sensors

MDPI

Article

Voice Communication in Noisy Environments in a Smart House Using Hybrid LMS+ICA Algorithm

Radek Martinek [1,*], Jan Vanus [1], Jan Nedoma [2], Michael Fridrich [2], Jaroslav Frnda [3] and Aleksandra Kawala-Sterniuk [4]

[1] Department of Cybernetics and Biomedical Engineering, Faculty of Electrical Engineering and Computer Science, VSB-Technical University of Ostrava, 17. Listopadu 15, 708 33 Ostrava-Poruba, Czech Republic; jan.vanus@vsb.cz

[2] Department of Telecommunications, Faculty of Electrical Engineering and Computer Science, VSB-Technical University of Ostrava, 17. Listopadu 15, 708 33 Ostrava-Poruba, Czech Republic; jan.nedoma@vsb.cz (J.N.); michael.fridrich@vsb.cz (M.F.)

[3] Department of Quantitative Methods and Economic Informatics, Faculty of Operation and Economics of Transport and Communications, University of Zilina, 01026 Zilina, Slovakia; jaroslav.frnda@fpedas.uniza.sk

[4] Faculty of Electrical Engineering, Opole University of Technology, Automatic Control and Informatics, 45-758 Opole, Poland; a.kawala-sterniuk@po.edu.pl

* Correspondence: radek.martinek@vsb.cz

Received: 28 August 2020; Accepted: 19 October 2020; Published: 23 October 2020

Abstract: This publication describes an innovative approach to voice control of operational and technical functions in a real Smart Home (SH) environment, where, for voice control within SH, it is necessary to provide robust technological systems for building automation and for technology visualization, software for recognition of individual voice commands, and a robust system for additive noise canceling. The KNX technology for building automation is used and described in the article. The LabVIEW SW tool is used for visualization, data connectivity to the speech recognizer, connection to the sound card, and the actual mathematical calculations within additive noise canceling. For the actual recognition of commands, the SW tool for recognition within the Microsoft Windows OS is used. In the article, the least mean squares algorithm (LMS) and independent component analysis (ICA) are used for additive noise canceling from the speech signal measured in a real SH environment. Within the proposed experiments, the success rate of voice command recognition for different types of additive interference (television, vacuum cleaner, washing machine, dishwasher, and fan) in the real SH environment was compared. The recognition success rate was greater than 95% for the selected experiments.

Keywords: automatic speech recognition; Smart Home (SH); LabVIEW; independent component analysis (ICA); least mean squares algorithm (LMS)

1. Introduction

Spoken communication is the basic and most widely used way of transmitting information between people. The computer industry, where the goal is to make the computer a fully-fledged partner of human beings in spoken language, is no exception. This goal is pursued mainly because such a way of communication can be beneficial and can significantly facilitate a person's life. Voice communication systems are increasingly used in industrial and social practice. In most applications, the usability is limited to a narrow area of tasks, i.e., dictionary limitations or predetermined commands which are to be recognized by a computer. Thus, various systems of machine and equipment control using voice commands or automatic dictation transcription are generally applicable. These systems are especially suitable in cases where a person's eyes and hands are employed in other activities.

Dotihal et al. deals with the smart home (communication between the devices and gateway takes place through Power Line Communication (PLC) and the RF links either through TCP protocol or Message Queue Telemetry Transport (MQTT) protocol) with aims at controlling home appliances via smartphone and voice by using Alexa acting as a client [1]. Erol et al. built and tested a digital voice assistants system with an IoT device to control and simulate the process of assistive robotic workload within voice activation and control in order to improve human–robot interactions with IoT perspectives [2]. Social robotics is becoming a reality and voice-based human–robot interaction is essential for a successful human–robot collaborative symbiosis. The main objective of Diaz et al. is to assess the effect of visual servoing in the performance of a linear microphone array regarding distant ASR in a mobile, dynamic, and non-stationary robotic testbed that can be representative of real HRI scenarios [3]. Novoa et al. proposed to replace the classical black-box integration of automatic speech recognition technology in HRI applications with the incorporation of the HRI environment representation and modeling, and the robot and user states and contexts [4]. Grout in his paper, the role of the human-computer interface for remote, or online, laboratories are considered, for example, hand position/motion/gesture control and voice activation, which are modes of human-computer interaction (HCI) that are of increasing interest [5]. He et al. designed to implement an Arduino board alongside motion sensors and audio receiver to control a robot car by means of a cloud server, and IoT technologies, where the system for control the robot car by preset voice commands integrates Google Voice API [6]. Kennedy et al. investigate a new passive attack, referred to as voice command fingerprinting attack, on smart home speakers with experimental results on a real-world dataset suggest that voice command fingerprinting attacks can correctly infer 33.8% of voice commands by eavesdropping on encrypted traffic [7]. Knight et al. uses a combination of sensors, Raspberry Pis, camera feeds, and multiple interaction methods (voice, text, and visual dashboards) to facilitate laboratory communication for the fully interconnected laboratory of the future [8]. Kodali et al. presented a solution for applications, where is crucial for the development of Smart Cities as a whole along with Smart Homes (for example, switch a light source, HVAC systems, or any other electrical equipment on or off, by being physically present in the premises, remotely or automatically based on time or a sensor's reading) with a speech recognition system to give the users a much more intuitive and natural mean to communicate with and control the connected devices [9]. Leroy et al. propose federated learning for keyword spotting to solve out-of-domain issues with continuously running embedded speech-based models such as wake word detectors with the aim of fostering further transparent research in the application of federated learning to speech data [10]. Based on the interactive experience principle of smart design in the smart building system Li et al. classifies and summarizes intelligent design from the "five senses" interaction, including visual interaction, voice interaction, tactile interaction, cognitive interaction, and emotional interaction, and proposes future research suggestions and directions and promotes the sustainable development of the smart building [11]. Liu designed and implemented of Smart Home Voice Control System based on Arduino [12]. Vanus used Voice communication within the monitoring of the daily living activities in smart home care [13] with the assessment of the quality of speech signal processing within voice control of operational –technical functions in the smart home [14].

This work is focused on the implementation of innovative voice control of operational and technical functions in the real Smart Home (SH) environment for subsequent testing of selected filtration methods. We tested and compared the methods of noise filtration by using an adaptative system (LMS) and a hybrid system (LMS+ICA). For this study's purposes, the plug-and-play platform seemed to be the ideal tool for testing, or more precisely, connection with our virtual devices created in the LabVIEW graphically oriented interface. In this paper, we do not have ambitions to develop algorithms of recognition, but we present a way how to improve the effectiveness of speech recognizer via the mentioned adaptive systems. The partial goals of the work are as follows.

- To ensure control of operational and technical functions (blinds, lights, heating, cooling, and forced ventilation) in the SH rooms (living room, kitchen, dining room, and bedroom) using the KNX technology.
- To ensure recognition of individual commands for the control of operational and technical functions in SH.
- To record individual voice commands ("Light on", "Light off", "Turn on the washing machine", "Turn off the washing machine", "Dim up", "Dim down", "Turn on the vacuum cleaner", "Turn off the vacuum cleaner", "Turn on the dishwasher", "Turn off the dishwasher", "Fan on", "Fan off", "Turn on the TV", "Turn off the TV", "Blinds up", "Blinds down", "Blinds up left", "Blinds up right", "Blinds up middle", "Blinds down left", "Blinds down right", "Blinds down left", and "Blinds down middle").
- To ensure data connectivity among the building automation technology, the sound card, and the speech recognition software tool.
- To upload sample additive interference in a real SH environment (TV, vacuum cleaner, washing machine, dishwasher, and fan).
- To ensure additive noise cancelling in the speech signal using the least mean squares algorithm (LMS) and the independent component analysis (ICA).
- To ensure visualization of the aforementioned processes of Visualization software application with a SH simulation floor plan; in this work, the measurement and processing of the speech signal were implemented using the LabView software tool together with a database of interference recordings.
- To ensure the highest possible recognition success rate of speech command in a real SH environment with additive noise.

2. Related Work

As part of the currently addressed issue of "smart home" automation, automatic or semi-automatic control and monitoring of household appliances and the operational technical functions, such as lights, blinds, heating, cooling or forced ventilation, is provided. Amrutha focuses upon different steps involved for speaker identification using MATLAB Programming with a speech recognition accuracy of more than 90% within Voice Controlled Smart Home [15]. Kamdar elaborates on the different methods of integrating voice recognition technology in home automation systems [16] Kango describes networked smart home appliances like a ubiquitous culture within SH [17]. Smart appliances often use innovative technologies and communication methods [18] that enable a variety of services for both the consumers and the manufacturers. Smart homes can, therefore, be defined as those that have the characteristics of central control of home devices, networking capabilities, interaction with the users via smart interfaces, etc. For natural interaction with the users, one of the most user-friendly methods is vocal interaction (VI), which corresponds to the physical environment of a smart home. System VI, which can be accessed from, for example, the garage, the bathroom, the bedroom, or the kitchen, requires a distributed set of microphones and speakers together with a centralized processing unit [19]. Automatic Speech Recognition (ASR) can be divided into three basic groups [20]. The first group consists of isolated word recognition systems (each word is spoken with a pause before and after the speech, for example in banking or airport telephone services). The second group comprises small glossary systems for application commands and control, and the last group consists of large glossary systems for continuous speech applications. From the ASR perspective, the smart home system is a mixture of the second and third group, wherein it is possible to dictate e-mail messages and use grammatically limited commands for household management, the so-called command and control vocabulary, etc. Predominantly, we can classify the ASR system by means of vocal interaction in two main categories: first, these are specific control applications that form the essence of smart homes (voice control of operational and technical functions and appliances), and, second, there are general vocal applications that can be used in all ASR systems [19]. Using a computer speech recognition technology, a multipurpose wireless system can be designed and created and such a system can turn off and on any household electrical appliance depending on the user's voice command. Thoraya Obaid et al. [21] proposed a wireless voice system for the elderly and the disabled. The proposed system has two main components, namely a voice recognition system and a wireless system. The LabVIEW software was used to implement the voice recognition system. ZigBee modules were used for wireless communication. The proposed system only needs to be "trained" once. Based on the data received by and stored in the wireless receiver connected to the appliances, the required operations are performed. Another home automation system was designed by Dhawan S. Thakurand and Aditi Sharma [22]. The proposed system can be integrated as a standalone portable unit; allows wireless control of lights, fans, air conditioners, televisions, security cameras, electronic doors, computer systems, and audiovisual equipment; and can turn on or off all appliances that are connected to the electrical network. However, more complex commands can be managed through a set of alternatives where the vocabulary is limited. To handle these tasks, syntax, which specifies the given word and phrase with their admissible combinations (alternatives), is required [19]. Dictation involves automatic translation of speech into the written form. Dictation systems include large vocabulary and, in some cases, applications that include additional professional dictionaries for the application [23]. Domain-specific systems can thus increase recognition accuracy [19]. Many factors in the ASR system for VI can be regulated. For example, speech variability is generally of limited use. Language flexibility can be limited by a suitable grammatical proposal, etc. The ability to accurately recognize the speech captured that has been limited depends primarily on the vocabulary size and the signal-to-noise ratio (SNR). Thus, recognition can be improved, first, by reducing the size of the vocabulary and, second, by improving the signal-to-noise ratio. The size of vocabulary constraints in VI systems is based on grammar. Reducing vocabulary, for example by shortening individual commands, can lead to improved recognition [24,25]. Similarly, the quality of speech captured affects the accuracy of recognition [26]. Real-time response is another required characteristic. System performance is affected by three aspects: recognition speed, memory size requirements, and recognition accuracy. These aspects are in conflict with each other, as it is relatively easy to improve recognition speed while reducing memory requirements at the expense of reducing recognition accuracy [27]. The task of designing a voice recognition system is, therefore, to reduce the size of the vocabulary at each moment of the conversation as much as possible. ASR systems often use specific domains and specific applications tailored to improve performance, but vocabulary size is important in any general ASR language, regardless of the technique used in the implementation. Some systems have been designed from the ground up to examine the effects of vocabulary limitation, such as the Bellcore system [25], which contains up to 1.5 million individual words. Recognition accuracy decreased linearly with a logarithmic increase in directory size [25]. ASR systems currently are widely used also in the field of industrial robotics [28], in the field of wheelchair steered [29], in the field of defense and aviation [30], and in the field of telecommunications industry [31]. The IoT platform [32] within the cyber-physical system [33], which can be understood as a combination of physical [34], network [35] and computational processes [36,37], also plays an important role in current voice control applications. Speech contains information that is usually obtained by processing a speech signal captured by a microphone using sampling, quantization, coding [38], parametrization, preprocessing, segmentation, centring, pre-emphasis, and window weighting [39,40]. The next step is speech recognition with

