Technologia i Automatyzacja Montażu

Volume 126, Issue 4/2024, Pages 44-50
https://doi.org/10.7862/tiam.2024.4.6

Original Research

VOICE CONTROL SYSTEM FOR COLLABORATIVE ROBOT USED IN ASSEMBLY PROCESS

SYSTEM STEROWANIA GŁOSEM DLA ROBOTA KOLABORACYJNEGO WYKORZYSTYWANEGO DO PROCESU MONTAŻU

Jan CZYŻEWSKI [1,*]

[1] Rzeszów University of Technology, Aleja Powstańców Warszawy 12, 35-959 Rzeszów, Poland

* Corresponding author: j.czyzewski@prz.edu.pl

Abstract

This work is devoted to the issue of recognizing voice commands by the voice control system of a collaborative robot in an industrial environment. Such a robot will be designed to cooperate with a human during the assembly process. This method of communication significantly facilitates and speeds up work, increasing the efficiency of the assembly process and reducing the operator's workload. However, production halls and other places where assembly takes place are environments filled with sounds with a wide spectrum of intensity and frequency. Such an environment significantly impedes the functioning of the voice control system. This work examined the impact of sound pollution on the robot's command recognition.

Keywords: voice control, collaborative robot, command recognition

Streszczenie

Niniejsza praca poświęcona jest zagadnieniu rozpoznawania komend głosowych przez system głosowego sterowania robotem kolaboracyjnym w środowisku przemysłowym. Robot taki, przeznaczony do współpracy z człowiekiem przy procesie montażu, będzie sterowany za pomocą głosu. Ten sposób komunikacji w znacznym stopniu ułatwia i przyspiesza pracę, pozwalając na zwiększenie wydajności procesu montażu i zmniejszenie obciążenia operatora. Hale produkcyjne i inne miejsca, w których odbywa się montaż są jednak środowiskiem wypełnionym dźwiękami o szerokim spektrum natężenia i częstotliwości. Środowisko takie utrudnia w znacznym stopniu funkcjonowanie systemu sterowania głosem. W pracy tej zbadano wpływ zanieczyszczenia otoczenia dźwiękiem na rozpoznawanie komend przez robota.

Słowa kluczowe: sterowanie głosem, robot kolaboracyjny, rozpoznawanie komend

1. Introduction

Basically, since the creation of the industry, there has been constant effort to improve the production process and its very important part, i.e. the assembly process. One of the trends is the automation and robotization of these processes. Entrusting the assembly process to machines allows to relieve the employee, especially when performing burdensome or monotonous activities. On the other hand, it also ensures high repeatability of the product quality. Naturally, this allows to increase the efficiency of the process, reduce the workload on the employee and improve his working conditions. Nowadays, robots play an increasingly important role in production plants, which is related to the improvement of their design and the growing range of applications. It also turns out that the availability of specialized assembly specialists on the labor market can be problematic, which is why efforts are being made to support already employed, qualified and experienced employees with robots. This way, one person, with less physical burden, can supervise the work of many robots. Human presence is usually indispensable, and this approach produces beneficial results. However, for proper cooperation between humans and robots, it is necessary to provide them with appropriate tools for mutual communication. A very convenient system is voice communication, where the operator issues verbal commands that are recognized and executed by the system controlling the operation of robots or other devices. Such systems have a number of advantages. First of all, they use speech, which is a natural human communication tool. Secondly, they allow to bypass switches, switchboards and classic interfaces used on machines. The interfaces themselves are often complicated, have multi-level menus and require long training and getting used to them by the operator. They often differ significantly depending on the model of a specific machine, its manufacturer, etc. It is not always possible to operate them remotely. Voice control can solve these problems. A properly designed voice control system can allow the operator to conveniently find and select the necessary option and issue a command to perform specific activities. This significantly reduces the operator's workload and simplifies his training. Additionally, it can be done remotely. One of the basic challenges when creating such a system is to ensure its effectiveness in the conditions of a production plant. On the one hand, in such a place we are dealing with many sound sources, with a very wide spectrum of intensity and frequency. On the other hand, command recognition must work efficiently and error-free. Both the failure of the system to respond to the issued command and the incorrect interpretation of the issued command may lead to a dangerous situation, threatening both the operator and the machines and manufactured products. Therefore, it is extremely important to thoroughly check the voice recognition system for sensitivity to factors disturbing its operation. This work is devoted to this issue.

