Education of the Czech Republic, Project No. SP2021/32.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Z. Hua and W. L. Ng, "Speech recognition interface design for in-vehicle system," in *Proc. of the 2nd Int. Conf. on Automotive User Interfaces and Interactive Vehicular Applications - AutomotiveUI '10*, Pittsburgh, pp. 29–33, 2010.

[2] A. H. Michaely, X. Zhang, G. Simko, C. Parada and P. Aleksic, "Keyword spotting for google assistant using contextual speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Okinawa, pp. 272–278, 2017.

[3] T. Kuhn, A. Jameel, M. Stumpfle and A. Haddadi, "Hybrid in-car speech recognition for mobile multimedia applications," in *1999 IEEE 49th Vehicular Technology Conf. (Cat. No.99CH36363)*, Houston, vol. 3, pp. 2009–2013, 1999.

[4] X. Feng, B. Richardson, S. Amman and J. Glass, "On using heterogeneous data for vehicle-based speech recognition: A DNN-based approach," in *2015 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, pp. 4385–4389, 2015.

[5] M. Sidikova, R. Martinek, A. Kawala-Sterniuk, M. Ladrova, R. Jaros *et al.*, "Vital sign monitoring in car seats based on electrocardiography, ballistocardiography and seismocardiography: A review," *Sensors,* vol. 20, no. 19, pp. 1–28, 2020.

[6] C. Y. Loh, K. L. Boey and K. S. Hong, "Speech recognition interactive system for vehicle," in *2017 IEEE 13th Int. Colloquium on Signal Processing & its Applications (CSPA)*, Penang, pp. 85–88, 2017.

[7] I. Bisio, C. Garibotto, A. Grattarola, F. Lavagetto and A. Sciarrone, "Smart and robust speaker recognition for context-aware in-vehicle applications," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 9, pp. 8808–8821, 2018.

[8] T. Lojka, M. Miškuf and I. Zolotová, "Industrial IoT gateway with machine learning for smart manufacturing," in *Advances in Production Management Systems. Initiatives for a Sustainable World,* Cham, vol. 488, pp. 759–766, 2016.

[9] J. H. Kim, J. Park, M. Ahn, Y. Lee, W. Kim *et al.*, "Online speech dereverberation using RLS-WPE based on a full spatial correlation matrix integrated in a speech enhancement system," in *2018 16th Int. Workshop on Acoustic Signal Enhancement (IWAENC)*, Tokyo, pp. 36–40, 2018.

[10] P. P. Mini, T. Thomas and R. Gopikakumari, "Wavelet feature selection of audio and imagined/vocalized EEG signals for ANN based multimodal ASR system," *Biomedical Signal Processing and Control,* vol. 63, no. 1, pp. 1–11, 2021.

[11] S. Silarbi, R. Tlemsani and A. Bendahmane, "Hybrid PSO-ANFIS for speaker recognition:," *International Journal of Cognitive Informatics and Natural Intelligence,* vol. 15, no. 2, pp. 83–96, 2021.

[12] H. Yu, W.-P. Zhu and B. Champagne, "Speech enhancement using a DNN-augmented colored-noise Kalman filter," *Speech Communication,* vol. 125, no. 1, pp. 142–151, 2020.

[13] L. Kerkeni, Y. Serrestou, K. Raoof, M. Mbarki, M. A. Mahjoub *et al.*, "Automatic speech emotion recognition using an optimal combination of features based on EMD-TKEO," *Speech Communication,* vol. 114, no. 1, pp. 22–35, 2019.

[14] A. Winursito, R. Hidayat, A. Bejo and M. N. Y. Utomo, "Feature data reduction of MFCC using PCA and SVD in speech recognition system," in *2018 Int. Conf. on Smart Computing and Electronic Enterprise (ICSCEE)*, Shah Alam, pp. 1–6, 2018.

[15] W. Liu, Q. Liao, F. Qiao, W. Xia, C. Wang *et al.*, "Approximate designs for fast Fourier transform (FFT) with application to speech recognition," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 66, no. 12, pp. 4727–4739, 2019.

[16] H. M. Soe Naing, R. Hidayat, R. Hartanto and Y. Miyanaga, "Using double-density dual tree wavelet transform into MFCC for noisy speech recognition," in *2020 12th Int. Conf. on Information Technology and Electrical Engineering (ICITEE)*, Yogyakarta, pp. 302–306, 2020.

[17] M. Alam, M. Islam and M. Amin, "Performance comparison of STFT, WT, LMS and RLS adaptive algorithms in denoising of speech signal," *International Journal of Engineering and Technology (IJET),* vol. 3, no. 3, pp. 1793–8236, 2011.

[18] M. R. Bachute and D. R. D. Kharadkar, “Performance analysis and comparison of complex LMS, sign LMS and RLS algorithms for speech enhancement application,” *Asian Journal for Convergence in Technology (AJCT),* vol. 3, no. 3, pp. 1–6, 2018.

[19] L. Wang, N. Kitaoka and S. Nakagawa, “Distant-talking speech recognition based on spectral subtraction by multi-channel LMS algorithm,” *IEICE Transactions on Information and Systems,* vol. 94, no. 3, pp. 659–667, 2011.

[20] C.-S. Ahn and S.-Y. Oh, “CHMM modeling using LMS algorithm for continuous speech recognition improvement,” *Journal of Digital Convergence,* vol. 10, no. 11, pp. 377–382, 2012.

[21] S. He, Z. Tong, M. Tong, S. Tang, M. Li *et al.,* “Research on sound separation and identification of trapped miners based on fastica algorithm,” in *2017 7th IEEE Int. Conf. on Electronics Information and Emergency Communication (ICEIEC)*, Macau, pp. 228–231, 2017.

[22] H. Ngarianto, A. Gunawan and W. Budiharto, “Separating multi speeches in intelligent humanoid robot using FastICA,” *IPTEK The Journal for Technology and Science,* vol. 29, no. 1, pp. 1–4, 2018.

[23] A. Hyvärinen, “Survey on independent component analysis,” *Neural Computing Surveys,* vol. 2, no. 1, pp. 94–128, 1999.

[24] S. Kadambe, “Robust speech recognition in adverse environments by separating speech and noise sources using JADE-ICA,” *The Journal of the Acoustical Society of America,* vol. 108, no. 5, pp. 2629–2629, 2000.

[25] L. Tong, V. C. Soon, Y. F. Huang and R. Liu, “AMUSE: A new blind identification algorithm,” in *IEEE Int. Symposium on Circuits and Systems*, New Orleans, pp. 1784–1787, 1990.

[26] H.-Y. Jung, “Adaptive channel normalization based on infomax algorithm for robust speech recognition,” *ETRI Journal,* vol. 29, no. 3, pp. 300–304, 2007.

[27] R. Martinek, P. Bilik, J. Baros, J. Brablik, R. Kahankova *et al.,* “Design of a measuring system for electricity quality monitoring within the SMART street lighting test polygon: Pilot study on adaptive current control strategy for three-phase shunt active power filters,” *Sensors,* vol. 20, no. 6, pp. 1–31, 2020.