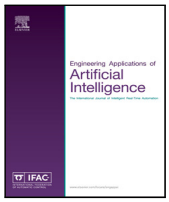
Engineering Applications of Artificial Intelligence 95 (2020) 103903

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Performing predefined tasks using the human–robot interaction on speech recognition for an industrial robot

Mustafa Can Bingol *, Omur Aydogmus

Fırat University, Technology Faculty, Mechatronics Engineering, Elazig, Turkey

ARTICLE INFO

Keywords:
Deep neural networks
Intelligent robots
Human–robot interaction
Robotic vision
Turkish speech recognition

ABSTRACT

People who are not experts in robotics can easily implement complex robotic applications by using human–robot interaction (HRI). HRI systems require many complex operations such as robot control, image processing, natural speech recognition, and decision making. In this study, interactive control with an industrial robot was performed by using speech recognition software in the Turkish language. The collected voice data were converted to text data by using automatic speech recognition module based on deep neural networks (DNN). The proposed DNN (p-DNN) was compared to classic classification algorithms. Converted text data was improved in another module to select the process to be applied. According to selected process, position data were defined using image processing. The determined position information was sent to the robot using a fuzzy controller. The developed HRI system was implemented on a KUKA KR Agilus KR6 R900 sixx robot manipulator. The word accuracy rate of the p-DNN model was measured as 90.37%. The developed image processing module and fuzzy controller worked with minimal errors. The contribution of this study is that an industrial robot is easily programming using this software by people who are not experts in robotics and know Turkish.

1. Introduction

Over the last decades, a large number of robots have work isolated from people for security purposes. However, nowadays, technological developments allow robots to work interactively together with humans in the same environment and on the same process. Communication between humans and robots is the most essential requirement in this working relationship. In the literature, there are many studies on the communication between humans and robots based on speech (Jacob and Wachs, 2016; Yongda et al., 2018), vision (Cherubini et al., 2016; Qu et al., 2019), gestures (Stančić et al., 2017), touch (Bingol and Aydogmus, 2020), emotion identification (Wei and Zhao, 2016) and the use of electromyography signals (EMG) (Zeng et al., 2018). In general, these types of communication are not used alone but together, as in the communication between human and human (Dean-Leon et al., 2018; Jacob and Wachs, 2016; Lin et al., 2018).

Speech is the communication tool that human beings use most frequently because of their natural structure. Therefore, in the literature, speech recognition has been a topic of study for many years. Moreover, with the development of data processing methods and artificial intelligence in recent years, it has become one of the more popular subjects (Moriya et al., 2019; Olson and Belarf, 1957).

Mel Frequency Cepstral Coefficients (MFCC) calculating algorithm is often used to extract the features of speech signals (Grozdić et al., 2017). Features obtained from speech data are classified by algorithms such as Hidden Markov Model (HMM), Gaussian Mixture Model (GMM) and Deep Neural Networks (DNN) (Amrouche et al., 2010; Esfandian et al., 2012; Ting et al., 2013). In addition to such studies, it has been shown that robots can be controlled with ready speech recognition tools (Du et al., 2018; Zinchenko et al., 2017). Communication was achieved with a human, speaking Turkish, to an industrial robot in the current study. Ready-to-use Turkish language speech recognition programs do not work effectively because Turkey has seven different geographical regions with three major Anatolian Turkish dialect groups. Turkish is an official language in Turkey and all dialects is part of Turkish. Although there are three fundamental dialects, they have many variation forms in themselves. For example, “I am coming” in English can be used differently such as “geliyorum”, “geliyom”, “geliyim”, “gelim” Turkish because of dialects of Turkish. Hence, in this study, a ready speech recognition software was not used. An automated speech recognition (ASR) software was developed in order to overcome this variation problem. Speech, which is the primary human communication method, may not always work perfectly as intended. When two people talk to each other, word pronunciation can often be confused. The words; *to*, *two* and *too* in English can be given as a good example. Scientists continue to developed computer-aided software for the detecting of mispronounced words and correcting the pronunciation of the word

* Corresponding author.
E-mail addresses: mustafacanbingol@gmail.com (M.C. Bingol), oaydogmus@gmail.com (O. Aydogmus).

https://doi.org/10.1016/j.engappai.2020.103903
Received 21 January 2020; Received in revised form 3 August 2020; Accepted 16 August 2020
Available online 28 August 2020

(Lee and Glass, 2012; Shahrul Azmi, 2016; Stolcke et al., 2018). However, the purpose of the current study is not to teach the correct pronunciation of words to any system. Unlike other studies, software was employed to detect and correct instances of incorrectly pronounced words. In robot control systems, the information obtained as a result of speech commands is always converted to text data. In a paper written by Wang et al. (2016), each word contained in a text was classified and the classified data used to control at robot. In other similar studies, certain voice commands were converted into text data in order to enable the control of robots (Huang and Lu, 2014; Mašek and Růžička, 2014). Another feature area where robots need to operate smarter is their visual capability. Robots can perform certain desired operations by recognizing their surroundings through visual capability like that of a human or other sighted animal. Zhang et al. realized an application whereby a robot detected people around itself through the use of a camera (Zhang et al., 2013). A similar study was also performed by Li et al. (2012).

Over the years, many methods have been developed for human and robots to work together (Bowyer and Baena, 2015; Du and Zhang, 2015; Ficuciello et al., 2015; Iwata et al., 2005; Kimmel and Hirche, 2017; Nguyen et al., 2005; Rahman and Ikeura, 2016; Yang et al., 2018). It has been seen that robots that are in contact with humans are used more in areas such as wearable robotics (Huang et al., 2019; Li et al., 2018; Pan et al., 2015), and industrial robotics (Kimmel and Hirche, 2017; Rahman and Ikeura, 2016; Sadrfaridpour and Wang, 2018). Speech is the easiest method for an operator to communicate with a robot because speech provides maximum data transfer between operator and robot with minimum effort compared to other methods such as touch or vision. Using this method, Imai, Jensen and their teams succeeded in user-to-robot communication (Imai et al., 2003; Jensen et al., 2005). Since then, Stiefelhagen and his team have provided sight ability to a humanoid robot as well as the ability for a robot to talk with humans (Stiefelhagen et al., 2007). Recent studies have shown that HRI is a constantly developing area of study. Du et al. managed to control an industrial robot using a speech and vision-based method called audio-visual fusion based text (AVFT) (Du et al., 2018). In another study, Yongda and his team achieved the controlling of a robotic arm using motion and speech communication methods (Yongda et al., 2018). In another study, a surgical robotic arm was controlled through voice commands (Zinchenko et al., 2017). In a similar application, a different surgical robot was controlled based on audio and image by Jacob and Wachs (2016). Lin and his team talked to a robot in their study and provided objects to be sorted using a mechanism called case-based reasoning (CBR), belief–desire–intention (BDI), CBR-BDI (Lin et al., 2018).

In another study, a robot was used in order to guide people to their chosen destination within a complex scene (Hu et al., 2019). Part of the study focused on the understanding of text data obtained from a Baidu speech recognition module; a process performed via three different long and short term memory (LSTM) based DNN. In general, studies have translated text data into speech data using speech recognition programs such as Microsoft Azure Software Development Kit (SDK) and Sphinx.

2. Structure of developed software

Developed software includes several sequential processing subprograms. The subprograms, flow-chart and data types are illustrated as shown in Fig. 1.

A single "*Beep*!" sound is heard at the beginning of the recording and a double "*Beep*! *Beep*!" sound is heard through the headset when the recording has finished. The recording starts as automatic and continues in an infinite loop. The duration between these two sounds is two seconds and the operator records speech data by speaking naturally into a microphone. If this speech data is not the "Kuka", recording between "*Beep*!"s continues. If this speech data is the "Kuka", the duration between single "*Beep*!" and double "*Beep*! *Beep*!" changes to five seconds so that it can record more speech data. These recorded data are converted the text by Automatic Speech Recognition (ASR) then transmitted to the Improvement Module of Mispronounced Words (IMMW) in order to obtain the improved data. At this stage, the operator must confirm the task by stating commands of "*Evet*" (yes) or "*Hayır*" (no). If the determined data is not matched by the operator speech, the operation returns to the first step. If the determined task matches the demand task requested by the operator, the Text Understanding (TU) unit and activated. At the same time, the captured image is sent to both the information screen and the Object Detection Module (ODM). The ODM transmits position reference information depending on the desired task to the Robot Position Control (RPC) section. The RPC directs the manipulator to the determined position in order to perform the desired task. At this time, the robot's position information is sent to the information screen and the robot simulation image is updated. Axes and endpoint of manipulator data were used as robot actual position data. Axes data consist of angular positions of six axes (from A1 to A6). Endpoint of manipulator data consist of three linear positions (X, Y, and Z) and three angular positions (A, B, and C). When all operations are completed, the program returns to the start to perform the new commands.

2.1. Automatic speech recognition algorithm

ASR algorithm consists of three parts; words separation, speech feature extraction, and classification.

2.1.1. Words separation method

Speech data can consist of one or more words to be a command sentence. The process of separating words was applied to speech data because of this reason. Steps of the word separation method are as illustrated in Fig. 2.