- statistical approach for continuous speech recognition [41] with different approaches [42] for speech recognition system's [43] using the perceptual linear prediction (PLP) of speech [44], for example,
    - Audio-to-Visual Conversion in Mpeg-4 [45],
    - acoustic modeling and feature extraction [46],
    - speech activity detectors [47] or joint training of hybrid neural networks for acoustic modeling in automatic speech recognition [48],

- the RASTA method (RelAtive SpecTrAl) [38], and
- the Mel-frequency cepstral analysis (MFCC), for example,
  - dimensionality reduction of a pathological voice quality assessment system [49],
  - content-based clinical depression detection in adolescents [50],
  - speech recognition in an intelligent wheelchair [51],
  - speech recognition by using the from speech signals of spoken words [52],
- the hidden Markov models (HMM) [53], and
- artificial neural networks (ANN) [54], for example,
  - feed-forward Neural Network (NN) with back propagation algorithm and a Radial Basis Functions Neural Networks [55],
  - an automatic speech recognition (ASR) based approach for speech therapy of aphasic patients [56],
  - fast adaptation of deep neural network based on discriminant codes for speech recognition [57],
  - implementation of dnn-hmm acoustic models for phoneme recognition [58],
  - combination of features in a hybrid HMM/MLP and a HMM/GMM speech recognition system [59], and
  - hybrid continuous speech recognition systems by HMM, MLP and SVM [60].

An essential part of speech signal processing is also the suppression of additive noise in the speech signal using single-channel or multichannel methods [61], for example, single-channel methods like

- speech enhancement using spectral subtraction-type algorithms [62],
- use of complex adaptive methods of signal processing [63,64],
- model-based speech enhancement [65,66],
- increasing additive noise removal in speech processing using spectral subtraction [67], and
- noise reduction of speech signal using wavelet transform with modified universal threshold [68] or denoising speech signals by wavelet transform [69].

Multichannel methods include

- the least mean square algorithm (LMS) [70,71],
- the recursive least squares algorithm (RLS) [72,73],
- the independent component analysis (ICA) [74,75], and
- the principal component analysis (PCA) [76,77] or beamformer (BF) methods for speech acquisition in noisy environments [78], to linearly constrained adaptive beamforming [79] with a robust algorithm [80].

3. The Hardware Equipment in SH

3.1. SH Automation with the KNX Technology

Individual modules of the KNX technology were used for the implementation of SH automation. The KNX technology is designed for complex automation of intelligent buildings and households in accordance with European standard EN50090 (European Standard for Home and Building Systems) and ISO/IEC 14543 standard. It is used not only to control the shading elements (blinds, shutters, and awnings), but also to control the lighting (dimmable lights and lights being switched), the heating in the house, and, also, to control the other equipment in the building. It combines all technological parts in the house into one, logically arranged system, to increase the comfort of living. Based on the preference for building automation voice control by seniors, the elderly and the disabled, to support independent living, GHOST visualization software for voice control of operational and technical functions in SH was designed and tested. The GHOST visualization software connects the environment for recognition of individual voice commands and the KNX technologies [81]. The connection between the computer with voice control and the KNX communication bus, within the voice communication, was implemented using a Siemens KNX IP Router N146 module (5WG1 146-1AB01) (Figure 1). The module described has a KNX interface on one side and an Ethernet connector on the other side. However, when using this router, it is necessary to communicate over the network using the UDP protocol. Due to the automatic assignment of IP addresses, a switch was added between the computer and the KNX IP Router.

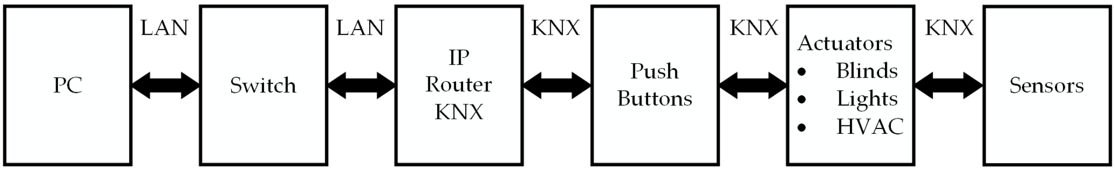

Figure 1. Block diagram of PC and KNX technology connection using an IP router.

The next step is the software adaptation of the individual protocols (KNX and UDP). Input data in the form of component addresses and states of the individual modules can be changed using the ETS application. The method used is based on mapping IP addresses and UDP messages sent via the bus during each change in the sensor part of the system. KNX sensors, KNX bus buttons and switching, blind, dimming, and fan coil KNX actuators were used to control the operational and technical functions in the bedroom, kitchen, hallway, bathroom, and living room.

3.2. *Steinberg UR44 Sound Card*

In a classic USB mode, special drivers are required for Windows or Mac OS X, and the card then supports the ASIO, WDM, or Core Audio standards established (Table 1).

Table 1. Steinberg UR44 sound card specifications.

Sound Card Type	**USB**
Number of analogue outputs	6
Number of microphone inputs	4
Number inputs	4
Number outputs	4
MIDI	YES
Phantom power supply	+48VDC
Sampling frequency	44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz, 176.4 kHz, 192 kHz
Resolution	up to 24 bits at a maximum sampling rate

3.3. *RHODE NT5 Measuring Microphones*

The RHODE NT5 microphone is a small-diaphragm condenser microphone for recording sound sources, consisting of an externally deflected condenser, a 1/2″ capsule with a gold-covered diaphragm, an active J-FET impedance transducer with a bipolar output buffer and dual power supply (Table 2).

Table 2. RHODE NT5 microphone specifications.

Acoustic Principle	Pressure Gradient
Sound pressure level	143 dB
Active electronics	J-FET impedance converter with a bipolar output buffer
Directional characteristics	Cardioid (kidney)
Frequency range	20 Hz–20 KHz
Output impedance	100 Ω
Power supply options	24VDC or 48VDC
Sensitivity	−38 dB re 1 Volt/Pascal (12 mV @ 94 dB SPL) +/− 2
Equivalent noise level	16dBA
Output	XLR
Weight	101 g

4. The Software Equipment in SH

4.1. ETS5 - KNX Technology Parametrisation

The ETS5 SW tool was used for the actual parametrization of the individual modules (sensors, bus buttons, and actuators) of the KNX technology (Figures 2 and 3).

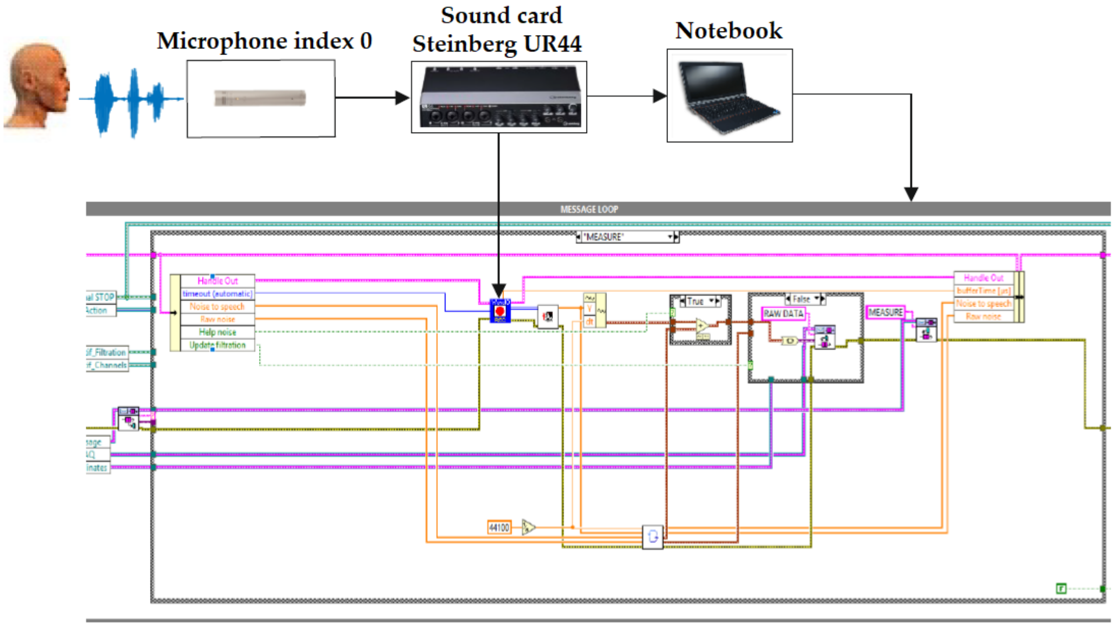

Figure 2. The main loop for continuous data reading.