Many research works are devoted to the issue of voice control. Nzuva (2021) writes generally about creating an easy way of communication between humans and the machines they operate. Norda et al. (2024) evaluate the usefulness of voice control in industry and its impact on improving work efficiency. Longo et al. (2020) also write about the need to modernize methods of collaboration between humans and devices in the environment of a modern factory. Janíček et al. (2021a), Chmielowiec et al. (2024) and Bingol et al. (2020) work on the problem of controlling robots using voice in industrial applications. Anggraen et al. (2018), Priyadarshana et al. (2022) and Rendyansyah et al. (2022) also focus on controlling robots using voice. Andrew et al. (2021) additionally focus on reducing the costs of such a system. Lavrynenko (2024) deals with the problem of ensuring the proper functioning of a speech recognition system in a noisy environment. Similarly, Janíček et al. (2021b) examine the problem of the influence of sound intensity on the operation of the system. Voice is considered a good form of communication with universal robots designed to support humans in everyday duties. What is important here is the simplicity of this form of communication, which does not require additional qualifications. Li et al. (2023) and Linda et al. (2020) writes about it. Khan et al. (2021) write about the application of a voice control system in a robotic car. Also, great potential for voice control has been noticed in the case of robots and other devices intended for disabled people. Prostheses replacing limbs can also be voice-controlled - this is described by Oyelami et al. (2023). Piyaneeranart et al. (2022) propose the use of voice control in a robot supporting older people, and Bakouri et al. (2022) in a robotic wheelchair.

2. Materials and methods

A simple scheme for issuing commands to the robot was adopted. The commands are supposed to control the robot's movements (fig. 1). For testing purposes, it was assumed that simple movements along the coordinate system axes were performed by given values, expressed in millimeters. For now, more complex movements, changes in movement speed, or movement values after the decimal point were omitted.

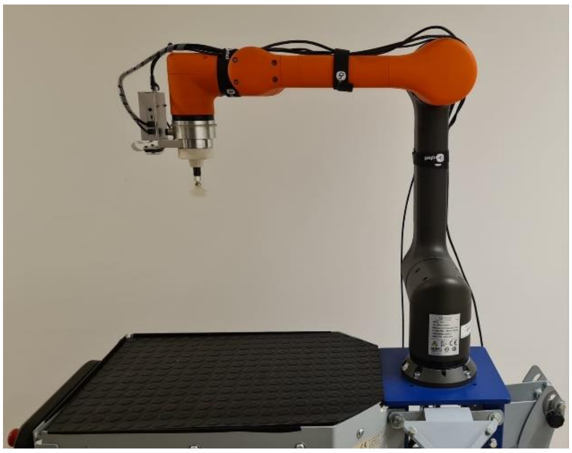

Fig. 1. Hanwha collaborative robot, planned for system testing

Commands consist of single words, representing letters and numbers. These words were taken from the ICAO (International Civil Aviation Organization) phonetic alphabet, also known as the NATO phonetic alphabet (tab. 1 and 2). The main advantage of this coding is its simplicity, unambiguity and resistance to interference. Since the arrangement of syllables that make up individual words of the code is not repeated, even a fragmentary transmission can be understood. Therefore, it is widely used in radio communication, especially in aviation, where unambiguity and ease of understanding of messages are of key importance. A complete command contains the coordinate in which the robot is to move (X, Y or Z), a + or – sign specifying the direction, and a sequence of numbers specifying the displacement value in mm.