Equations were shown as *Eq.* in Fig. 2. Dilation process was applied to received speech data. This Dilation process was given as follows:

$$y_d = |y_i| \circ M_d. \tag{1}$$

where y_d is the output of the dilation process, y_i is the speech data, M_d is the dilation mask, and $\circ$ symbolizes the dilation process. It was observed that the integrity of the word broke down when the size of M_d was reduced. More than one word or together with word and background noises were detected when the size of M_d was increased. In the current study, M_d was chosen as a size 10,000×1 true vector. After this step, the threshold step was carried out the obtained signal. Eq. (2) was applied to the obtained signal in order to represent it in a logical form.

$$y_l(k) = \begin{cases} 1 & y_d(k) > T_d \\ 0 & y_d(k) \le T_d, \end{cases} \quad k = 1, 2, \ldots, n \tag{2}$$

where, y_l is the logical representation of the dilation signal, k the is discrete time index, n equals samples number of speech data, and T_d represents the threshold coefficient determined for logical expression. After the trials, noise terms were detected as words when T_d was chosen as a low value and any word was not detected when T_d was chosen as a high value. Therefore, T_d can be chosen range of 0.001 and 0.1. T_d was chosen as 0.01 in the current study. Possible start and end indices of the words are obtained using Eqs. (3) and (4).

$$s_{start}(k) = (y_l(k) \oplus y_l(k+1)) \cdot y_l(k+1), k = 1, 2, \ldots, n-1 \tag{3}$$

$$s_{stop}(k) = (y_l(k) \oplus y_l(k+1)) \cdot y_l(k), k = 1, 2, \ldots, n-1 \tag{4}$$

where S_{start}, S_{stop}, "$\oplus$" and "$\cdot$" refer to the start operation, stop operation, XOR operator, and the AND operator, respectively. After this process, S_{start} and S_{stop} are signal such as "000100". Here, "*1*" indicates a possible start or and stop of word index in these signals. The speech data is divided into probable words using the start and end indices of

Fig. 1. Block diagram of developed software.

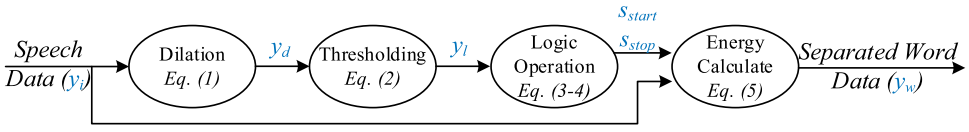


Fig. 2. Steps of word separation method.

the words determined, and the energies of the signals are calculated by using as follows:

$$Ep = \sum_{i=m}^{n} \frac{(y_i(i)^2 + y_i(i+1)^2)}{2} Ts. \tag{5}$$

In this equation, E_p represents the energy of the component and the T_s is the sampling period. In addition, m denotes the index of the S_{start} signal for the logic-1 and n symbolizes the index of the S_{stop} signal for the first logic-1 after the m-th moment. Part from $y_i(m)$ to $y_i(n)$ is treated as words when their energy value exceeds a predefined threshold. Energy values below the predefined threshold are evaluated as noise components. For example, Kuka and cleaning words were separated as given in Fig. 3.

Where, blue, red, and yellow signals were shown speech data, dilation output, and logical form of dilation signal, respectively. Green and red dot were illustrated possible start and stop of word index, respectively. Also, E_p was calculated between these determined start and stop index. Words are ready to be cut and use because they are higher than the determined threshold value.

2.1.2. Speech feature extraction method

There are many feature extraction methods for audio signals in the literature. The MFCC calculating algorithm is one of the most popular feature extraction methods in the literature (Davis and Mermelstein, 1980; Jothilakshmi et al., 2009). Therefore, this method was chosen to be used in the current study. The MFCC calculating algorithm steps are as illustrated in Fig. 4.

The pre-emphasis process improves the signal noise ratio by increasing the amplitude values at higher frequencies more than at low frequencies. The expression of this process is given in Eq. (6).

$$y_e(k) = y_w(k) - a y_w(k-1), k = 2, 3, \ldots, n, a = 0.97 \tag{6}$$

where, y_e is the emphasized signal, y_w is the word separated speech signal and a is a filter coefficient between 0.9 and 1.0. The signal obtained after this process is framed as follow:

$$y_{fr} = \begin{bmatrix} y_e(1) & y_e(2) & . & . & y_e(f_L) \\ y_e(R+1) & y_e(R+2) & . & . & y_e(R+f_L) \\ y_e(2R+1) & y_e(2R+2) & . & . & y_e(2R+f_L) \\ . & . & . & . & . \\ y_e(mR+1) & y_e(mR+2) & . & . & y_e(mR+f_L) \end{bmatrix}_{f_{sig} \times f_L},$$

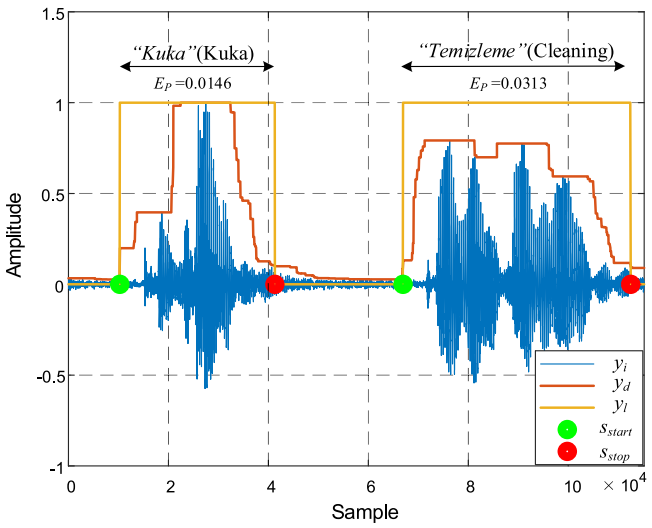


Fig. 3. Example of word separation method.. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

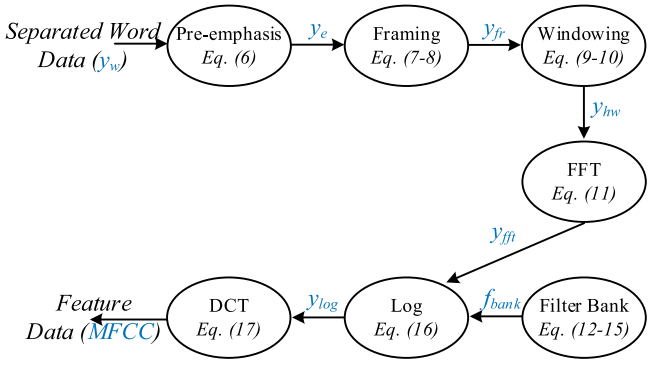


Fig. 4. MFCC calculating algorithm steps.

$$m = 0, 1, \ldots, f_{sig}. \tag{7}$$

y_{fr} and f_{sig} are framed signal and framed signal size, respectively. The f_{sig} can be calculated as follow:

$$f_{sig} = \frac{s_d - o_L}{R}, R = f_L - O_L, f_L = 1920(40\text{ ms}), O_L = 480(10\text{ ms}). \tag{8}$$

Here, s_d, f_L, and O_L are illustrated speech duration time, frame length, and overlap length, respectively. The overlap is intersection of the sequential two framed signal. After this step, hamming window is applied as below:

$$hw(k) = \alpha - \beta\cos(\frac{2\pi k}{f_L - 1}), k = 1, 2, \ldots, f_L, \alpha = 0.54, \beta = 0.46, \tag{9}$$

$$y_{hw} = y_{fr} \cdot \begin{bmatrix} hw(1) & hw(2) & . & . & hw(f_L) \\ hw(1) & hw(2) & . & . & hw(f_L) \\ . & . & . & . & . \\ . & . & . & . & . \\ hw(1) & hw(2) & . & . & hw(f_L) \end{bmatrix}_{f_{sig} \times f_L}. \tag{10}$$

Where, α and β represent the window coefficients. A y_{hw} matrix is obtained which is framed and windowed by f_{sig}×$f_L$dimensional. The y_{hw} matrix is converted y_{fft} using Fast Fourier Transform (FFT). The expression of this process is given in Eq. (11).

$$y_{fft} = |FFT(y_{hw})|^2 \tag{11}$$

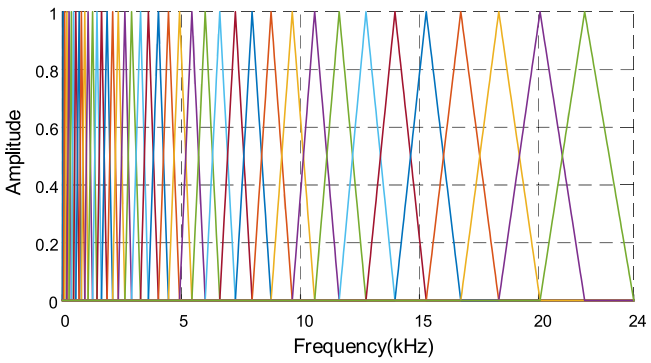


Fig. 5. Filter bank.

After this process, filter bank is determined using Eqs. (12)–(15).

$$Mel(f) = 2595\log(1 + \frac{f}{700}), f = 48\text{ kHz} \tag{12}$$

$$P_{mel}(k) = \Delta_{mel}k, \Delta_{mel} = \frac{Mel}{F_{size} + 2}, k = 1, 2, \ldots, n \tag{13}$$

$$f_{mel}(k) = 700(10^{P_{mel}(k)/2595} - 1), k = 1, 2, \ldots, n \tag{14}$$

$$y_{tri}(k) = \begin{cases} 0 & k < f_{mel}(m-1) \\ \frac{k - f_{mel}(m-1)}{f_{mel}(m) - f_{mel}(m-1)} & f_{mel}(m-1) \le k < f_{mel}(m) \\ 1 & k = f_{mel}(m) \\ \frac{f_{mel}(m+1) - k}{f_{mel}(m+1) - f_{mel}(m)} & f_{mel}(m) < k \le f_{mel}(m+1) \\ 0 & k > f_{mel}(m+1), \end{cases}$$
$$k = 1, 2, \ldots, n$$
$$m = 2, 3, \ldots, F_{size} + 1 \tag{15}$$

Here, *Mel*, F_{size}, and y_{tri} represent the Mel-Frequency, feature size, and triangle function of filter banks, respectively. Other parameters are calculated in equations. y_{tri} is calculated for each of *m*-value (in Eq. (15)) and these values are formed the filter bank (f_{bank}). f_{bank} can be seen for 30-F_{size} in Fig. 5.

After this step, logarithm process is applied as in Eq. (16).

$$y_{\log} = 20\log(y_{fft}f_{bank}) \tag{16}$$

y_{log} is typified power spectrum of signal and power spectrum of y_i, which was given in Fig. 3, was given in Fig. 6.