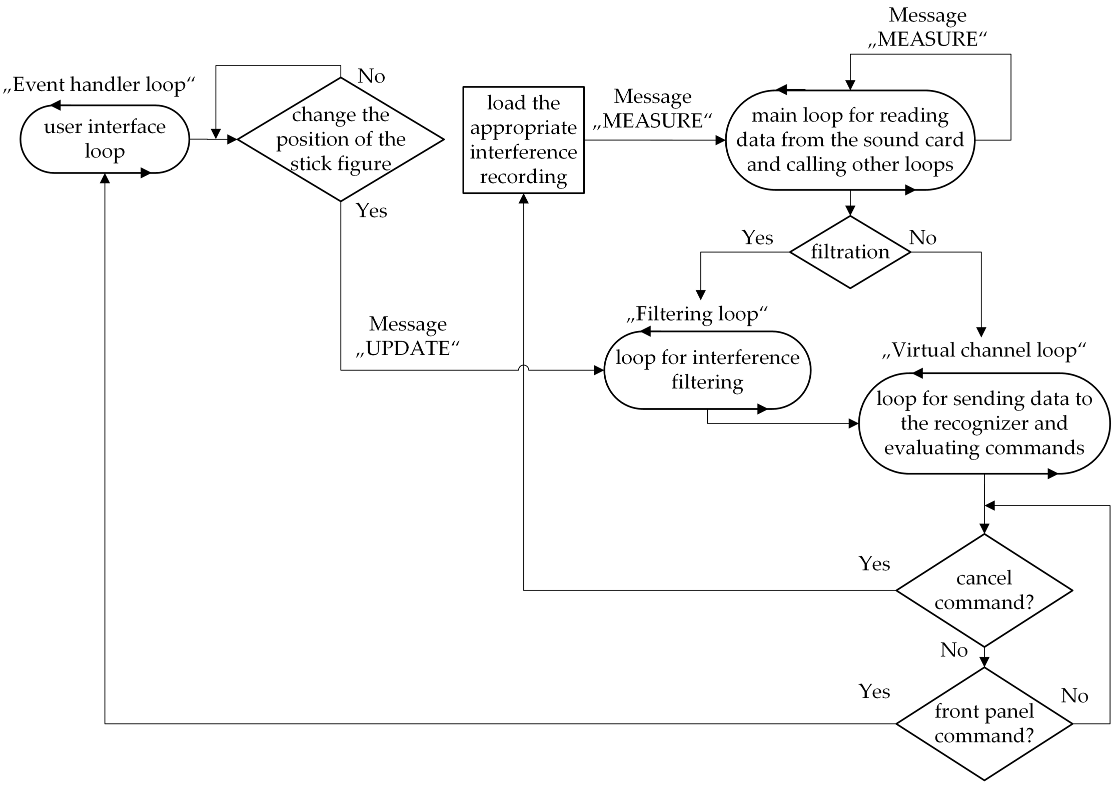
„Event handler loop“
user interface loop
change the position of the stick figure
No
Yes
load the appropriate interference recording
Message „MEASURE“
Message „MEASURE“
main loop for reading data from the sound card and calling other loops
filtration
Yes
No
„Filtering loop“
loop for interference filtering
Message „UPDATE“
„Virtual channel loop“
loop for sending data to the recognizer and evaluating commands
cancel command?
Yes
No
front panel command?
Yes
No

Figure 3. Simplified block diagram of the “smart home” algorithm.

4.2. LabVIEW Graphical Development Environment

The LabVIEW (Laboratory Virtual Instruments Engineering Workbench) graphical development environment is a product made by National Instruments, and this environment enables programming in a specific graphical programming language called “G”. This makes it intuitive even for inexperienced programmers/beginners, allowing programming without a deeper knowledge of syntax. The environment is, therefore, at the level of, for example, the C language, but unlike this language, it is not oriented towards the text, but graphically. The final product of this development environment is called a virtual instrument (Virtual Instrument, abbreviated VI), because its character and activity resemble a classic device in its physical form. Therefore, the virtual instrument is a basic unit of every application created in this development environment and contains [82] the following.

- Interactive Graphical User Interface (GUI)—the so-called front panel, which simulates the front panel of a physical device. It contains objects, such as controls and indicators, which can be used to control the running of the application, to enter parameters and to obtain information about the results processed.
- Block diagram, in which the sequence of evaluation of the individual program components (the program algorithm itself, their interconnection and parameters) is defined. Each component contains input and output connection points. The individual connection points can be connected to the elements on the panel using a wiring tool.
- Subordinate virtual instruments, the so-called subVI. The instrument has a modular, hierarchical structure. This means that it can be used separately, as an entire program, or as its individual subVI’s. Each VI includes its icon, which is represented in the block diagram, and a connector with locations connected for the input and output parameters.

The sequence of the program run is given by the data flow. The block diagram node is executed when it receives all the inputs required. After the node is executed, it creates the output data and passes the data to another node in the data stream path. The movement of data across the nodes determines the order of execution of the VI and the functions in the block diagram.

4.3. Speech Recognition

A commercially available recognizer by Microsoft within the Windows OS was used for the actual speech recognition.

5. SW Application for Automation Voice Control in a Real SH

5.1. Visualization

One of the goals of the work is to create a model that will correspond to the recordings measured from a real SH environment using the LabView software. Based on this requirement, a visualization of a smart home was created. A commercially available Windows recognizer by Microsoft was used for voice control of the visualizations.

5.2. Speech Recognition

In order to be able to communicate with the recognizer, it is necessary to install the Speech SDK 5.1 driver, which works on the principle of converting a voice command into text. It is freely available software allowing developers to apply speech synthesis and recognition in Windows from various programming languages. A freely available VI (Speech recognition engine) was used for the communication between LabVIEW and the recognizer, which only needs to define the input field of the commands in a string format to the Grammar dictionary input connector, according to which the voice commands are compared. Then, the result of the command compared is converted into text at the output from the "Recognized command".

5.3. Virtual Cable Connection

Therefore, in order to be able to filter voice commands to which interference was added and, thus, to send the data filtered to the recognizer, it was necessary to install the VB-CABLE (Virtual Audio Cable) program. It is software for transferring audio streams between applications or devices. It creates a set of virtual audio devices (so-called Virtual Cables), each of which is composed of a pair of input/output endpoints. Any application can then send an audio stream to the output part of the virtual cable, and another application can receive this stream through its input part. All transmissions are made digitally and there is no loss of quality. The program is suitable for recording audio output of the applications in real-time or for transferring the audio stream to another application, in which it is further processed.

On the computer, it is necessary to enable recording from this virtual connection in the audio settings and to disable other devices (microphone integrated in the laptop and the Steinberg UR44 sound card, or other devices).

5.4. The Main Loop for Data Reading

One microphone, which is set to index 0, was used for voice control of the visualizations. All visualizations contain a main loop, the task of which is to continuously read data, normalized it, add interference to speech and send this data to other loops. The measuring chain is shown in Figure 2.

As described above, the output data from the sound card is represented by a signed int32 resolution $(-231 + 231 - 1)$. Therefore, it is necessary to recalculate the amplitude values so that the input signals were first indexed and divided by the value of $\frac{2^{32}}{2}$.

If the state for loading interference recordings is activated, these interferences are added from the database to the real speech. As the interference recordings are approximately 5 s long, but the data collection from the sound card can take any time, it was necessary to synchronize the lengths of these signals. This is achieved using subVI rotation sums, when the time window of data collection is first detected and, based on this window, the time window of interference recordings is then defined. This is subsequently read and moved forward by another time window. As soon as you move to the end of the recording, the entire cycle starts from the beginning.

5.5. Visualization of a "Smart Home"

The application consists of a user interface loop, which is used to turn on/off the filtration and to capture the change in the position of the person icon, whose task is to simulate the part of the house in which the user is currently located. For example, if the person icon is placed in the bathroom area, it is not possible to activate lights or other devices located in other parts of the house (Figure 3).

If the position changes, the message "UPDATE" is triggered in the loop intended for filtering, where the data on the current coordinates of the occurrence of the simulated person (the person icon) is transferred, and the room the person is located in is evaluated. Subsequently, the data is converted into variables and stored in a cluster.

5.6. Glossary of Commands

The glossary for a "smart home" is defined by commands for switching the lights on/off and closing/opening the blinds, which can be used in all rooms. Other commands can then be used within the specific room (Table 3).

Table 3. Glossary of commands for voice control of "smart" household.

Command	Room
"Light on"	All
"Light off"	All
"Turn on the washing machine"	bathroom
"Turn off the washing machine"	bathroom
"Dim up"	bedroom
Dim down	bedroom
"Turn on the vacuum cleaner"	bedroom
"Turn off the vacuum cleaner"	bedroom
"Turn on the dishwasher"	kitchen
"Turn off the dishwasher"	kitchen
"Fan on"	Hall
"Fan off"	Hall
Turn on the TV	Living room
Turn off the TV	Living room
"Blinds up"	Kitchen/hall/living room/bedroom
"Blinds down"	Kitchen/hall/living room/bedroom
"Blinds up left"	Kitchen/hall/living room
"Blinds up right"	Kitchen/hall/living room
"Blinds up middle"	Kitchen/hall/living room
"Blinds down left"	Kitchen/hall/living room
"Blinds down right"	Kitchen/hall/living room
"Blinds down left"	Kitchen/hall/living room
"Blinds down middle"	Kitchen/hall/living room

5.7. Application Control

The "smart home" application consists of 5 rooms, namely, a bathroom, a bedroom, a hall, a living room, and a kitchen, in which voice commands can be issued. After turning on the application, the person icon is placed in the initial position, which is located in the hall. This state is the default, so it is possible to issue voice commands, but it is not possible to respond to them. For voice control, it is always necessary to place the person icon in a preselected position, where the command is executed based on the evaluation of the coordinates. For example, if the user wants to pull the blinds in the bedroom, they must move the person icon to this position, see Figure 4. This prevents unwanted conditions where the user moves the person icon into the kitchen, for example, and wants to control the device in other parts of the house.