Table 1. Letters in ICAO code

Letter	Code	Pronunciation
A	**Alfa**	[ˈalfa] ***al**fa*
B	**Bravo**	[ˈbravo] ***bra**wo*
C	**Charlie**	[ˈtʃali] ***czar**li*
D	**Delta**	[ˈdɛlta] ***del**ta*
E	**Echo**	[ˈɛko] ***e**ko*
F	**Foxtrot**	[ˈfɔks.trɔt] ***foks**trot*
G	**Golf**	[ˈgɔlf] ***golf***
H	**Hotel**	[hoˈtɛl] *ho**tel***
I	**India**	[ˈɪndia] ***in**dija*
J	**Juliett**	[ˈdʒuliˈɛt] ***dżu**li**jet***
K	**Kilo**	[ˈkilo] ***ki**lo*
L	**Lima**	[ˈlima] ***li**ma*
M	**Mike**	[ˈmai̯k] ***majk***
N	**November**	[noˈvɛmba] *no**wem**ber*
O	**Oscar**	[ˈɔska] ***os**kar*
P	**Papa**	[paˈpa] *pa**pa***
Q	**Quebec**	[keˈbɛk] *ke**bek***
R	**Romeo**	[ˈromio] ***ro**mijo*
S	**Sierra**	[si'ɛra] *si**je**ra*
T	**Tango**	[ˈtaŋgo] ***tan**go*
U	**Uniform**	[ˈjunifɔrm] ***ju**niform* lub [ˈunifɔm] ***u**niform*
V	**Victor**	[ˈvɪktor] ***wik**tor*
W	**Whiskey**	[ˈwɪski] ***łys**ki*
X	**Xray**	[ˈɛks.rei̯] ***eks**rej*
Y	**Yankee**	[ˈjaŋki] ***jan**ki*
Z	**Zulu**	[ˈzulu] ***zu**lu*

Table 2. Numbers in ICAO code

Number	Code	Pronunciation
0	Zero	[ˈziro] ***zi**ro*
1	One	[ˈwan] ***łan***
2	Two	[ˈtu] ***tu***
3	Three	[ˈtri] ***tri***
4	Four	[ˈfoa] ***fo**wer*
5	Five	[ˈfai̯f] ***fajf***
6	Six	[ˈsɪks] ***siks***
7	Seven	[ˈsɛvən] ***sev**en*
8	Eight	[ˈei̯t] ***ejt***
9	Niner	[ˈnai̯ner] ***Nin**er*

Sound analysis was created in Python using the open source librosa library. Librosa is used for music and sound analysis, helping software developers create applications for working with audio and music file formats using Python. This library can handle both basic and advanced tasks related to sound and music processing.

After separating the speech signal from the surrounding silence, its parameterization is necessary, because it is characterized by a large amount of information redundancy. For this purpose, the method of cepstral coefficients in the mel frequency scale (mel-Frequency Cepstral Coefficients) can be used. As Furui (2009) writes, MFCC is the longest-developed method of human speech parameterization. The signal spectrum obtained by the fast Fourier transform is filtered by a bank of filters covering the entire frequency band of the signal. Mel-cepstral parameters are created from the cepstrum of the signal presented in the mel scale (melcepstrum). The set of filters imitates the characteristics of the human hearing system. The mel scale is obtained by filtering the signal with a bank of filters with a triangular characteristic. The k-th mel-cepstral coefficient corresponds to the content of the k-th band.

The mel-cepstral parameter vector is a vector of cepstrum coefficients in the appropriate mel bands. They are intended to reflect the natural response of the auditory system to stimulation by speech sounds. The mel-cepstral parameters are characterized by low sensitivity to noise.

At the beginning of the algorithm, the speech signal is subjected to a pre-emphasis process, i.e. formative filtering, aimed at weakening low-frequency components and amplifying high-frequency components.

$$x'_n = x_n - a x_{n-1}, \quad (1)$$

where x_n and x'_n are the signal before and after pre-emphasis, respectively, and a is the coefficient (the most commonly assumed value is 0.97).

The next stage is signal framing, i.e. dividing the signal into short fragments called frames. It is possible to use overlap of subsequent time frames. Then windowing is performed using a Hamming window.

$$\mathrm{Ham}(N) = 0{,}54 - 0.46\cos\left(2\pi\frac{n-1}{N-1}\right) \quad (2)$$

where N is the frame lenght, and $n = 1, 2, \ldots, N$. Each frame is subjected to the FFT to obtain the power spectrum $|FFT|^2$ | or the amplitude spectrum FFT| depending on the algorithm implementation. The obtained spectrum is subjected to the action of a set of filters. The number of filters and their shape differs depending on the algorithm implementation, but the most common in the literature are triangular filters, whose centers are evenly distributed in a given frequency range on the mel scale. An example set of filters is shown in the figure. At the output of each filter, the band energy is obtained according to the formula: :

$$S_m = \sum_{k=1}^{N} |X_r(k)|^2 H_m(k) \quad (3)$$

where m is the filter number, and X_r is the frame spectrum.