The MFCC features is obtained from using Eq. (17).

$$MFCC = DCT(y_{\log}) \tag{17}$$

Here, DCT represents discrete cosine transform. After obtained feature matrix, the matrix and target label was classified using some algorithms.

2.1.3. Classification methods of speech features

Along with the developing technology, deep neural networks (DNN) are now used in many fields. Three basic structure of DNN structures such as CNN, LSTM, and CNN+LSTM is frequently used (Oh et al., 2018; Petridis et al., 2020; Zoughi et al., 2020). The DNN models such as single layer convolutional neural networks (CNN), single layer long short-term memory (LSTM), and complex DNN which was formed single layer CNN with single-layer LSTM were created as basically. The models were trained by using feature matrix and target labels. Created models were given in Fig. 7.

The meanings of used layers were given as follows.

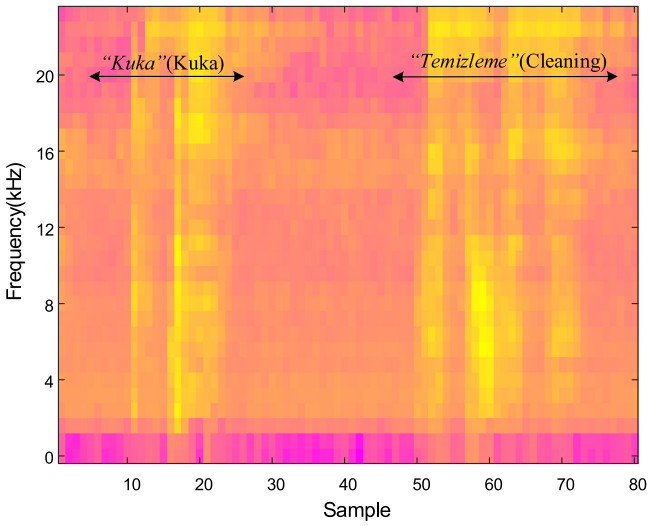


Fig. 6. Power spectrum of y_i.

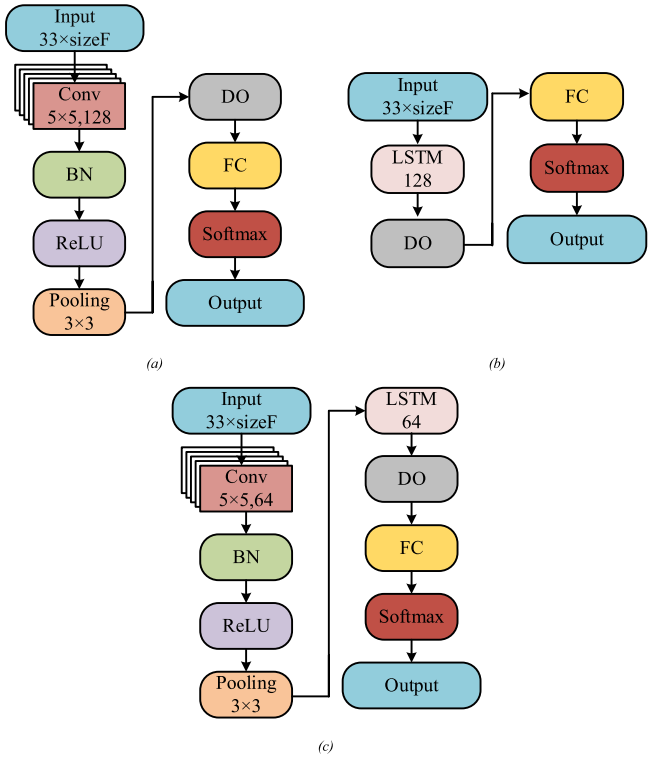


Fig. 7. Three basic DNN models; (a) Single layer CNN, (b) Single layer LSTM, (c) Complex DNN.

- *Convolutional Layer (Conv):* In this layer, a large number of mathematical operations are performed. This layer applies convolution process to the input matrix using a pre-determined mask.
- *Long Short-Term Memory Layer (LSTM):* LSTM layer processes like recurrent nets. The layer learns dependencies of long time sequences.
- *Batch Normalization Layer (BN):* BN normalizes the mean and variance before using the activation function. Thus, the learning performance of the designed network increases as a result.
- *Rectified Linear Unit Layer (ReLU):* This layer, which can be defined as the rectifier of artificial neural networks. Activation of the data is achieved using $f(x)=max\ (0,\ x)$ equation.
- *Pooling Layer*: This provides a selection according to the desired features of input matrix elements in the pre-determined dimensions. Varieties include average, maximum pooling layer, etc.
- *Dropout Layer (DO):* DO was used to avoid overfitting, which is one of the problems facing artificial neural networks, by passivating some inter-layer connections.
- *Fully Connected Layer (FC):* The FC layer links all data between the input and output layers, and determines the relationship between the output value of each input value.
- *Softmax Layer*: This layer, which is used for classification problems, converts the input data to the number of classes using the *softmax* function.

The input layer shown in Figs. 7–8 is the layer in which the feature matrix of the audio data is applied. In the first layer, sizeF symbolizes the number of features. In addition, the number 33 consists of a one second speech signal with an overlap duration of 10 ms and 40 ms of framing. The number 33 was calculated using Eq. (8). The last layer is the output layer of the text data. In single layer CNN (Fig. 7a), the filter size of the convolutional layer was chosen as row× columns = 5×5, padding was selected as same, stride was set as 1×1, and the number of the filter was selected as 128. Also, the pooling layer was chosen maximum pooling. The filter size of the pooling layer was determined as 3×3 and stride was assigned as 2×2. In single layer LSTM (Fig. 7b), hidden unit number of LSTM layer was chosen as 128. In complex DNN (Fig. 7c), the filter size of the convolutional layer was chosen as 5×5, padding was selected as same, stride was set as 1×1, and the number of the filter was selected as 64. The pooling layer was chosen maximum pooling. The filter size of the pooling layer was determined as 3×3 and stride was assigned as 2×2. Hidden unit number of LSTM layer was chosen as 64.

The created DNN models were trained and the best result was obtained using designed single layer CNN, also detailed results were given Section 4. The proposed DNN (p-DNN) structure, which was designed by improving single layer CNN, is shown in Fig. 8.

In p-DNN, the filter sizes of the convolutional layers were chosen as 5×5, paddings were selected as same, strides were set as 1×1, and the numbers of the filters were selected as 32, 32, 64, and 128, respectively. The pooling layers was chosen maximum pooling. The filter size of the pooling layer was determined as 3×3 and stride was assigned as 2×2.

Drop out parameter was chosen as 0.2 in all DNN structures to solve overfitting problem. All DNN structures were trained during 25 epochs using Adam Optimization Algorithm. Learn rate was chosen as 1e-4 during the first 20 epochs then learn rate was updated 1e-5. Also, mini batch sizes of DNNs was assigned as 8.

After completion of all these operations, the natural speech data is translated into text data.

2.2. Improvement module of mispronounced words

Human beings often confuse closely-pronounced words and decide upon which variant to use according to the meaning of the sentence. As with most languages, there are similar letters and words that can be mispronounced in the Turkish language. In the current study, P was used to express the point where the robot is in space and the letter B, which is one of the rotation axis of the robot, was also used. The pronounced audio data of the letters "P" and "B" in Turkish are shown in Fig. 9. Also, detailed information about the pronounce of letters in Turkish can be found in webaddress (Turkish Alphabet Pronunciation, 2019).

In Fig. 9c, the red dot is the peak point of the similarity point. Audio data of the letter "P" was shifted in the time domain according to the red dot and the similarity of two signals was calculated as 0.0307 using mean-squared-error (MSE). The similarity of letters "Z" and "B" audio data was investigated using the same methods to understand how

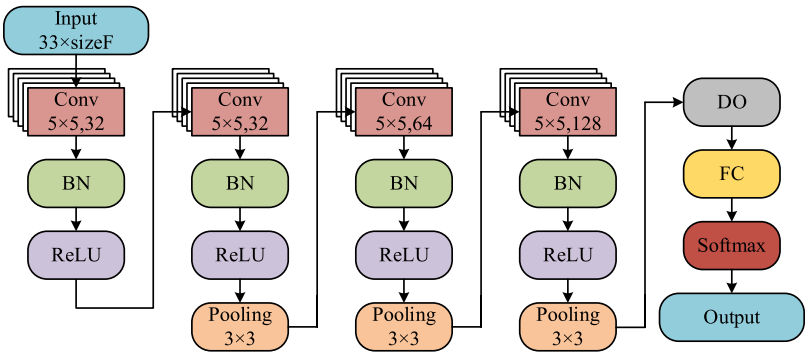


Fig. 8. Structure of proposed DNN (p-DNN).