Figure 4 shows the front panel of our virtual SMART Home. The whole house is functional and fully distributed with actual recordings from a real house. Within the recording process, several thousands of real recordings of the selected interference sources (television, vacuum cleaner, washing machine, dishwasher, and fan) were made. It is possible to place a "figurine" into the picture that simulates a user controlling the household via voice. "Noises" coming from the individual sources (household appliances) were recorded from various distances and positions. The retention measurements were taken in a semi-reflective room when the background noise level was being changed during the recording via the user's movement within the room. According to the scenario, every command was repeated 100 times for the individual position, and 20 different speakers participated (10 men and 10 women, various age). These experimental scenarios bring us results in the form of a ratio. The aim was to provide a realistic view of the importance of filtering in commercial speech recognizers in a real environment. Within the virtual device, the user's movement in the room was being simulated, and the level of background noise was changing. In general, the interference level ranged between 0 and 20 dB. Our attempt was to come as close as possible to real scenarios. In other words, we wanted to create a virtual device that would provide additionally checked records for the purposes of development and testing of the filtration methods.

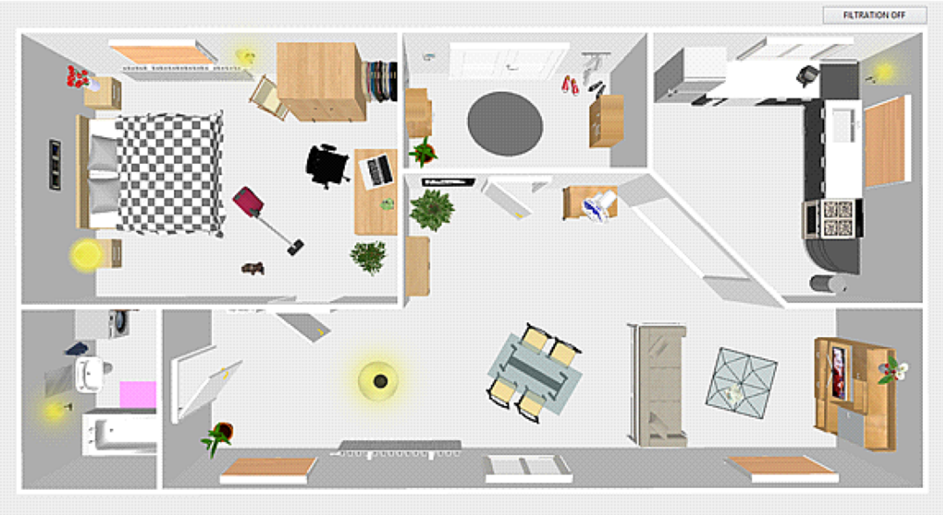

Figure 4. Front panel of the "smart home" application.

6. The Mathematical Methods Used

6.1. Least Mean Squares Algorithm

The LMS algorithm is one of the most widespread and most widely used adaptive algorithms employed in current practice. The strength of the LMS algorithm lies in its simplicity and mathematical incomplexity [71]. These algorithms are based on a gradient search algorithm, also called the maximum gradient method. The dependence of the adaptive FIR filter output error signal standard deviation on the filter coefficients is a quadratic curve with one global minimum [72]. The output equation is defined according to Equation (1).

$$y(n) = \mathbf{w}(n)\mathbf{x}(n), \tag{1}$$

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n). \tag{2}$$

Filter recursion is

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

where μ represents the step size of the adaptive filter, $\mathbf{w}(n)$ is a vector of filter coefficients, and $\mathbf{x}(n)$ is the input vector of the filter.

Figure 5 shows a general diagram of an adaptive filter system where *y(n)* represents the output signal of the filter, *d(n)* represents the noisy signal measured, *n(n)* represents the noise from the reference sensor and *e(n)* represents the deviation of the output signal from the measured one.

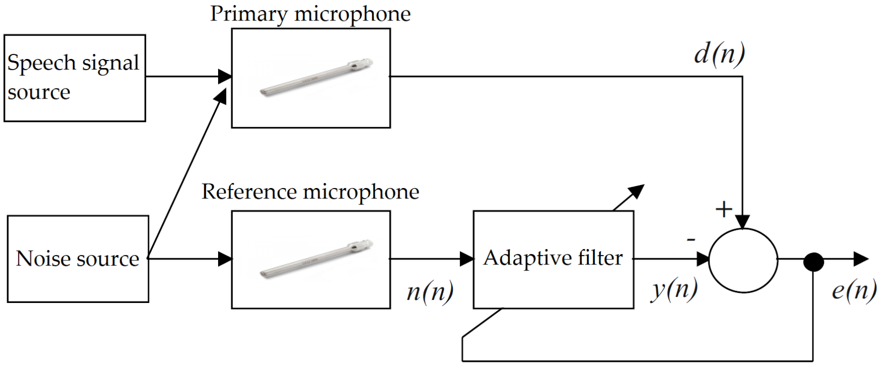

Figure 5. General block diagram of an adaptive system.

6.2. Independent Component Analysis

Independent component analysis is one of the blind source separation methods (BSS), which are methods used to estimate independent sources from multichannel signals. BSS methods in the field of digital signal processing consist in a situation where several signals are mixed together, and the task is to find out what the source signals looked like.

Independent component analysis, as one of the possible solutions to the "cocktail-party problem", is a statistical and computational method for detecting hidden factors that are the basis of groups of random variables, measurements, or signals. This method defines a model for observing many randomly variable data, which is typically defined as a large sample database. In this model, the data variables are considered as linear mixtures of some unknown hidden variables, and the mixing system is not known. Hidden variables are considered non-Gaussian and independent of each other and are called independent components of the data observed. These independent components, also called sources or factors, can be found using the ICA method. Independent component analysis is superficially related to principal component analysis and factor analysis. However, ICA is a much more powerful method capable of finding the underlying factors or resources, even if these other methods fail completely. There is the following transformation,

$$x(k) = \mathbf{A}s(k) + v(k), \tag{4}$$

where $\mathbf{A}$ represents a mixing matrix. The goal is to find the separation matrix, i.e., a matrix $\mathbf{H}$ having a size of $N{*}M$ to which the following applies, $\mathbf{H} = \mathbf{H}^{-1}\mathbf{A}$. The two basic limitations of the ICA method include the impossibility to recover the energy of source signals and the impossibility to maintain the order of the source signals. Thus, the output components have a different amplitude with respect to the input signals, and when the ICA method is applied again, the components have a different order and polarity of signals.

These limitations are compensated by multiplying the resulting separation matrix $\mathbf{H}$ by two matrices. Matrix $\mathbf{P}$ is a permutation matrix that adjusts the order of the separated components, and matrix $\mathbf{D}$ is a diagonal matrix that adjusts the energies of separated signals. In summary, therefore, the following applies:

$$\mathbf{H} = \mathbf{A}^{-1}\mathbf{DP}. \tag{5}$$

6.3. Prerequisites for ICA Method Processing

Before the actual application of the ICA method (see Figure 6), preprocessing in the form of centering and bleaching the input signals is performed [76]. The centering is supposed to remove the DC component from the signal edited. In this step, the signal's mean value is subtracted from the input signal. Therefore, the following applies.

$$x_c(k) = x(k) - \frac{1}{K}\sum_{1=k}^{K} x(k) \tag{6}$$

After processing, the inverse process can be performed using a separation matrix **H** and estimates $y(k)$:

$$y_c(k) = y(k) - H\frac{1}{K}\sum_{1=k}^{K} x(k) \tag{7}$$

Bleaching is a process of modifying a signal after the application of which the input signals are uncorrelated and are scattered per unit. Therefore, if the sensor signals *x(k)* are bleached, then their correlation matrix is equal to the unit matrix: $E\{xx^T\} = \mathbf{I} =$. This transformation can be written as follows,

$$x_B(k) = \mathbf{Bx}(k), \tag{8}$$

where $\mathbf{Bx}(k)$ denotes a bleached vector and **B** represents the so-called whitening dimension matrix $N * M$ for which the following applies: $\mathbf{BB}^T = \mathbf{I}$. The Singular Value Decomposition (SVD) method can be used to calculate the bleach matrix and to design a bleach matrix using eigenvectors and eigenvalues of the correlation matrix of mixture vectors.

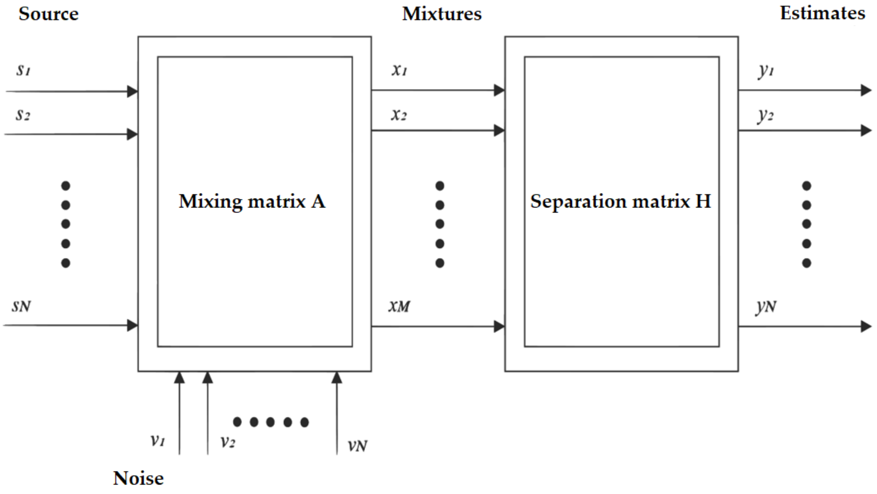

Figure 6. Basic model of the independent component analysis (ICA) method.

7. Experimental Part—Results

7.1. Selected Filtering Methods and Recognition Success Rate

To suppress interference, the ICA method was selected together with the adaptive method with the LMS algorithm. Despite its simplicity and mathematical incomplexity, the LMS algorithm produced good-quality results of the global SNR.