In further calculations, the logarithm of energy is used, which allows to reduce the sensitivity of filters to very loud and very quiet sounds and to model the nonlinear amplitude sensitivity of the human ear. Moreover, the use of the logarithm significantly affects the quality of recognition. The last stage of the algorithm is to apply the discrete cosine transform (DCT). The final values of the MFCC coefficients are calculated as:

$$c_i = \sqrt{\frac{2}{M}} \sum_{m=1}^{M} \log(S_m)\cos\left(\frac{\pi \cdot i}{M}(m - 0{,}5)\right) \quad (4)$$

where i is the coefficient number and M is the number of filters used. In most recognition systems, i takes values from 1, and the coefficient c_0 is omitted. Often, the logarithm of the frame energy is added to the obtained coefficients:

$$E = \log\left(\sum_{k=1}^{N} x_k^2\right) \quad (5)$$

It is beneficial for the machine learning process to prepare a sufficiently large number of samples. 100 sound samples were recorded for each letter and digit.

The samples are then subjected to a normalization process.

Audio normalization is the process of applying a constant amount of sound gain to an audio recording. So it is the process of changing the amplitude value of a signal in a consistent way (for all values). Gain would then be the amount by which the sound is changed. This can be a positive or negative value. There are two types of normalization, these are: peak normalization and average normalization.

Average normalization determines the average level of an audio file and similarly raises or lowers it to a target level.

In this case, the peak normalization process is used. This process finds the highest PCM (pulse-code modulation) value or pulse-code modulation value of an audio file. Basically, the audio signal is normalized so that it is related to the loudest point recorded in the audio waveform. The calculation involves subtracting the individual values from the measured peak value. Audio normalization solves the problem of varying loudness levels in a sequence of recordings (fig. 2).

Then, in each of the recorded samples, the silence area was identified and cut out. Silence was defined as sound of intensity not exceeding 30 dB and higher volume pulses whose duration does not exceed 0.5 s (fig. 3).

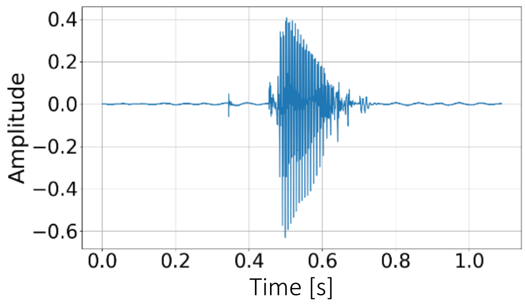

Fig. 2. Single word recording

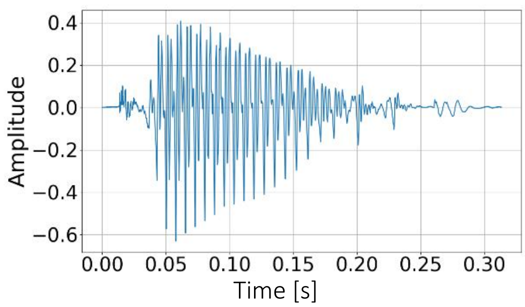

Fig. 3. Single word recording after cutting out the silence

The samples were then subjected to a process called resampling, or changing the sampling rate. The sampling rate is a measure of the number of samples recorded per second from an audio signal. It is measured in hertz (Hz) and indicates how often a sample of the audio signal is recorded per second. A single sample is a measure of the volume of the audio signal at a specific moment in time. The original frequency of 44,100 Hz is reduced to 16,000 Hz at this stage.

A series of typical MFCC statistics were then determined.

Two types of classifier were used:

- K-NN – k nearest neighbors classifier. Nearest neighbor analysis is a method of classifying observations based on their similarity to other observations. The method was developed in machine learning as a way to recognize patterns in data without having to ensure exact correspondence to any memorized patterns or observations. Similar observations are close together, and dissimilar ones are far away. Therefore, the distance between two observations is a measure of their dissimilarity. Observations that are close together are called "neighbors." When a new observation is presented, the distance from each observation to the model is calculated. The classification of the most similar nearest neighbor observations is determined, and the new observation is placed in the category that contains the largest number of nearest neighbor observations.
- SVM (Support Vector Machines). The SVM algorithm works by mapping data into a multidimensional feature space in a way that allows data points to be categorized, even if the data cannot otherwise be linearly separated. First, a separator between categories is found. Then, the data is transformed in a way that allows the separator to be drawn as a hyperplane. Once this is done, the characteristics of the new data can be used to predict the group to which a new record should belong.

Sample recordings are shown in the figures (fig. 4 and 5). The recording in conditions of high ambient noise emission is visible in Figure 6.