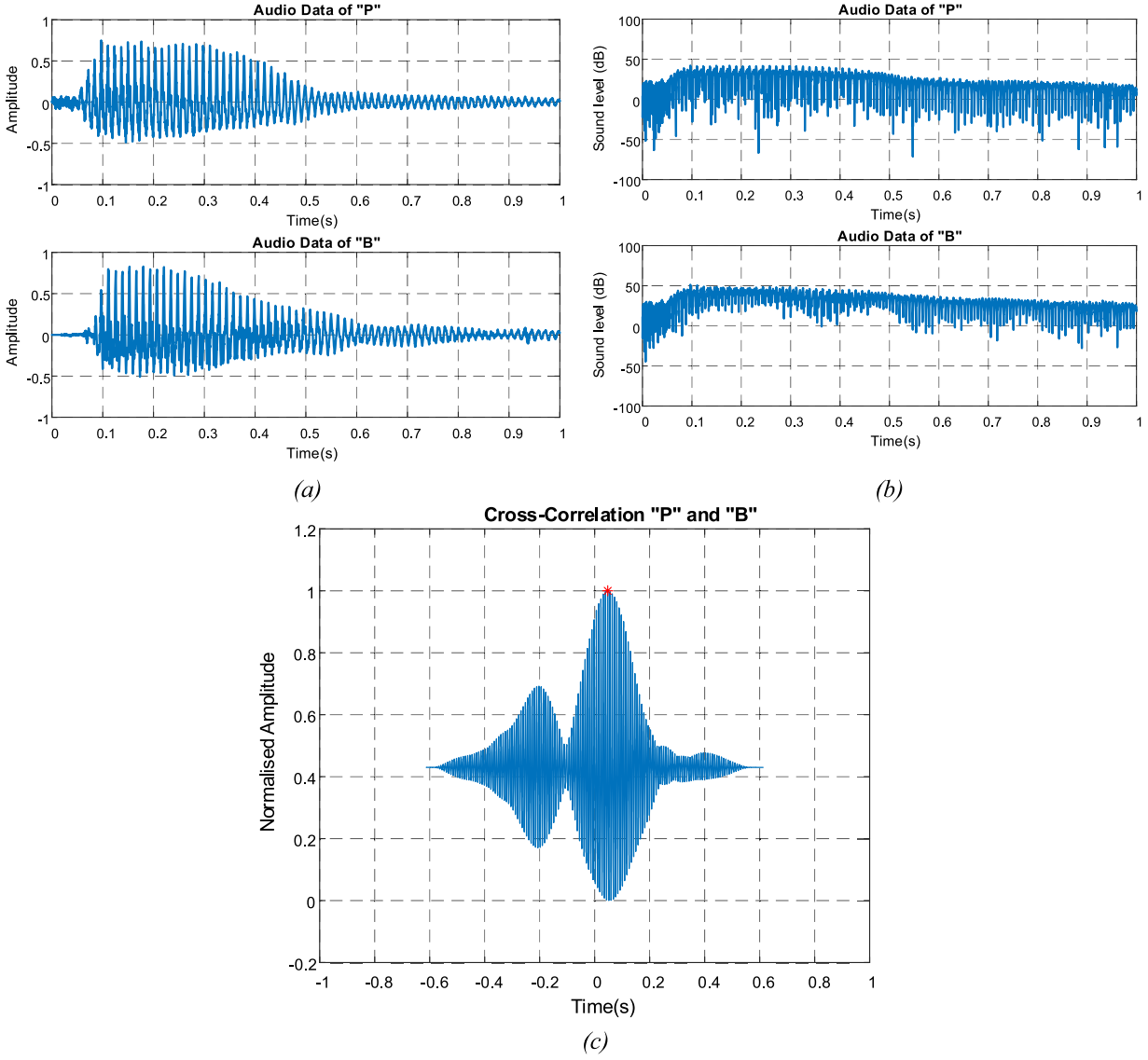


Fig. 9. Turkish pronounced audio data for letters "P" and "B"; *(a)* amplitude of audio data, *(b)* sound level of audio data; *(c)* cross-correlation between audio data.

similar of "P" and "B" audio data. MSE of "Z" and "B" audio data was found as 0.0646. This result was shown that audio data of the letter "P" was more similar to the letter "B" audio data than the letter "Z" audio data. Due to the similarity of these sound signals, the designed ASR system can sometimes produce false results. Therefore, IMMW was included in the software design and build. The IMMW consists of Finite Automata (FA), as can be seen in Fig. 10.

In the mechanism illustrated in Fig. 10, S_t is the starting node and the end node is F_n. The processes of N, S_r, and C_h indicate that selecting the next word in the text, searching word in the text, and changing the word in the text, respectively. Starting node steers to the end node in all other conditions. The red words in Fig. 10 are the English equivalents of the Turkish words. As an example of IMMW, "z yap (do z)" is the wrong sentence because z is named as one axis position of manipulator

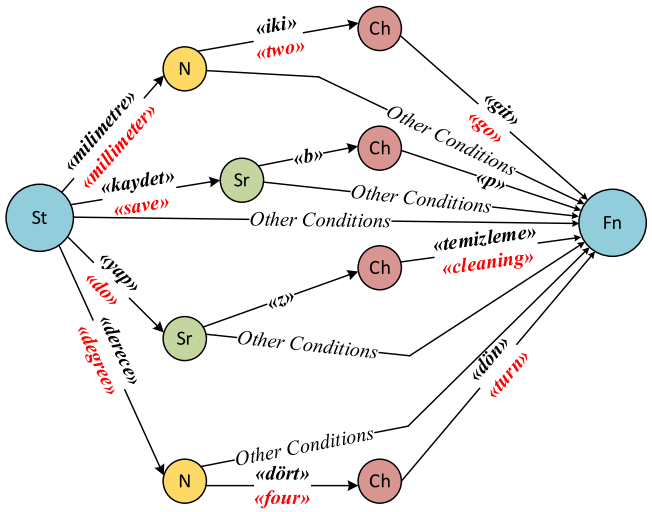


Fig. 10. Mechanism of IMMW.

orientation. The wrong sentence is changed with "temizleme yap (do cleaning)" by IMMW. As another example of IMMW, "b-1 kaydet (save b-1)" is the wrong sentence because b is named as one axis position of manipulator orientation. The wrong sentence is changed with "p-1 kaydet (save p-1)" by IMMW and the sentence is made sense for Text Understanding Algorithm.

2.3. Text understanding algorithm

Each word in the text obtained from the speech signal is classified into nine groups as *noise*, *letter*, *direction*, *number*, *other*, *unit*, *operation*, *order,* and *confirmation*. Detailed information about clusters content was given appendix. These clusters are numbered from zero to eight, respectively. The words are sorted from small to large according to their numbers. For example, when the operator says; *"git x'de sekiz milimetre"* in Turkish, which is "*go eight millimeters on x*" in English, the value vector becomes [8,1,3,5]. The value vector is arranged from small to large and the text data is arranged as *"x'de sekiz milimetre git"* because the verb is the last word of the sentence in Turkish grammar. The process is determined according to the verb and the remaining words are converted to numerical data and used in the function. If the process is confirmed by the operator, the final version of the expression is *git(x, 8)* in the program, as given in the example. Picking, cleaning, aligning, welding, drilling, and sealing operations, which are frequently performed in the industry, can be automatically performed with the ASR, IMMW, and Text Understanding Algorithm. Also, an industrial robot can be programmed as a manual with the same structures. Tool center point of robot could be moved six linear (go- (forward, backward, right, left, up, and down)) on X, Y, and Z axes and two angular motion (turn- (right and left)) on the axes for programming the robot manually. Technical terms such as clockwise, counterclockwise, plus, and minus were not used to move robot tool center point because the aim of study is that an industrial robot was easily programmed by people who are not experts in robotics. Also, wake-up system services could be performed with the word dataset for your *'Kuka'* robots. The current paper was contained wake-up, picking, cleaning, alignment, welding, and drilling processes.

2.4. Object detection module

In this study, the robot can be programmed by a non-expert user. Both in order to enable this feature and in order to perceive the environment, ODM was developed. The processes performed by the robot were chosen in accordance with industrial applications so as to obtain a modernization of an industrial robot. Our previous work was that a symbolic hole was drilled in the center of random objects within the robot's workspace via the robot equipped with a symbolic drilling tool (Bingol and Aydogmus, 2018). In the current study, in addition to the drilling process, the robot is programmed to undertake three different additional tasks which are picking, aligning and cleaning. Red points, marked on A4 paper, was determined as the target in the drilling process. The determined process consists of two steps: from Red–Green–Blue (RGB) image to Hue-Saturation-Value (HSV) image converting and HSV thresholding. Parameters was selected as "Hue<0.065 or 0.952<Hue", "0.282<Saturation<1.000", and "0.474<Value<0.882" for HSV thresholding process. After determining the target drilling point, the manipulator takes the drilling tool and goes to the drilling point. Circle objects, which are 30 mm diameter and 20 mm width, were used in the picking and aligning process. Algorithm 1, shown in Fig. 11, was developed to picking and aligning process.

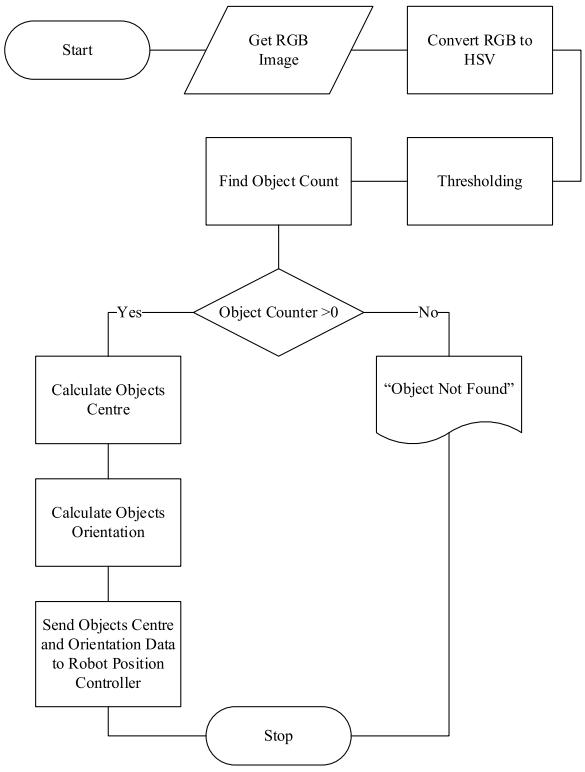


Fig. 11. Object detection algorithm.

Where, information about *Find Object Count*, *Calculate Object Center*, and *Calculate Objects Orientation* algorithms was given in the appendix.

2.5. Robot position control algorithm

A low cost (approx. $2) camera was used in the study. Therefore, the quality of the taken image does not allow for linear transformation from pixel to cm. Therefore, endpoint of manipulator cannot be moved directly to the position of the determined object. The communication between the robot and the PC is realized via Transmission Control Protocol-Internet protocol (TCP-IP). There is no fixed time concept because of communication protocol. For this reason, it is not possible for the controller to simultaneously transmit/receive integral and derivative components. Only proportional controller was preferred since there is no time term. Three kinds of proportional controllers were designed in order to control the system. Later, a Takagi-Sugeno type fuzzy logic controller was designed to switch between these three controllers. The controller parameters were given as 0.10, 0.05, and 0.00 in fuzzy rules,

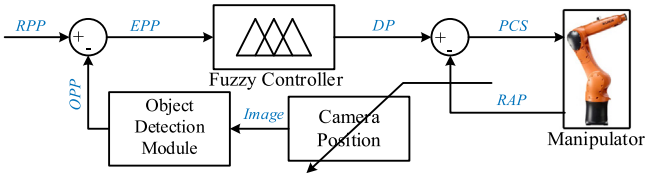


Fig. 12. Controller block diagram.