7.2. Search for Optimal Parameter Settings for the LMS Algorithm

As, in the visualizations, it is not possible to determine, in advance, what the next command will be (for this reason the ideal parameters of the LMS algorithm cannot be set), it was first necessary to perform offline identification. This was performed by finding the optimal values for each command and interference according to the global SNR. From these values found, the best filter length M was then selected as well as convergence constant μ. The filtration takes place in two steps (Figure 7), where the speech signal contaminated with interference and the reference noise are first fed to a bandpass filter set at 300 Hz–3400 Hz, which is approximate frequency range corresponding to human speech. Then, the filtered signals are sent to the LMS algorithm, where *y(n)* is the filtered signal, and *e(n)* is the filtration error.

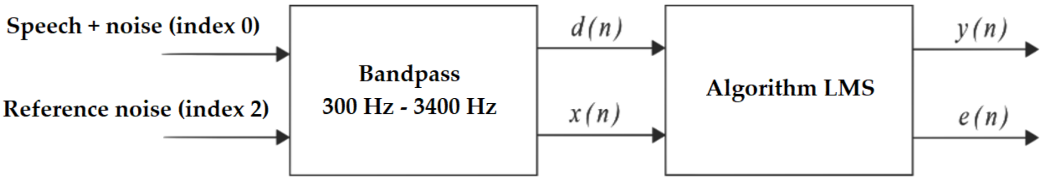

Figure 7. Filtration measuring chain for the LMS algorithm.

Table 4 shows that, with increasing interference energy, there will be greater demands on the adaptive filter. It means higher filter length M and convergence constant μ. When testing, the filter appeared to require a higher filtration length with increasing interference energy, but there is a problem when, at high values (filter length of 1.000 and higher), the useful signal, which is partially filtered, is distorted, and the filtration error increases. The same applies to the convergence constant, where, at high values (above 0.1), the scales get disbalanced and the filter thus becomes unstable. Computing time is another problem. The greater the length of the filter and the smaller the convergence constant, the longer the calculation will take and vice versa. This creates a conflicting situation where the effort is for the best possible filtration in a minimum of time.

Table 4. Optimal parameter settings for the LMS algorithm, visualization of a "smart home".

Interference	Filter Length M	Convergence Constant μ [-]
Washing machine	240	0.01
Vacuum cleaner	80	0.001
Fan	210	0.01
Dishwasher	40	0.01
TV	110	0.01

7.3. Independent Component Analysis

Only one microphone was used for voice control of the visualizations and, therefore, it is not possible to solve the classic "cocktail party problem". For this reason, hybrid filtration was used (Figure 8), where the ICA method was implemented behind the output of the adaptive filter (Table 5). After passing through the bandpass, the signals are sent to the LMS algorithm and filtered out. It is clear from the waveforms that the algorithm significantly suppresses the interference, but, at the same time, the filtration error increases. This is due to the effort of the LMS algorithm to suppress the interference as much as possible while partially filtering the speech. This is one of the features of adaptive algorithms that must be taken into account. In LabVIEW, a function in the Signal Processing $\rightarrow$ Time Series Analysis library was used.

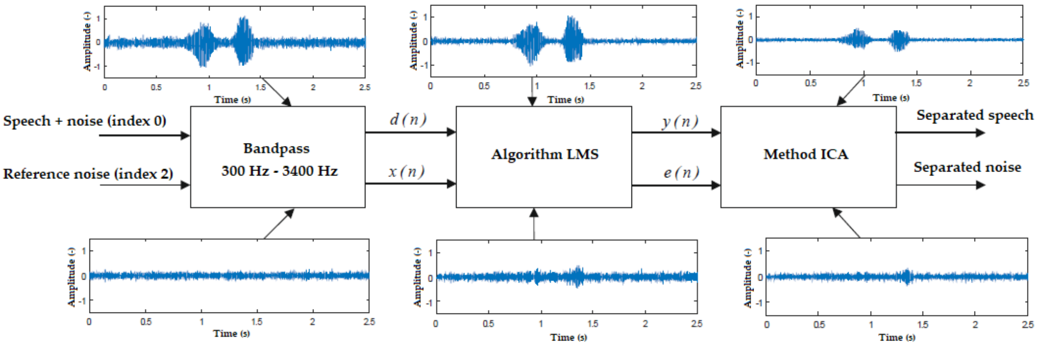

Figure 8. Measuring chain of hybrid LMS and ICA filtrations.

Table 5. ICA function block parameter settings.

Parameter	Value
Method	FastICA
Number of components	2
Number of iterations	1000
Convergence tolerance	0.000001

7.4. Recognition Success Rate

One-hundred iterations were performed for each command, and, based on the recognized /unrecognized status, the recognition success rate was evaluated. The commands were spoken into the microphone at a constant distance of 15 cm. The recognizer had the lowest recognition rate for words ending in "off" ("light off", "i-stop off", and "radio off"). This can be caused by the phonetic aspect of the command, which has low energy. Another reason is the property of adaptive filters when the word is suppressed (slightly filtered).

8. Discussion

Three commands were tested for washing machine interference: "Light on", "Light off", and "Turn off the washing machine". Table 6 shows that the success rates before filtration were 28%, 21%, and 85%, respectively. When the washing machine is switched off, the high success rate of the filtration is caused mainly by the fact that the recognizer itself has a learning algorithm, wherein it returned previous values, which can be seen in the filtration results, where the success rate of the recognition was worse.

Table 6. The results of recognition success rate for washing machine interference.

LMS and ICA	Washing Machine Maximum Volume		
Command	**Before [%]**	**LMS [%]**	**LMS + ICA [%]**
"Light on"	28	100	100
"Light off"	21	45	70
"Turn off the washing machine"	85	60	78

For vacuum cleaner interference (Table 7), the average success rate before filtration was only 1%, when only the commands "dim up" (5%) and "dim down" (3%) were recognized. After filtration, the average success rate for the LMS algorithm was 80% and the average success rate for the ICA algorithm was 86%. The lowest success rate was recorded for the "light off" command, where the filtration for the algorithm was only 27%, and 45% for the ICA method.

Table 7. The results of recognition success rate for vacuum cleaner interference.

LMS and ICA	Vacuum Cleaner Maximum Volume		
Command	**Before [%]**	**LMS [%]**	**LMS + ICA [%]**
"light on"	0	100	100
"light off"	0	27	45
"blinds down"	0	95	100
"blinds up"	0	93	100
"Dim up"	5	100	100
"dim down"	3	100	100
"Turn off the vacuum cleaner"	0	45	60

For fan interference (Table 8), the average success rate before filtration was only 11%, when the commands "light on" (42%) and "light off" (24%) were most recognized. After filtration, the average success rate for the LMS algorithm was 82% and the average success rate for the ICA algorithm was 91%. The lowest success rate was recorded for the "fan off" command again, where the filtration for the algorithm was only 18%, and 13% for the ICA method.

Table 8. The results of recognition success rate for fan interference.

LMS and ICA	Fan Maximum Volume		
Command	**Before [%]**	**LMS [%]**	**LMS + ICA [%]**
"light on"	42	100	100
"light off"	24	28	90
"blinds down"	2	100	100
"blinds up"	5	91	100
"blinds down left"	3	88	100
"blinds down right"	0	83	100
"blinds down middle"	0	100	100
"blinds up left"	6	95	100
""blinds up right"	9	100	100
"blinds up middle"	15	100	100
"Fan off"	18	18	13

For dishwasher interference (Table 9), the average success rate before filtration was only 2%, when the command "Turn off the dishwasher" was most recognizable, but, on the other hand, it had zero success rate after filtration. The average success rate for the LMS algorithm and the ICA method was the same, namely, 85%.

Table 9. The results of recognition success rate for dishwasher interference.

LMS and ICA	Dishwasher Maximum Volume		
Command	**Before [%]**	**LMS [%]**	**LMS + ICA [%]**
"light on"	0	100	100
"light off"	0	65	66
"blinds down"	0	100	100
"blinds up"	0	100	100
"blinds down left"	0	100	100
"blinds down right"	3	100	100
"blinds up left"	5	100	100
"blinds up right"	0	100	100
"Turn off the dishwasher"	10	0	0

With the TV on (Table 10), the average success rate before filtration was only 20%, wherein the commands “blinds up middle” (42%), “blinds down left” (24%), “blinds down right” (21%), and “blinds down middle” (20%) were most recognizable. This is due to the fact that the recognizer was able to capture such long words between pauses of dialogues from the television. The average success rate for the LMS algorithm was 84% while, for the ICA method, it was 82%. The lowest recognition success rate after filtration was with the “blinds down” command, where the recognizer usually evaluated another alternative (“blinds down middle”, “blinds down right”, and “blinds down left”) (Figure 9).

Table 10. The results of recognition success rate for vacuum cleaner interference.

LMS and ICA	**TV Maximum Volume**		
Command	**Before [%]**	**LMS [%]**	**LMS + ICA [%]**
“light on”	60	100	100
“light off”	0	74	68
“blinds down”	0	62	51
“blinds up”	0	74	60
“blinds down left”	24	97	80
“blinds down right”	21	98	96
“blinds down middle”	20	100	98
“blinds up left”	15	91	88
“blinds up right”	8	74	90
“blinds up middle”	42	100	98
“Turn off the TV”	27	52	69

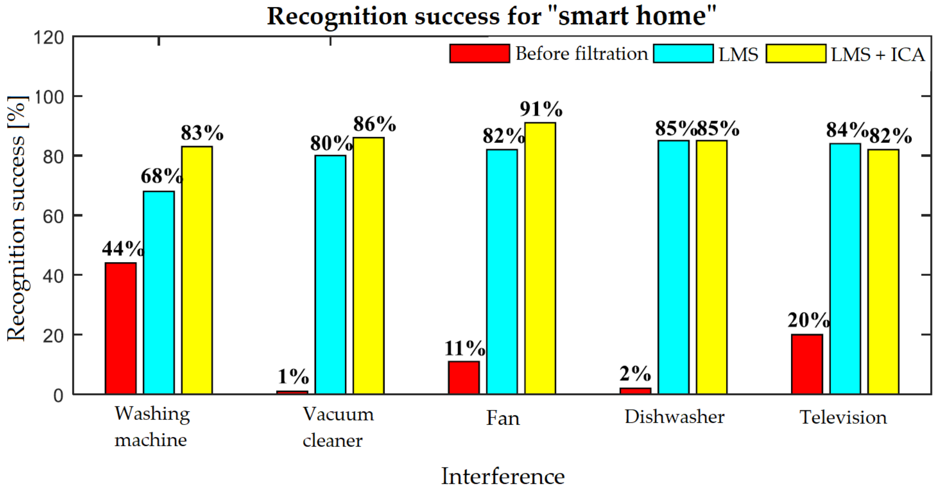

Figure 9. Recognition results for speech commands for additive interference in the real SH environment.