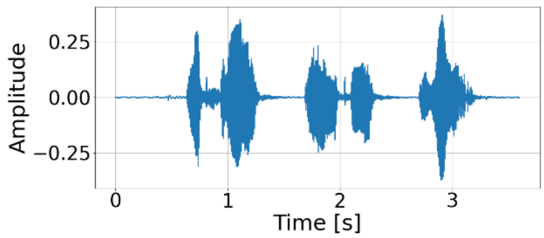

Fig. 4. Example recording of actions in 3 words

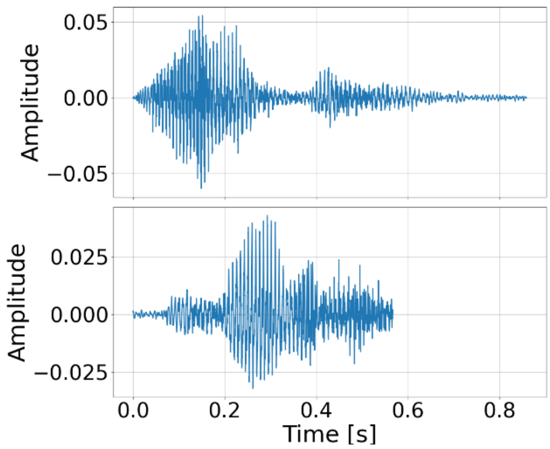

Fig. 5. Division into single commands

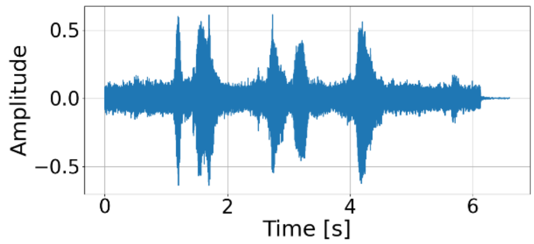

Fig. 6. Example recording of 3-word actions in a noisy environment

The following sets of classifiers and statistics were used in the study:

1) nearest neighbor classifier for 5 neighbors, MFCC statistics vector as the basis for classification (KN - S),

2) nearest neighbor classifier for 5 neighbors, linear interpolation vector for MFCC (KN - I),

3) support vector machine classifier, MFCC statistics vector as the basis for classification (SVM - S),

4) support vector machine classifier, linear interpolation vector for MFCC (SVM - I).

In all sets, the silence level was defined at the aforementioned level of 30 dB.

Since MFCC values are tensors dependent on the length of the audio signal, it is not possible to use them directly in machine learning systems. It is necessary to first standardize the lengths of the vectors representing the audio signal. Therefore, two approaches were used to address this issue. The first one consisted in determining the statistics vector based on the MFCC tensor, including: mean, standard deviation, median, minimum, maximum, skewness and kurtosis.

The second approach consisted of interpolating the MFCC components and transforming them into vectors of equal length equal to 100 coordinates. This operation aimed to capture the main features of the audio signal and reduce them to a tensor with a dimension independent of the signal length. The interpolation method adopted in this case was linear interpolation, which is very fast and does not generate too much additional computational load.

Before determining the MFCC coefficients, each audio signal was normalized by rescaling it to the interval [-1;1]. This made comparing signals of different amplitudes more efficient. This procedure also eliminated the need to normalize the tensors intended for the machine learning model, because the MFCC coefficients were determined on signals of fixed amplitude (fig. 7, 8 and 9).