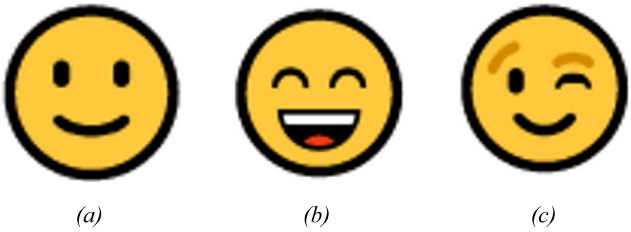


Fig. 13. Information expression of developed software; (a) active listening expression, (b) aware listening expression, (c) desired command execution expression.

respectively. A control block diagram of the system is presented in Fig. 12.

Reference pixel position (RPP) was setting as x=280 px and y=311 px according to camera position on the tool of the manipulator in Fig. 12. Error pixel position (EPP) was calculated using RPP and Object pixel position (OPP), which getting from ODM. According to EPP, Delta Position (DP) was adjusted by the fuzzy controller. Position Control Signal (PCS) was calculated using DP and robot actual position (RAP). Also, when the manipulator is moving, position of the camera changes. Thus, image data getting from was updated continuously. Endpoint of manipulator was moved according to PCS. Using this control system, the position of the reference pixel is matched with the position of the gripper. The fuzzy rules for the study as follows:

- R1: IF $EPP > 5$ THEN $DP = 0.10EPP$
- R2: IF $EPP < 5$ AND $EPP \neq 0$ THEN $DP = 0.05EPP$
- R3: IF $EPP = 0$ THEN $DP = 0$

2.6. Information screen software

Fundamental of HRI is that the human communicates with the robot and vice versa. Two different methods were used to facilitate communication between the robot and operator. In the first method, the robot notifies the operator of the range that he is listening to using the start and end notification sounds. The second method utilizes the information screen. This information screen, as illustrated in the experimental setup in Fig. 14, consists of four different parts such as emoji, text, camera, and simulation.

- The emoji part provides a visual feedback to the operator (see Fig. 13) on whether or not the system is actively listening, aware listening, or executing an operation.

- The text provides information about whether or not the operators speech has been understood correctly.
- The camera provides the operator with an image of the gripper's orientation and position.

The simulation displays information regarding the position of all joints as a robot dummy on the screen.

3. Implementation of developed software

The developed software was tested on an industrial robot KUKA KR Agilus KR 6 R900 sixx. The experiment was setup as illustrated in Fig. 14.

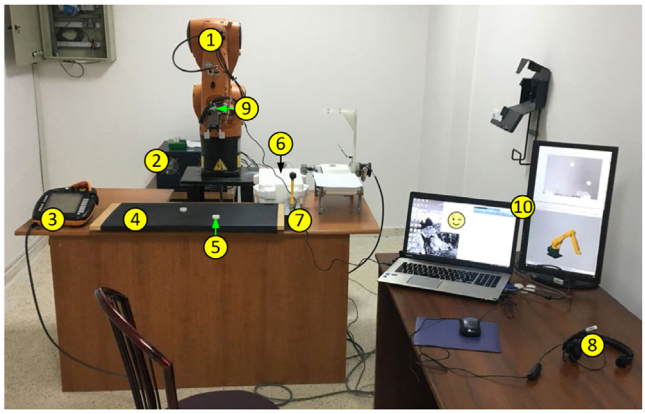


Fig. 14. Experimental setup.

Table 1
Distribution of individuals according to regions.

Region	Dialects	n	(%)
Black Sea	NAD	2	6.67
Marmara	WAD	4	13.33
Aegean	WAD	2	6.67
Mediterranean	WAD	5	16.67
Central Anatolia	WAD-EAD	4	13.33
Eastern Anatolia	EAD	9	30.00
Southeastern Anatolia	EAD	4	13.33

Table 2
Word accuracy rate of DNN models.

Feature size	Single layer CNN (%)	Single layer LSTM (%)	Complex DNN (%)	p-DNN (%)
5	74.88	52.19	71.64	84.66
10	77.96	65.46	76.15	87.75
20	85.18	72.16	80.28	86.73
30	86.72	75.64	77.83	**90.37**

An industrial robot has standard parts such as a manipulator (1), a cabinet "*robot PC and motor drives*" (2) and a control pad (3), as shown in the experimental setup in Fig. 14. Other elements shown in Fig. 14 are labeled as follows; reference plane (4), objects (5), picking box (6), and tool magazine (7). The headset and microphone are used for audio data (8), whilst the camera (9) is used to obtain information of the working space. A computer (10) was used to operate the developed software with an additional information screen. KUKAVARPROXY software was used to provide the connection between the PC and the robot via TCP-IP. The system was realized in MATLAB program.

4. Results

The ASR system is created by using as 60 different audio files obtained from 30 different individuals. The audio files were collected as mixed from each of the regions of Turkey. Distribution of individuals according to regions was given in Table 1.

Here, NAD, WAD, and EAD were shown Northeastern Anatolian Dialects, Western Anatolian Dialects, and Eastern Anatolian Dialects, respectively. The number of Eastern Anatolia Region participants was greatest because the study was conducted in the Eastern Anatolia Region. The situation has not performed any disadvantage when examined the result of the speaker independence test. One-thirds of the 60 sound files collected were not used due to recording in very noisy environments. The remaining 40 audio files were used in the current study. These initial audio files contained a total of 45 different words. Unknown words and background noise were taken from other dataset words as randomly (Speech Commands Dataset Version 1,

Table 3
Word accuracy rate of conventional classification algorithms and p-DNN.

Feature size	DT (%)	DA (%)	SVM (%)	k-NN (%)	RF (10) (%)	RF (100) (%)	RF (1000) (%)	p-DNN (%)
5	32.47	63.78	69.71	56.95	50.38	67.13	70.36	84.66
10	31.70	71.26	73.71	58.11	51.80	71.39	73.06	87.75
20	34.14	75.15	72.68	61.72	52.31	69.84	74.61	86.73
30	33.76	71.90	76.67	50.51	51.67	70.74	74.35	**90.37**

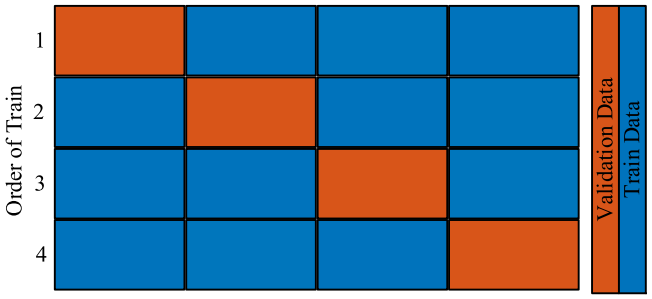


Fig. 15. Parts of data for cross validation.. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

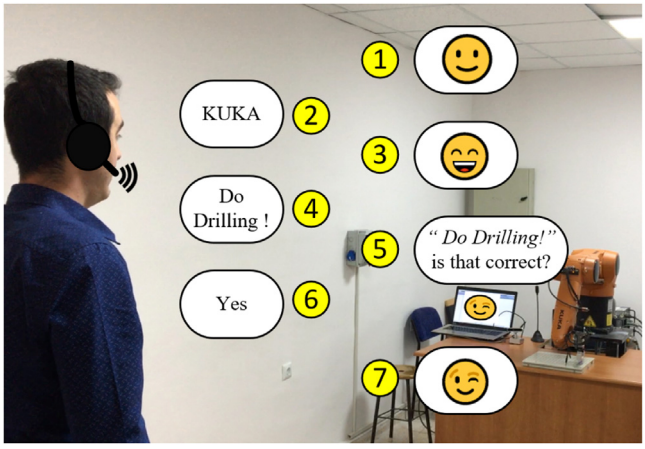


Fig. 16. Example of interaction between human and robot.

2017). A total of 47 different words (45 different, unknown words, and background noise) were classified. The feature size was selected as 5, 10, 20, and 30 in order to investigate relationships between feature size and accuracy. The feature size was not selected greater than 30 due to operational load. The aggregated data was separated into two parts, which were train dataset (75.00%) and validation dataset (25.00%). Also, to investigate whether the system is dependent on speakers, two hundred speech word data were collected from four females (from Aegean Region) and four males volunteer speakers (from Eastern Anatolia Region), who were not in dataset. The collected data was used as test dataset. In summary, the collected data is divided into three parts: train, test and validation. Train dataset was used in order to train model. Validation dataset was used in order to tune model parameters such as filter size of the convolutional layer and filter number of the convolutional layer. Test dataset was used to measure performance of model (Ripley, 2008).

The values in Table 2 and III were calculated using Word Accuracy Rate (WAR) formula that dividing the number of true predictions of validation data to the total number of validation data. The classification results of DNN models are given in Table 2.

Decision tree (DT), discriminant analysis (DA), support vector machine (SVM), and k-Nearest Neighbors (k-NN) are widely used in classification problems (Bingol and Aydoğmuş, 2019). Kernel functions such as Polynomial, Gaussian, and Linear were tested kernel functions of SVM. Second-order polynomial kernel function was selected as SVM kernel functions because the minimum error value was obtained. Some neighbor numbers from 1 to 15 were tested number neighbors of k-NN. The Number of neighbors was selected as ten because the best result was obtained. Also, recently random forest (RF) has been used to classify audio features (Vafeiadis et al., 2020). In the current paper, forest trees leaf count was selected as 50. As can be seen in Table 2, algorithms are tested after the training process according to feature sizes. Confusion matrices of p-DNN were given in the appendix.