The spectrograms (Figures 10–14) show that the behavior of the appliances is similar after filtration, especially as for the washing machines and the vacuum cleaner. For the fan, the results are identical. This is mainly due to the uniform distribution of the noise, which is close to the Gaussian one. Furthermore, by theory, adaptive filters do not function well with these noises, and, as for the ICA method, this is a basic limitation when these interferences cannot be handled well. The spectrogram also shows that the best filtered interference was in the dishwasher. In the case of television interference, it can be seen, on the other hand, that the quality of filtration favors the LMS algorithm. The main reason is that only one microphone is used, so the classic principle of the ICA method cannot be addressed.

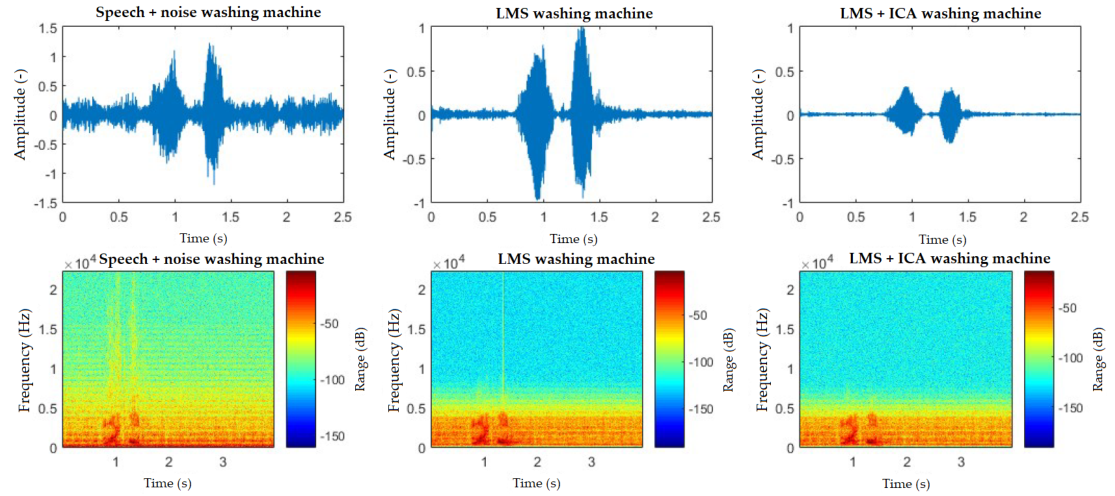

Figure 10. Comparison of spectrograms with closed windows, "light on" command, washing machine interference.

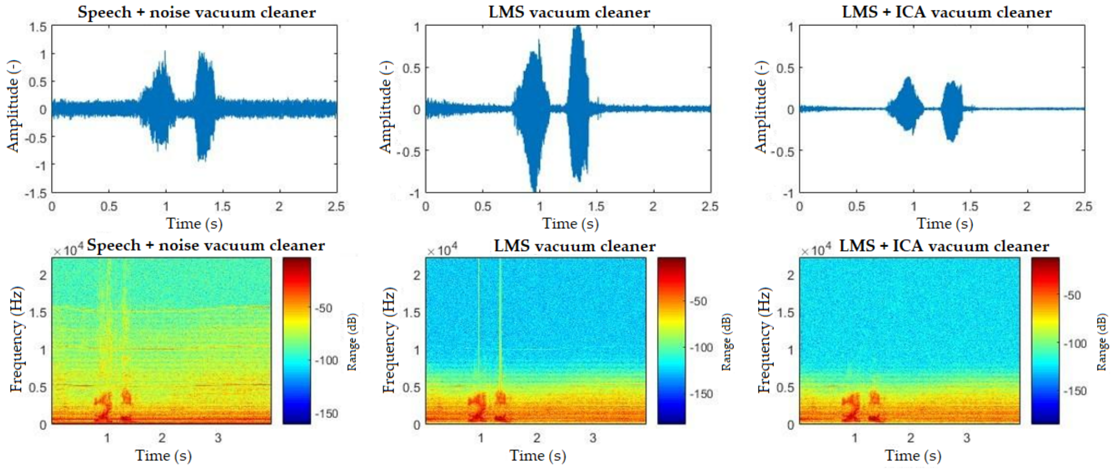

Figure 11. Comparison of spectrograms with closed windows, "light on" command, vacuum cleaner interference.

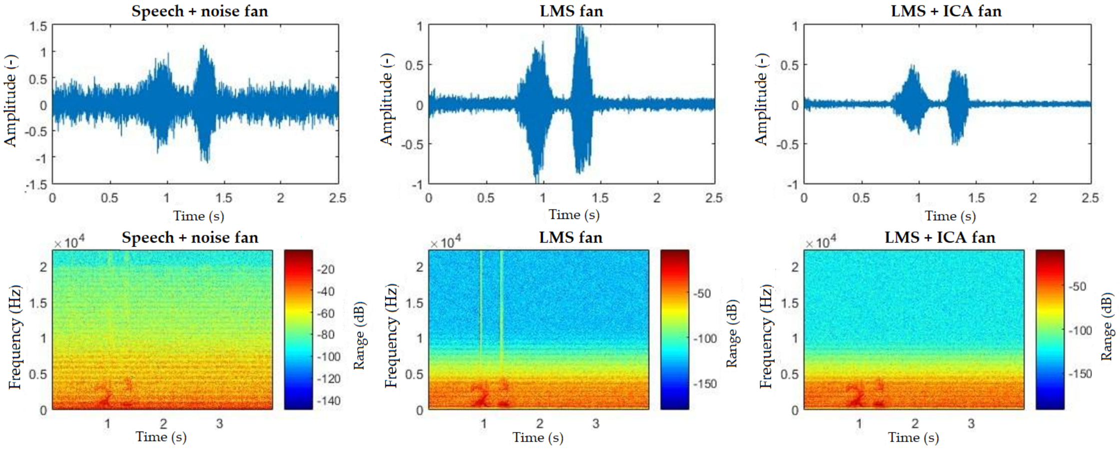

Figure 12. Comparison of spectrograms with closed windows, "light on" command, fan interference.

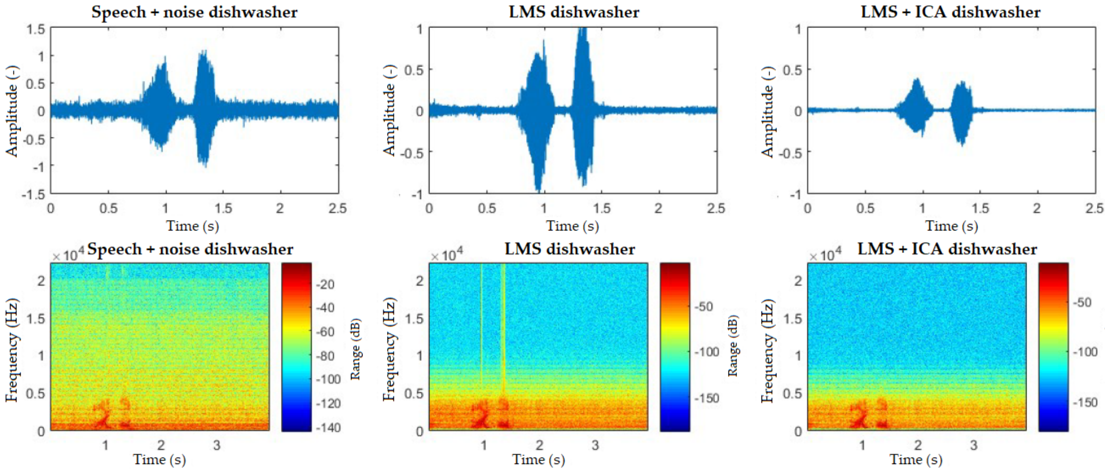

Figure 13. Comparison of spectrograms with closed windows, "light on" command, dishwasher interference.

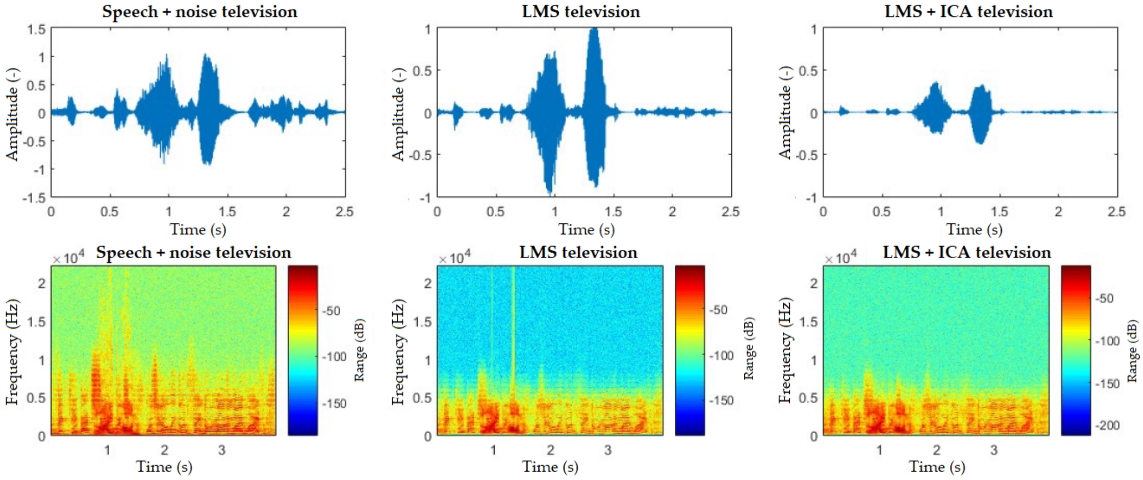

Figure 14. Comparison of spectrograms with closed windows, "light on" command, television interference.