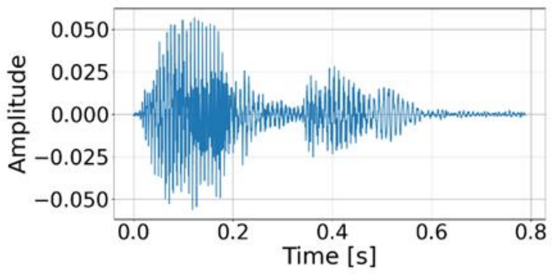

Fig. 7. Example Alpha command

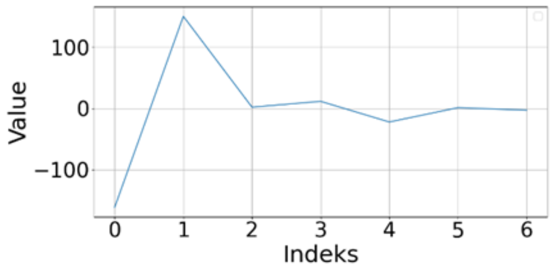

Fig. 8. Statistics graph for the first MFCC coefficient for the Alpha command

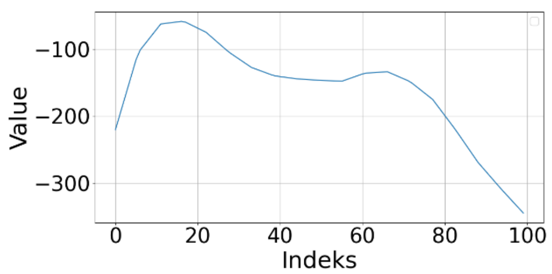

Fig. 9. Interpolation graph for the first MFCC coefficient for Alpha command

3. Results

The process of preparing vectors for machine learning consisted of using 40 previously created samples for each letter and digit. The machine learning process used 80% of the samples for model training and 20% for verification.

The level of accuracy of the models was as follows:

- KN - S: 0.98
- KN - I: 1.00
- SVM - S.: 0.99
- SVM - I: 1.00

Next, the recognition of requests composed of a single command was tested. The results were as follows:

- KN - S: 0.79
- KN - I: 0.92
- SVM - S: 0.85
- SVM - I: 0.95

Then, tests were performed on requests consisting of two commands, which were generated as the Cartesian product of the sets {‘a’, ‘b’, ‘c’} x {‘0’, ‘1’, …, ‘9’}. The results of the recognition efficiency are as follows:

- KN - S:
  - for single command request: 0.70
  - for two command request: 0.50
- KN - I:
  - for single command request: 0.80
  - for two command request: 0.63
- SVM - S:
  - for single command request: 0.82
  - for two command request: 0.67
- SVM - I:
  - for single command request: 0.88
  - for two command request: 0.80

Table 3. Accuracy results summary

	Accuracy of the model	Accuracy of single command request recognition	Accuracy of two commands request recognition
KN-S	**0,98**	**0,79**	**0,70**
KN-I	**1,00**	**0,92**	**0,50**
SVM-S	**0,99**	**0,85**	**0,80**
SVM-I	**1,00**	**0,95**	**0,63**

4. Conclusions

The results of the conducted studies indicate that the effectiveness of recognizing single voice commands using nearest neighbor classifiers and support vector machine-based classifiers is very high, ranging from 98% to 100%. However, tests performed on requests consisting of one or several commands clearly show that this effectiveness significantly decreases when it becomes necessary to extract a single command from the audio signal. An additional challenge in such cases is the noise present in workshops or production halls. Therefore, the feasibility of using voice control in industrial assembly automation systems largely depends on the effectiveness of methods for extracting single commands. Consequently, future research should focus on developing methods for precise localization of commands within the analyzed audio signal.

References

Andrew, A.B., Rajiv, S.A., Jusat, N., Zainuddin, A.A., Subramaniam, K., Rahman, N.A. (2021). Implementation of Low-Cost Voice Command Robot Using Arduino Uno Platform, 2021 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), 2021,pp. 134-139, 23-25 August, 2021, ILMA Journal of Technology & Software Management - IJTSM Vol. 2 Issue. 2. https://doi.org/10.1109/ICSIMA50015.2021.9526315.

Anggraeni, D., Sanjaya, W.S.M., Nurasyidiek, M.Y.S., Munawwaroh, M. (2018). The Implementation of Speech Recognition using Mel-Frequency Cepstrum Coefficients (MFCC) and Support Vector Machine (SVM) method based on Python to Control Robot Arm. IOP Conference Series: Materials Science and Engineering. https://doi.org/10.1088/1757899X/288/1/012042.

Bakouri, M. (2022). Development of Voice Control Algorithm for Robotic Wheelchair Using NIN and LSTMModels, 73(2), 2441-2456. https://doi.org/10.32604/cmc.2022.025106.

Bingol, M. C. & Aydogmus, O. (2020). Performing predefined tasks using the human–robot interaction on speech recognition for an industrial robot. Eng. Appl. Artif. Intell., vol. 95. https://doi.org/10.1016/j.engappai.2020.103903.

Chmielowiec, A., Łysiak K., Vilis J. (2024). Development of a Basic CAM Processor for a Collaborative Robot for Workshop Automation. Technologia i Automatyzacja Montażu, 123(1/2024), p. 37-45. https://doi.org/10.7862/tiam.2024.1.6.