Where, RF (10), RF (100) and RF (1000) were shown that there were 10, 100 and 1000 trees in the forest, respectively. The best result, which was shown bold, in Table 3 was obtained using k-fold cross-validation. k was selected as 4 and data parts was given in Fig. 15.

In Fig. 15, speech word dataset was divided into 4 parts and validation accuracy of each part was calculated. Train and validation parts were shown blue and red colored boxes, respectively. WAR results was calculated as 89.56%, 90.98%, 90.72%, and 90.21% according to train order, respectively. Final WAR was obtained as 90.37% by taking average of four WAR. Thirty feature size was used since the word accuracy rate of the feature size was highest. The feature matrix was obtained from the test dataset according to 30 feature size. The test dataset was classified using p-DNN and WAR was obtained as 78.50%. p-DNN and other classification were compared by using Tukey multivariate comparison test and results was given in Table 4.

There is significantly statistical difference between p-DNN and other algorithms. Also, statistical relationship between feature size and WAR was investigated for p-DNN and any significant correlation was not found (p=0.20). The value was shown that two parameters did not linear relationship together. Therefore, designed p-DNN was nonlinear. The reason for p-DNN nonlinearity is that the p-DNN was formed using many parameters, hyperparameters, and nonlinear mathematical formulations.

Drilling, aligning, picking, and cleaning operations were performed by the robot after analyzing the speech expression in the ASR, IMMW, and Text Understanding sections. An example of the speech between an operator and the robot is given in Fig. 16.

The image captured from the camera mounted on the gripper during the drilling process is shown in Fig. 17.

In this work, a real drilling tool was not used because the reliability test of the software has not yet. However, after making sure that the software is safe in the long term, we will perform real drilling operation. The drilling operation was imitated to a plastic apparatus as can be seen in Fig. 17. First, the robot set the target after receiving the drill command. The process was shown in Fig. 17a. Here, the red dot was determined as a drilling position. Then, the manipulator moved the tool station as in Fig. 17b in order to get drilling tool. Finally, the drilling tool was carried away to the target point by the manipulator. The drilling tool was oriented satisfactorily to the center of the goal (see as Fig. 17c). The tool center point was oriented from the home position to the marker position as illustrated in Fig. 18.

Satisfactory results were obtained as shown in Fig. 18. The centers and orientation information of the objects must be determined first for aligning and picking operations. These processes were achieved using the object detection algorithm, the same as in the drilling process. The orientation lines are indicated on the objects to be aligned and picked as shown in Fig. 19a.

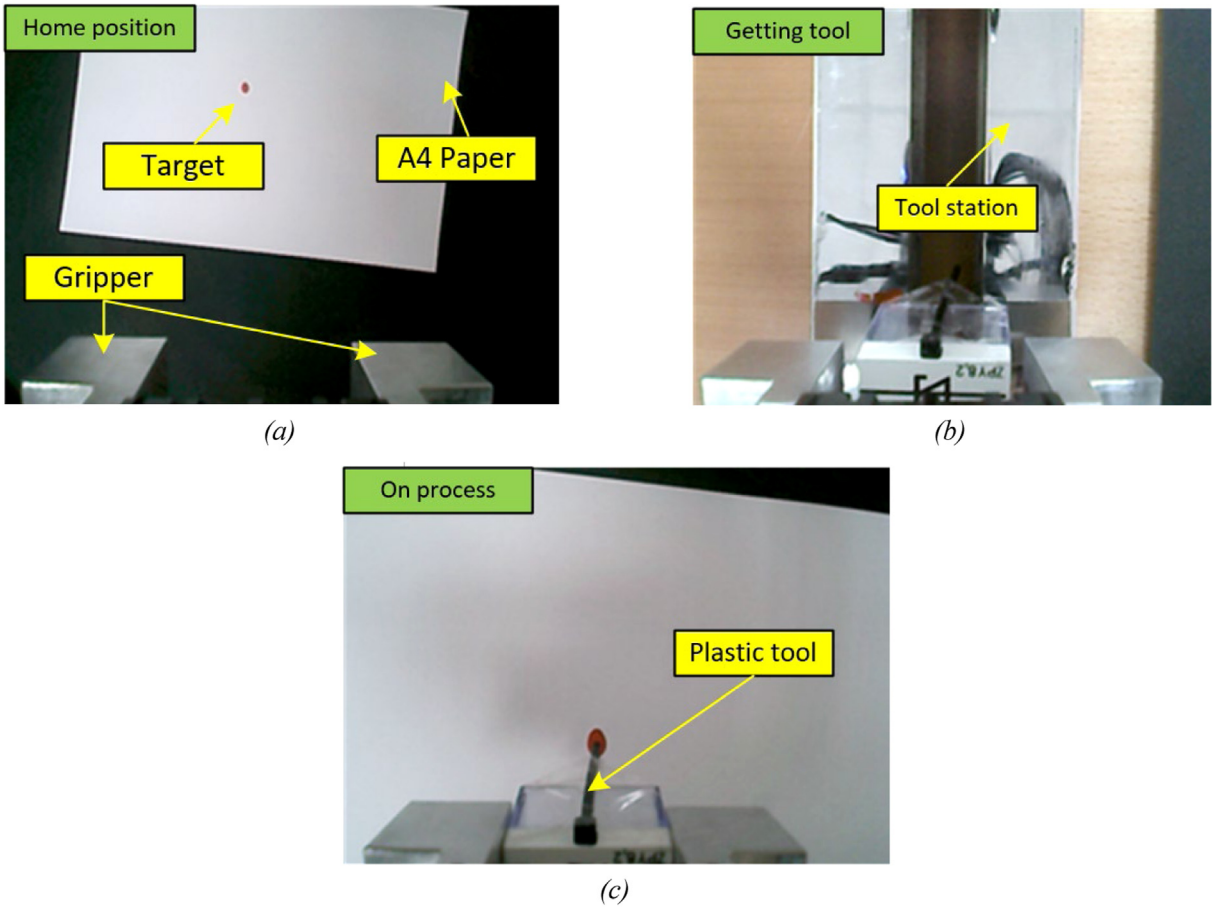


Fig. 17. Images of drilling steps; (a) home position, (b) imitated drilling tool gripping, (c) end of drilling.

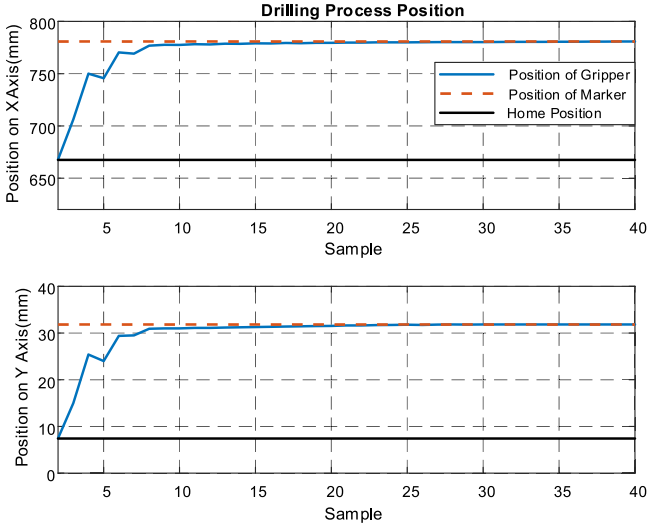


Fig. 18. Positions of gripper, marker, home during drilling process.

Fig. 19b shows the sections which are used for the aligning and picking operations. The objects are stacked in the left area for the aligning operation, and they are picked up and placed into the pickup box in the picking operation. The aligning and picking operations were performed using randomly placed objects situated on the plane. The position of the gripper was obtained as shown in Fig. 20 during both of these operations.

As can be seen in Fig. 20, the robot correctly detected the positions of randomly placed objects on the plane and performed the desired alignment and picking operations successfully. Furthermore, the implementation video of the current study can be watched in webaddress (Video of current work, 2020).

5. Discussion

The developed software consists of a complex structure containing multiple modules. This software cannot compare any studies in the literature because the software is focused on a specific problem solution. Therefore, the developed modules in the software were compared studies in the literature. Firstly, when it was investigated the articles that were used the speech as a communication type with robots, it was seen that third-party speech recognition software was utilized in most of the studies (Hu et al., 2019; Stiefelhagen et al., 2007; Yongda et al., 2018). As the third-party speech recognition software depends on an internet connection or the Turkish language is unsupported in the software, the third-party speech recognition software did not use in the current study.

Secondly, when it was examined Turkish ASR studies in the literature, the studies could be separated two classes as phoneme-based and word-based (Cakir and Sirin, 2018; Çarkı et al., 2000; Tombaloglu and Erdem, 2017; Tombaloğlu and Erdem, 2016). Word-based Turkish ASR software was performed in the study of Cakir and Sirin, similar to current study (Cakir and Sirin, 2018). Fifteen speakers vocalized ten different words and the speech voices were classified with an accuracy of 86.60% in the study of Cakir and Sirin (Cakir and Sirin, 2018). It was determined that created Turkish ASR was dependent on speakers in the other two studies. (Tombaloglu and Erdem, 2017; Tombaloğlu and Erdem, 2016). In the current study, in order to be independent

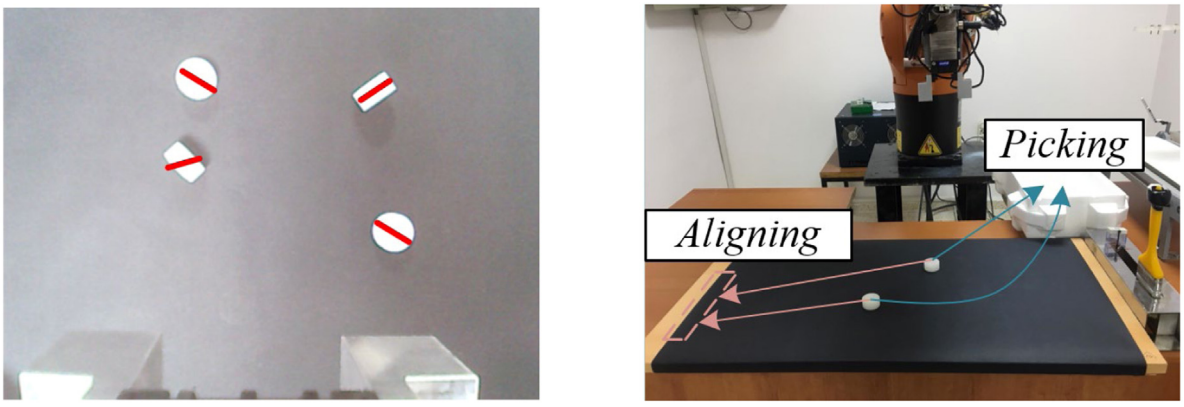


Fig. 19. (a) Object center and orientation image, (b) information alignment and picking process.