9. Conclusions

This study was focused on an innovative method of processing speech signals used for voice control of operational and technical functions in Smart Home, with subsequent testing of selected filtering methods. To control the operational and technical functions (blinds, lights, heating, cooling, and forced ventilation) in the SH rooms (living room, kitchen, dining room, and bedroom), a program for controlling the KNX technology was created using the ETS 5 software tool. A Microsoft recognizer was used to recognize the individual voice commands. To ensure visualization and data connectivity among the building automation technology, the sound card, and the SW tool for speech recognition, a LabView SW tool was used in this work together with a database of additive interference recordings in a real SH environment (television, vacuum cleaner, washing machine, dishwasher, and fan). A linear adaptive LMS filter and the ICA method were chosen to filter speech signals that contained additive noise from the real SH environment. The criterion for successful recognition was represented by a sequence of one hundred repetitions for each command based on which the recognized/unrecognized state was evaluated. During testing, commands for five types of interference were tested. The results show that the hybrid method showed a higher recognition success rate than the LMS algorithm, on average by 6%. The average recognition success rate before and after filtering was 64.2% higher for the LMS algorithm and 69.8% for hybrid filtering. The overall results reveal that hybrid filtration showed a higher success rate by only about 5%. Due to the computational complexity of the ICA method, it is much more advantageous to implement the LMS algorithm, which is capable of high levels of filtering despite its simplicity, but, with the increasing performance and quality of computer technology, there is room for more complex algorithms to address large tasks at relatively low cost.

In the next work, the authors will focus on optimizing the control of the operational and technical functions in SH and increasing the recognition success rate of the individual speech commands using appropriate speech recognition algorithms and appropriate algorithms for additive noise canceling in real time.

Author Contributions: R.M. proposed the idea and edited the manuscript. J.V., J.N., M.F., J.F. and A.K.-S. developed, tested, and validated data. J.V. and M.F. wrote the manuscript. R.M. performed statistical investigation and visualization. R.M., J.V., J.N., M.F. and J.N. critically evaluated the quality of the research data and experimental methods used to generate them as well as the soundness and validity of the scientific and engineering techniques and performed its final edits. All authors have read and agreed to the published version of the manuscript.

Funding: This article was supported by the Ministry of Education of the Czech Republic (Project No. SP2020/156). This work was supported by the European Regional Development Fund in Research Platform focused on Industry 4.0 and Robotics in Ostrava project CZ.02.1.01/0.0/0.0/17_049/0008425 within the Operational Programme Research, Development and Education.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dotihal, R.; Sopori, A.; Muku, A.; Deochake, N.; Varpe, D. *Smart Homes Using Alexa and Power Line Communication in IoT: ICCNCT 2018*; Springer Nature: Basel, Switzerland, 2019; pp. 241–248. [CrossRef]
2. Erol, B.A.; Wallace, C.; Benavidez, P.; Jamshidi, M. Voice Activation and Control to Improve Human Robot Interactions with IoT Perspectives. In Proceedings of the 2018 World Automation Congress (WAC), Stevenson, WA, USA, 3–6 June 2018; pp. 1–5.
3. Diaz, A.; Mahu, R.; Novoa, J.; Wuth, J.; Datta, J.; Yoma, N.B. Assessing the effect of visual servoing on the performance of linear microphone arrays in moving human-robot interaction scenarios. *Comput. Speech Languag.* **2021**, *65*, 101136. [CrossRef]
4. Novoa, J.; Wuth, J.; Escudero, J.P.; Fredes, J.; Mahu, R.; Yoma, N.B. DNN-HMM based Automatic Speech Recognition for HRI Scenarios. In Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, Chicago, IL, USA, 5–8 March 2018. [CrossRef]
5. Grout, I. Human-Computer Interaction in Remote Laboratories with the Leap Motion Controller. In Proceedings of the 15th International Conference on Remote Engineering and Virtual Instrumentation, Duesseldorf, Germany, 21–23 March 2019; pp. 405–414. [CrossRef]
6. He, S.; Zhang, A.; Yan, M. Voice and Motion-Based Control System: Proof-of-Concept Implementation on Robotics via Internet-of-Things Technologies. In Proceedings of the 2019 ACM Southeast Conference, Kennesaw, GA, USA, 18–20 April 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 102–108. [CrossRef]
7. Kennedy, S.; Li, H.; Wang, C.; Liu, H.; Wang, B.; Sun, W. *I Can Hear Your Alexa: Voice Command Fingerprinting on Smart Home Speakers*; IEEE: Washington, DC, USA, 2019; pp. 232–240. [CrossRef]
8. Knight, N.J.; Kanza, S.; Cruickshank, D.; Brocklesby, W.S.; Frey, J.G. Talk2Lab: The Smart Lab of the Future. *IEEE Int. Things J.* **2020**, *7*, 8631–8640. [CrossRef]
9. Kodali, R.K.; Azman, M.; Panicker, J.G. Smart Control System Solution for Smart Cities. In Proceedings of the 2018 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Zhengzhou, China, 18–20 October 2018, pp. 89–893.
10. Leroy, D.; Coucke, A.; Lavril, T.; Gisselbrecht, T.; Dureau, J. Federated Learning for Keyword Spotting. In Proceedings of the ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 6341–6345.

11. Li, Z.; Zhang, J.; Li, M.; Huang, J.; Wang, X. A Review of Smart Design Based on Interactive Experience in Building Systems. *Sustainability* **2020**, *12*, 6760. [CrossRef]
12. Irwin, S. Design and Implementation of Smart Home Voice Control System based on Arduino. In Proceedings of the 2018 5th International Conference on Electrical & Electronics Engineering and Computer Science (ICEEECS 2018), Istanbul, Turkey, 3–5 May 2018. [CrossRef]
13. Vanus, J.; Belesova, J.; Martinek, R.; Nedoma, J.; Fajkus, M.; Bilik, P.; Zidek, J. Monitoring of the daily living activities in smart home care. *Hum. Centric Comput. Inf. Sci.* **2017**, *7*. [CrossRef]
14. Vanus, J.; Weiper, T.; Martinek, R.; Nedoma, J.; Fajkus, M.; Koval, L.; Hrbac, R. Assessment of the Quality of Speech Signal Processing Within Voice Control of Operational-Technical Functions in the Smart Home by Means of the PESQ Algorithm. *IFAC-PapersOnLine* **2018**, *51*, 202–207. [CrossRef]
15. Amrutha, S.; Aravind, S.; Ansu, M.; Swathy, S.; Rajasree, R.; Priyalakshmi, S. Voice Controlled Smart Home. *Int. J. Emerg. Technol. Adv. Eng. (IJETAE)* **2015**, 272–276.
16. Kamdar, H.; Karkera, R.; Khanna, A.; Kulkarni, P.; Agrawal, S. A Review on Home Automation Using Voice Recognition. *Int. J. Emerg. Technol. Adv. Eng. (IJETAE)* **2017**, *4*, 1795–1799.
17. Kango, R.; Moore, P.R.; Pu, J. Networked smart home appliances-enabling real ubiquitous culture. In Proceedings of the 3rd IEEE International Workshop on System-on-Chip for Real-Time Applications, Calgary, Alberta, 30 June–2 July 2003; pp. 76–80.
18. Yi-Min Wang.; Russell, W.; Arora, A.; Jun Xu.; Jagannatthan, R.K. Towards dependable home networking: an experience report. In Proceedings of the International Conference on Dependable Systems and Networks (DSN 2000), New York, NY, USA, 25–28 June 2000; pp. 43–48.
19. McLoughlin, I.; Sharifzadeh, H.R. Speech Recognition for Smart Homes. In *Speech Recognition*; Mihelic, F., Zibert, J., Eds.; IntechOpen: Rijeka, Croatia, 2008; Chapter 27. [CrossRef]
20. Rabiner, L.R. Applications of voice processing to telecommunications. *Proc. IEEE* **1994**, *82*, 199–228. [CrossRef]
21. Obaid, T.; Rashed, H.; Nour, A.; Rehan, M.; Hasan, M.; Tarique, M. Zigbee Based Voice Controlled Wireless Smart Home System. *Int. J. Wirel. Mob. Netw.* **2014**, *6*. [CrossRef]
22. Singh, D.; Sharma Thakur, A. Voice Recognition Wireless Home Automation System Based On Zigbee. *IOSR J. Electron. Commun. Eng.* **2013**, *22*, 65–75. [CrossRef]
23. Mctear, M. *Spoken Dialogue Technology—Toward the Conversational User Interface*; Springer Publications: Berlin/Heidelberg, Germany, 2004.
24. Chevalier, H.; Ingold, C.; Kunz, C.; Moore, C.; Roven, C.; Yamron, J.; Baker, B.; Bamberg, P.; Bridle, S.; Bruce, T.; et al. Large-vocabulary speech recognition in specialized domains. In Proceedings of the 1995 International Conference on Acoustics, Speech, and Signal Processing, Detroit, MI, USA, 9–12 May 1995; Volume 1, pp. 217–220.
25. Kamm, C.A.; Yang, K.; Shamieh, C.R.; Singhal, S. Speech recognition issues for directory assistance applications. In Proceedings of the 2nd IEEE Workshop on Interactive Voice Technology for Telecommunications Applications, Kyoto, Japan, 26–27 September 1994; pp. 15–19.
26. Sun, H.; Shue, L.; Chen, J. Investigations into the relationship between measurable speech quality and speech recognition rate for telephony speech. In Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, QC, Canada, 17–21 May 2004; Volume 1, pp. 1–865.
27. Ravishankar, M.K. *Efficient Algorithms for Speech Recognition*; Technical Report; Carnegie-Mellon Univ Pittsburgh pa Dept of Computer Science: Pittsburgh, Pennsylvania, 1996.
28. Vajpai, J.; Bora, A. Industrial applications of automatic speech recognition systems. *Int. J. Eng. Res. Appl.* **2016**, *6*, 88–95.
29. Rogowski, A. Industrially oriented voice control system. *Robot. Comput. Integr. Manuf.* **2012**, *28*, 303–315. [CrossRef]
30. Collins, D.W.B.R. *Digital Avionics Handbook—Chapter 8: Speech Recognitionand Synthesis*; Electrical Engineering Handbook Series; CRC Press: Boca Raton, FL, USA, 2001.
31. Rabiner, L.R. Applications of speech recognition in the area of telecommunications. In Proceedings of the 1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, Santa Barbara, CA, USA, 17 December 1997; pp. 501–510.
32. MAŘÍK, V. *Průmysl 4.0: výZva Pro Českou Republiku*, 1st ed.; Management Press: Prague, Czech Republic, 2016.