Furui, S. (2009). Selected topics from 40 years of research in speech and speaker recognition. Interspeech 2009, Brighton (UK). https://doi.org/10.21437/Interspeech.2009-1.

Janíček, M., Ružarovský R., Velíšek, K., Holubek, R. (2021) Analysis of voice control of a collaborative robot. Journal of Physics: Conference Series 2021. https://doi.org/10.1088/1742-6596/1781/1/012025.

Janíček, M., Holubek, R., Ružarovský, R., Velíšek, K. (2021). Voice frequency impact for voice control of collaborative robots, IOP Conference Series: Materials Science and Engineering, vol. 1009, Oct. 2021. https://doi.org/10.1088/1757-899X/1009/1/012025.

Khan, R.L., Priyanshu, D., Alsulaiman, F.S. (2021). Implementation of Human Voice Controlled Robotic Car. 10th International Conference on System Modeling & Advancement in Research Trends (SMART), 2021, pp. 640-646, 10-11 Dec, 2021, India, https://doi.org/10.1109/SMART52563.2021.9676319.

Ławrynenko O. (2024). Voice Control System for Robotics in a Noisy Environment. Electronics and Control Systems, Vol. 3 No. 81. https://doi.org/10.18372/1990-5548.81.19016.

Li, S.-A., Liu, Y.-Y., Chen, Y.-C., Feng, H.-M., Shen, P.-K., Wu, Y.-C. (2023). Voice interaction recognition design in real-life scenario mobile robot applications. Applied Sciences, vol. 13, no. 5, pp. 3359. https://doi.org/10.3390/app13053359.

Linda, M., J., Nilesh, V., Rajat, S. (2020). Voice Control Human Assistant Robot. Proceeding of 2nd VNC; VNC-2020, 2nd National Conference on Technical Advancements for Social Upliftment, 4th April, 2020 at Vidyavardhini's College of Engineering & Technology (VCET), Vasai, India. https://doi.org/10.17577/IJERTCONV9IS03084.

Longo, F., Padovano, A. (2020). Voice-enabled assistants of the operator 4.0 in the social smart factory: Prospective role and challenges for an advanced human–machine interaction", Manuf. Lett., vol. 26, pp. 12-16, Oct. 2020. https://doi.org/10.1016/j.mfglet.2020.09.001.

Norda, M., Engel, Ch., Rennies, J., Appell, J.-E., Lange, S.C., Hahn, A. (2024). Evaluating the Efficiency of Voice Control as Human Machine Interface in Production. IEEE Transactions on Automation Science and Engineering 2024, vol. 21, issue 3. https://doi.org/10.1109/TASE.2023.3302951.

Nzuva, S.M. (2021). Investigating human-machine interfaces' efficiency in industrial machinery and equipment: A Kenyan context", International Journal of Information Technology, Control and Automation (IJITCA), vol. 11, no. 4, pp. 1-8, Oct. 2021. https://doi.org/10.5121/ijitca.2021.11401.

Oyelami A.T., Elegbeji, W.F., Adeniyi, O. (2023). 4-degree-of-freedom voice-controlled robotic arm. IAES International Journal of Robotics and Automation (IJRA)2023, Vol. 12, No. 4, pp. 341-351. https://doi.org/10.11591/ijra.v12i4.pp341-351.

Piyaneeranart, M., Ketcham, M. (2021). Automatically moving robot intended for the elderly with voice control. International Journal of Online and Biomedical Engineering (iJOE), vol. 17, no. 06, pp. 19-48. https://doi.org/10.3991/ijoe.v17i06.22299.

Priyadarshana, V., Yaparathne, Y., Jayasooriya, D., Thennakoon, T., Jayasekara, A., D.P., Chandima, D. (2022). Voice controlled robot manipulator for industrial applications. Proc. IEEE 13th Annu. Inf. Technol. Electron. Mobile Commun. Conf. (IEMCON), pp. 0160-0165, Oct. 2022. https://doi.org/10.1109/IEMCON56893.2022.9946545.

Rendyansyah, R., Prasetyo, A.P.P., Sembiring, S. "Voice command recognition for movement control of a 4-DoF robot arm", ELKHA, vol. 14, no. 2, pp. 118, Oct. 2022. https://doi.org/10.26418/elkha.v14i2.57556.