Table 4
p-DNN and other algorithms comparison.

Algorithm	Mean ± SD (WAR)	p (According to p-DNN)
DT	33.08 ± 1.09	<0.001
DA	70.52 ± 4.80	<0.001
SVM	73.19 ± 2.87	<0.001
k-NN	56.82 ± 4.67	<0.001
RF(10)	51.54 ± 0.82	<0.001
RF(100)	69.77 ± 1.87	<0.001
RF(1000)	73.09 ± 1.94	<0.001
p-DNN	87.38 ± 2.37	–

from the speakers, 45 different word data from 30 speakers that are in the different regions were collected and these data were classified with p-DNN with an accuracy of 90.37%. In the experiment to investigate whether the developed system is dependent on the speaker, WAR was obtained 78.50%.

Lastly, studies that were collected objects which were in the robot workspace by a robot arm were examined. In the examined studies, various algorithms, from deep learning methods to Random Sample Consensus, were used as image processing methods (Farag et al., 2019a,b; Huang et al., 2011; Kaymak and Ucar, 2019; Van Delden and Umrysh, 2011). The image processing algorithm used by Huang and his team is similar to the image processing algorithm in the current study but difference from two studies was the objects to be picking in different geometries (Huang et al., 2011). Hence, in this study, an image processing algorithm was developed that finds the orientation of the objects.

The developed software is designed to enable the robot control to be performed by users without knowledge of robotics. A new Turkish language automated speech recognition software was developed and the addition of a wrong pronunciation correction algorithm. A modern industrial robot was obtained that converts voice commands into text data, and senses the environment and performs desired operations. The developed system was tested for different industrial applications such as aligning, picking, drilling, and cleaning.

6. Conclusion

In this paper, a standard industrial robot was enabled by an operator and controlled by natural speech without any prior knowledge, skill or experience of robotics. First of all, speech data were separated into words, and then the features were separated by using the MFCC calculating algorithm. The obtained features were classified by using p-DNN, which has a higher performance than other classification methods

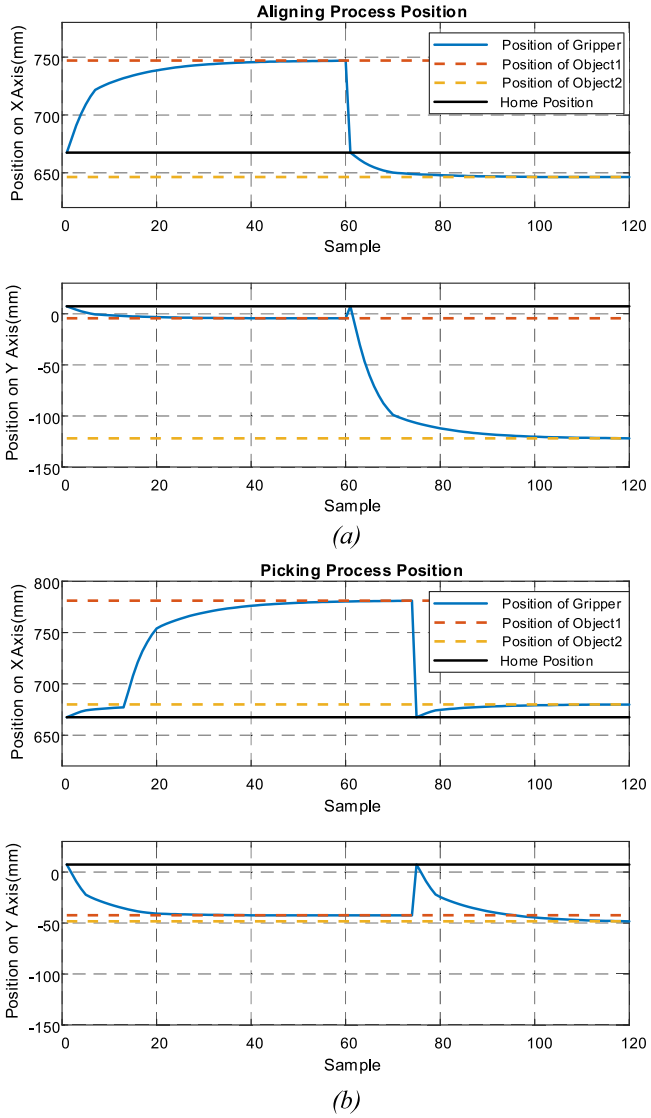


Fig. 20. (a) gripper position during aligning operation, (b) gripper position during picking operation.

as proven in this study. According to the results, the validation accuracy achieved was to 90.37%. In addition, 78.50% WAR value was obtained

in the test performed on speaker who are not in the database. The value of WAR indicates that the system is not dependent on speakers on a large scale. Also, significantly statistical difference between p-DNN and other algorithms was found. There was no significant relationship between the feature size and WAR. This indicates that the system is nonlinear and that a different feature size may work better for another language. After the speech data was translated into words, the misunderstood words were corrected by IMMW.

The arranged text data is first confirmed by the operator and then forwarded to the information screen. If determined word is not deemed to be correct, the robot continues listening. The robot starts to listen after hearing the word "KUKA" (the robot manufacturer's brand name) before every conversation. If the text is confirmed by the operator, the text data is then translated into the format that the robot can understand in the Text Understanding section. If the order given relates to the operations of either aligning or picking, the position and orientation of the objects are determined using object detection algorithm. The centers of the objects are determined using similar methods for the drilling operation. The gripper is routed to positions using a fuzzy controller, and the robot performs then the assigned tasks.

According to the results, it was seen that the gripper approached the reference object with less than 1 mm margin of error. This is considered to be a satisfactory result since the operation was performed with a low-grade camera costing approximately $2.

In future studies, the researchers plan to develop image processing algorithms that can perform more complex industrial processes such as welding. In addition, the speech dataset will be expanded in order to prove that the success rate can be increased accordingly.

Note

This study was performed at Fırat University within the scope of the doctoral thesis titled "The Development of an Artificial Intelligence-Based Self-Programmable Robot Software Using Human–Robot Interaction".

CRediT authorship contribution statement

Mustafa Can Bingol: Conceptualization, Methodology, Software, Validation, Investigation, Writing - original draft, Visualization. **Omur Aydogmus:** Conceptualization, Methodology, Investigation, Writing - original draft, Visualization, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.engappai.2020.103903.

References

Amrouche, A., Debyeche, M., Taleb-Ahmed, A., Michel Rouvaen, J., Yagoub, M.C.E., 2010. An efficient speech recognition system in adverse conditions using the nonparametric regression. Eng. Appl. Artif. Intell. 23, 85–94, https://doi.org/10.1016/j.engappai.2009.09.006.

Bingol, M.C., Aydogmus, O., 2018. The development of an intelligent drilling robot system based on Robot Operating System. In: International Eurasian Conference on Science, Engineering and Technology EurasianSciEnTech, Ankara-Turkey, pp. 808–813.

Bingol, M.C., Aydogmus, O., 2020. Practical application of a safe human–robot interaction software. Ind. Rob. 3, 359–368, https://doi.org/10.1108/IR-09-2019-0180.

Bingol, M.C., Aydoğmuş, Ö., 2019. İnsan-Robot Etkileşiminde İnsan Güvenliği için Çok Kanallı İletişim Kullanarak Evrişimli Sinir Ağı Tabanlı Bir Yazılımının Geliştirilmesi ve Uygulaması. Fırat Üniv. Müh. Bil. Derg. 31, 489–495, https://doi.org/10.35234/fumbd.557590.

Bowyer, S.A., Baena, F.R.Y., 2015. Dissipative control for physical human-robot interaction. IEEE Trans. Robot. 31, 1281–1293, https://doi.org/10.1109/TRO.2015.2477956.

Cakir, M.Y., Sirin, Y., 2018. . speaker independent turkish speech recognition optimization with energy derivates on feature vectors. In: 26th IEEE Signal Processing and Communications Applications Conference, SIU 2018. Izmir, Turkey. pp. 1–4, https://doi.org/10.1109/SIU.2018.8404809.

Çarkı, K., Geutner, P., Schultz, T., 2000. Turkish LVCSR: Towards better speech recognition for agglutinative languages. In: 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Istanbul, Turkey, pp. 1563–1566.

Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., Fraisse, P., 2016. Collaborative manufacturing with physical human–robot interaction. Robot. Comput. Integr. Manuf. 40, 1–13, https://doi.org/10.1016/j.rcim.2015.12.007.

Davis, S.B., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. Speech Signal Process. 28, 357–366, https://doi.org/10.1109/TASSP.1980.1163420.

Dean-Leon, E., Ramirez-Amaro, K., Bergner, F., Dianov, I., Cheng, G., 2018. Integration of robotic technologies for rapidly deployable robots. IEEE Trans. Ind. Inform. 14, 1691–1700, https://doi.org/10.1109/TII.2017.2766096.

Du, G., Chen, M., Liu, C., Zhang, B., Zhang, P., 2018. Online robot teaching with natural human–robot interaction. IEEE Trans. Ind. Electron. 65, 9571–9581, https://doi.org/10.1109/TIE.2018.2823667.