33. Newsroom, C. Cyber-Physical Systems [online]. Available online: http://cyberphysicalsystems.org/ (accessed on 25 August 2020).
34. Mardiana, B.; Hazura, H.; Fauziyah, S.; Zahariah, M.; Hanim, A.R.; Noor Shahida, M.K. Homes Appliances Controlled Using Speech Recognition in Wireless Network Environment. In Proceedings of the 2009 International Conference on Computer Technology and Development, Kota Kinabalu, Malaysia, 13–15 November 2009; Volume 2, pp. 285–288.
35. Techopedia. Smart Device Techopedia [online]. Available online: https://www.techopedia.com/definition/31463/smart-device (accessed on 25 August 2020).
36. Schiefer, M. *Smart Home Definition and Security Threats*; IEEE: Magdeburg, Germany, 2015; pp. 114–118. [CrossRef]
37. Kyas, O. *How To Smart Home*; Key Concept Press e.K.: Wyk, Germany, 2013.
38. Psutka, J.; Müller, L.; Matoušek, J.; Radová, V. *Mluvíme s Počítačem česky*; Academia: Prague, Czech Republic, 2006; p. 752.
39. Sakoe, H.; Chiba, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoust. Speech, Signal Process.* **1978**, *26*, 43–49. [CrossRef]
40. Bellman, R.E.; Dreyfus, S.E. *Applied Dynamic Programming*; Princeton University Press: Princeton, NJ, USA, 2015.
41. Kumar, A.; Dua, M.; Choudhary, T. Continuous hindi speech recognition using monophone based acoustic modeling. *Int. J. Comput. Appl.* **2014**, *24*, 15–19.
42. Arora, S.J.; Singh, R.P. Automatic speech recognition: A review. *Int. J. Comput. Appl.* **2012**, *60*, 132–136.
43. Saksamudre, S.K.; Shrishrimal, P.; Deshmukh, R. A review on different approaches for speech recognition system. *Int. J. Comput. Appl.* **2015**, *115*, 23–28.
44. Hermansky, H. Perceptual linear predictive (PLP) analysis of speech. *J. Acoust. Soc. Am.* **1990**, *87*, 1738–1752. [CrossRef]
45. Xie, L.; Liu, Z. A Comparative Study of Audio Features for Audio-to-Visual Conversion in Mpeg-4 Compliant Facial Animation. In Proceedings of the 2006 International Conference on Machine Learning and Cybernetics, Dalian, China, 13–16 August 2006; pp. 4359–4364.
46. Garg, A.; Sharma, P. Survey on acoustic modeling and feature extraction for speech recognition. In Proceedings of the 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 16–18 March 2016; pp. 2291–2295.
47. Rajnoha, J.; Pollák, P. Detektory řečové aktivity na bázi perceptivní kepstrální analýzy. In *České Vysoké učení Technické v Praze, Fakulta Elektrotechnická*; Fakulta Elektrotechnická: Prague, Czech Republic, 2008.
48. Saon, G.A.; Soltau, H. Method and System for Joint Training of Hybrid Neural Networks for Acoustic Modeling in Automatic Speech Recognition. U.S. Patent 9,665,823, 30 May 2017.
49. Godino-Llorente, J.I.; Gomez-Vilda, P.; Blanco-Velasco, M. Dimensionality Reduction of a Pathological Voice Quality Assessment System Based on Gaussian Mixture Models and Short-Term Cepstral Parameters. *IEEE Trans. Biomed. Eng.* **2006**, *53*, 1943–1953. [CrossRef]
50. Low, L.S.A.; Maddage, N.C.; Lech, M.; Sheeber, L.; Allen, N. Content based clinical depression detection in adolescents. In Proceedings of the 2009 17th European Signal Processing Conference, Glasgow, UK, 24–28 August 2009; pp. 2362–2366.
51. Linh, L.H.; Hai, N.T.; Thuyen, N.V.; Mai, T.T.; Toi, V.V. MFCC-DTW algorithm for speech recognition in an intelligent wheelchair. In Proceedings of the 5th international conference on biomedical engineering in Vietnam, Ho Chi Minh City, Vietnam, 16–18 June 2014; pp. 417–421.
52. Ittichaichareon, C.; Suksri, S.; Yingthawornsuk, T. Speech recognition using MFCC. In Proceedings of the International Conference on Computer Graphics, Simulation and Modeling, Pattaya, Thailand, 28–29 July 2012; pp. 135–138.
53. Vařák, J. Možnosti hlasového ovládání bezpilotních dronů. In *Bakalářská Práce*; Vysoká škola Báňská—Technická Univerzita Ostrava: Ostrava, Czech Republic, 2017.
54. Cutajar, M.; Gatt, E.; Grech, I.; Casha, O.; Micallef, J. Comparative study of automatic speech recognition techniques. *IET Signal Process.* **2013**, *7*, 25–46. [CrossRef]
55. Gevaert, W.; Tsenov, G.; Mladenov, V. Neural networks used for speech recognition. *J. Autom. Control.* **2010**, *20*, 1–7. [CrossRef]

56. Jamal, N.; Shanta, S.; Mahmud, F.; Sha'abani, M. Automatic speech recognition (ASR) based approach for speech therapy of aphasic patients: A review. *Aip Conf. Proc.* **2017**, *1883*, 020028.
57. Xue, S.; Abdel-Hamid, O.; Jiang, H.; Dai, L.; Liu, Q. Fast adaptation of deep neural network based on discriminant codes for speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 1713–1725.
58. Romdhani, S. Implementation of Dnn-Hmm Acoustic Models for Phoneme Recognition. Ph.D. Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
59. Pujol, P.; Pol, S.; Nadeu, C.; Hagen, A.; Bourlard, H. Comparison and combination of features in a hybrid HMM/MLP and a HMM/GMM speech recognition system. *IEEE Trans. Speech Audio Process.* **2005**, *13*, 14–22. [CrossRef]
60. Zarrouk, E.; Ayed, Y.B.; Gargouri, F. Hybrid continuous speech recognition systems by HMM, MLP and SVM: a comparative study. *Int. J. Speech Technol.* **2014**, *17*, 223–233. [CrossRef]
61. Chaudhari, A.; Dhonde, S.B. A review on speech enhancement techniques. In Proceedings of the 2015 International Conference on Pervasive Computing (ICPC), Pune, India, 8–10 January 2015 , pp. 1–3.
62. Upadhyay, N.; Karmakar, A. Speech enhancement using spectral subtraction-type algorithms: A comparison and simulation study. *Procedia Comput. Sci.* **2015**, *54*, 574–584. [CrossRef]
63. MARTINEK, R. The Use of Complex Adaptive Methods of Signal Processingfor Refining the Diagnostic Quality of the Abdominalu Fetal Cardiogram. Ph.D. Thesis, Vysoká škola báňská—Technická Univerzita Ostrava, Ostrava, Czech Republic, 2014.
64. Jan, J. *Číslicová Filtrace, Analýza a Restaurace Signálů*; vyd. 2. rozš. a dopl ed.; VUTIUM: Brno, Czech Republic, 2002.
65. Harding, P. Model-Based Speech Enhancement. Ph.D. Thesis, University of East Anglia, Norwich, UK, 2013.
66. Loizou, P.C. *Speech Enhancement: Theory and Practice*; CRC Press: Boca Raton, FL, USA, 2013.
67. Cole, C.; Karam, M.; Aglan, H. Increasing Additive Noise Removal in Speech Processing Using Spectral Subtraction. In Proceedings of the Fifth International Conference on Information Technology: New Generations (itng 2008), Las Vegas, NV, USA, 7–9 April 2008; pp. 1146–1147. [CrossRef]
68. Aggarwal, R.; Singh, J.K.; Gupta, V.K.; Rathore, S.; Tiwari, M.; Khare, A. Noise reduction of speech signal using wavelet transform with modified universal threshold. *Int. J. Comput. Appl.* **2011**, *20*, 14–19. [CrossRef]
69. Mihov, S.G.; Ivanov, R.M.; Popov, A.N. Denoising speech signals by wavelet transform. *Annu. J. Electron.* **2009**, *2009*, 2–5.
70. Martinek, R. Využití Adaptivních Algoritmů LMS a RLS v Oblasti Adaptivního Potlačování Šumu a Rušení. *ElectroScope* **2013**, *1*, 1–8.
71. Farhang-Boroujeny, B. *Adaptive Filters: Theory and Applications*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2013.
72. Vaseghi, S.V. *Advanced Digital Signal Processing and Noise Reduction*, 3rd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2008.
73. Martinek, R.; Žídek, J. The Real Implementation of NLMS Channel Equalizer into the System of Software Defined Radio. *Adv. Electr. Electron. Eng.* **2012**, *10*. [CrossRef]
74. Visser, E.; Otsuka, M.; Lee, T.W. A spatio-temporal speech enhancement scheme for robust speech recognition in noisy environments. *Speech Commun.* **2003**, *41*, 393–407. [CrossRef]
75. Visser, E.; Lee, T.W. Speech enhancement using blind source separation and two-channel energy based speaker detection. In Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03), Hong Kong, China, 6–10 April 2003; Volume 1, p. I.
76. Hyvarinen, A.; Oja, E. A fast fixed-point algorithm for independent component analysis. *Neural Comput.* **1997**, *9*, 1483–1492. [CrossRef]
77. Cichocki, A.; Amari, S.I. *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2002.
78. Fischer, S.; Simmer, K.U. Beamforming microphone arrays for speech acquisition in noisy environments. *Speech Commun.* **1996**, *20*, 215–227. [CrossRef]
79. Griffiths, L.; Jim, C. An alternative approach to linearly constrained adaptive beamforming. *IEEE Trans. Antennas Propagation* **1982**, *30*, 27–34. [CrossRef]
80. Zou, Q.; Yu, Z.L.; Lin, Z. A robust algorithm for linearly constrained adaptive beamforming. *IEEE Signal Process. Lett.* **2004**, *11*, 26–29. [CrossRef]

81. Vaňuš, J.; Smolon, M.; Martinek, R.; Koziorek, J.; Žídek, J.; Bilík, P. Testing of the voice communication in smart home care. *Hum. Centric Comput. Inf. Sci.* **2015**, *5*, 15.
82. Wittassek, T. *Virtuální Instrumentace I*, 1. vyd ed.; Vysoká škola Báňská—Technická Univerzita Ostrava: Ostrava, Czech Republic, 2014.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.