Du, G., Zhang, P., 2015. A markerless human-robot interface using particle filter and kalman filter for dual robots. IEEE Trans. Ind. Electron. 62, 2257–2264, https://doi.org/10.1109/TIE.2014.2362095.

Esfandian, N., Razzazi, F., Behrad, A., 2012. A clustering based feature selection method in spectro-temporal domain for speech recognition. Eng. Appl. Artif. Intell. 25, 1194–1202, https://doi.org/10.1016/j.engappai.2012.04.004.

Farag, M., Ghafar, A.N.A., Alsibai, M.H., 2019a. Real-time robotic grasping and localization using deep learning-based object detection technique. In: 2019 IEEE International Conference on Automatic Control and Intelligent Systems, I2CACIS 2019 - Proceedings. IEEE, Selangor, Malaysia, pp. 139–144, https://doi.org/10.1109/I2CACIS.2019.8825093.

Farag, M., Ghafar, A.N.A., Alsibai, M.H., 2019b. Grasping and positioning tasks for selective compliant articulated robotic arm using object detection and localization: Preliminary results. In: Proceedings - 2019 6th International Conference on Electrical and Electronics Engineering, ICEEE 2019. Istanbul, Turkey. pp. 284–288, https://doi.org/10.1109/ICEEE2019.2019.00061.

Ficuciello, F., Villani, L., Siciliano, B., 2015. Variable impedance control of redundant manipulators for intuitive human-robot physical interaction. IEEE Trans. Robot. 31, 850–863, https://doi.org/10.1109/TRO.2015.2430053.

Grozdić, .Đ.T., Jovičić, S.T., Subotić, M., 2017. Whispered speech recognition using deep denoising autoencoder. Eng. Appl. Artif. Intell. 59, 15–22, https://doi.org/10.1016/j.engappai.2016.12.012.

Hu, Z., Pan, J., Fan, T., Yang, R., Manocha, D., 2019. Safe navigation with human instructions in complex scenes. IEEE Robot. Autom. Lett. 4, 753–760, https://doi.org/10.1109/LRA.2019.2893432.

Huang, R., Cheng, H., Qiu, J., Zhang, J., 2019. Learning physical human-robot interaction with coupled cooperative primitives for a lower exoskeleton. IEEE Trans. Autom. Sci. Eng. 1–9, https://doi.org/10.1109/tase.2018.2886376.

Huang, G., Lin, H., Chen, P., 2011. Robotic arm grasping and placing using edge visual detection system. In: 2011 8th Asian Control Conference, ASCC. IEEE, Kaohsiung, Taiwan, pp. 960–964.

Huang, G.S., Lu, Y.A., 2014. Application of DSP speech synthesis system on service robots. In: CACS 2014 - 2014 International Automatic Control Conference, Conference Digest. IEEE, Kaohsiung, Taiwan. pp. 150–155, https://doi.org/10.1109/CACS.2014.7097179.

Imai, M., Ono, T., Ishiguro, H., 2003. Physical relation and expression : Joint attention for human–robot interaction. IEEE Trans. Ind. Electron 50, 636–643.

Iwata, H., Sugano, S., Member, S., 2005. Human – robot-contact-state identification based on tactile recognition. IEEE Trans. Ind. Electron. 52, 1468–1477.

Jacob, M.G., Wachs, J.P., 2016. Optimal modality selection for cooperative human-robot task completion. IEEE Trans. Cybern. 46, 3388–3400, https://doi.org/10.1109/TCYB.2015.2506985.

Jensen, B., Tomatis, N., Mayor, L., Drygajlo, A., Siegwart, R., 2005. Robots meet humans - interaction in public spaces. IEEE Trans. Ind. Electron. 52, 1530–1546, https://doi.org/10.1109/TIE.2005.858730.

Jothilakshmi, S.Ã., Ramalingam, V., Palanivel, S., 2009. Engineering applications of artificial intelligence speaker diarization using autoassociative neural networks 22, 667–675. https://doi.org/10.1016/j.engappai.2009.01.012.

Kaymak, C., Ucar, A., 2019. Implementation of object detection and recognition algorithms on a robotic arm platform using raspberry pi. In: 2018 International Conference on Artificial Intelligence and Data Processing, IDAP 2018. IEEE, Malatya, Turkey. pp. 1–8, https://doi.org/10.1109/IDAP.2018.8620916.

Kimmel, M., Hirche, S., 2017. Invariance control for safe human–robot interaction in dynamic environments. IEEE Trans. Robot. 33, 1327–1342, https://doi.org/10.1109/TRO.2017.2750697.

Lee, A., Glass, J., 2012. A comparison-based approach to mispronunciation detection. In: 2012 IEEE Work. Spok. Lang. Technol. SLT 2012 - Proc. pp. 238–387, https://doi.org/10.1109/SLT.2012.6424254.

Li, Z., Huang, B., Ye, Z., Deng, M., Yang, C., 2018. Physical human–robot interaction of a robotic exoskeleton by admittance control. IEEE Trans. Ind. Electron. 65, 9614–9624, https://doi.org/10.1109/TIE.2018.2821649.

Li, L., Yan, S., Yu, X., Tan, Y.K., Li, H., 2012. Robust multiperson detection and tracking for mobile service and social robots. IEEE Trans. Syst. Man, Cybern. B 42, 1398–1412, https://doi.org/10.1109/TSMCB.2012.2192107.

Lin, Y., Min, H., Zhou, H., Pei, F., 2018. A human-robot-environment interactive reasoning mechanism for object sorting robot. IEEE Trans. Cogn. Dev. Syst. 10, 611–623, https://doi.org/10.1109/TCDS.2017.2706975.

Mašek, P., Růžička, M., 2014. Speech recognition via STT API for autonomous mobile robot. In: Proceedings of the 16th International Conference on Mechatronics, Mechatronika 2014. Brno, Czech Republic. pp. 594–599, https://doi.org/10.1109/MECHATRONIKA.2014.7018326.

Moriya, T., Tanaka, T., Shinozaki, T., Watanabe, S., Duh, K., 2019. Evolution-strategy-based automation of system development for high-performance speech recognition. IEEE/ACM Trans. Audio, Speech, Lang. Process. 27, 77–88, https://doi.org/10.1109/TASLP.2018.2871755.

Nguyen, D.T., Oh, S.R., You, B.J., 2005. A framework for internet-based interaction of humans, robots, and responsive environments using agent technology. IEEE Trans. Ind. Electron. 52, 1521–1529, https://doi.org/10.1109/TIE.2005.858731.

Oh, S.L., Ng, E.Y., San Tan, R., Acharya, U.R., 2018. Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats. Comput. Biol. Med. 102, 278–287, https://doi.org/10.1016/j.compbiomed.2018.06.002.

Olson, H.F., Belarf, H., 1957. Time compensation for speed of talking in speech recognition machines. IRE Trans. Audio 17, 87–90, https://doi.org/10.1109/TAU.1960.1166250.

Pan, Y., Guo, Z., Yu, H., Chen, G., Huang, S., 2015. Human–robot interaction control of rehabilitation robots with series elastic actuators. IEEE Trans. Robot. 31, 1089–1100, https://doi.org/10.1109/tro.2015.2457314.

Petridis, S., Wang, Y., Ma, P., Li, Z., Pantic, M., 2020. End-to-end visual speech recognition for small-scale datasets. Pattern Recogn. Lett. https://doi.org/10.1016/j.patrec.2020.01.022.

Qu, J., Zhang, F., Wang, Y., Fu, Y., 2019. Human-like coordination motion learning for a redundant dual-arm robot. Robot. Comput. Integr. Manuf. 57, 379–390, https://doi.org/10.1016/j.rcim.2018.12.017.

Rahman, S.M.M., Ikeura, R., 2016. Weight-prediction-based predictive optimal position and force controls of a power assist robotic system for object manipulation. IEEE Trans. Ind. Electron. 63, 5964–5975, https://doi.org/10.1109/TIE.2016.2561879.

Ripley, B.D., 2008. Pattern Recognition and Neural Networks. Cambridge University Press, Cambridge, p. 354.

Sadrfaridpour, B., Wang, Y., 2018. Collaborative assembly in hybrid manufacturing cells: An integrated framework for human-robot interaction. IEEE Trans. Autom. Sci. Eng. 15, 1178–1192, https://doi.org/10.1109/TASE.2017.2748386.

Shahrul Azmi, M.Y., 2016. Malay word pronunciation application for pre-school children using vowel recognition. In: Proceedings - 8th International Conference on U- and E-Service, Science and Technology, UNESST 2015. IEEE, Jeju, South Korea. pp. 57–61, https://doi.org/10.1109/UNESST.2015.25.

Speech Commands Dataset Version 1, 2017. http://download.tensorflow.org/data/speech_commands_v0.01.tar.gz (Accessed 16 June 20).

Stančić, I., Musić, J., Grujić, T., 2017. Gesture recognition system for real-time mobile robot control based on inertial sensors and motion strings. Eng. Appl. Artif. Intell. 66, 33–48, https://doi.org/10.1016/j.engappai.2017.08.013.

Stiefelhagen, R., Ekenel, H.K., Fügen, C., Gieselmann, P., Holzapfel, H., Kraft, F., Nickel, K., Voit, M., Waibel, A., 2007. Enabling multimodal human–robot interaction for the karlsruhe humanoid robot. IEEE Trans. Robot. 23, 840–851, https://doi.org/10.1109/TRO.2007.907484.

Stolcke, A., Tjalve, M., Lopes, C., Candeias, S., Perdigao, F., Proenca, J., 2018. Mispronunciation detection in children's reading of sentences. IEEE/ACM Trans. Audio, Speech, Lang. Process. 26, https://doi.org/10.1109/taslp.2018.2820429, 1-1.

Ting, H.N., Yong, B.F., Mirhassani, S.M., 2013. Self-adjustable neural network for speech recognition. Eng. Appl. Artif. Intell. 26, 2022–2027, https://doi.org/10.1016/j.engappai.2013.06.004.

Tombaloglu, B., Erdem, H., 2017. Türk Dili için SVM Tabanli Konuşma-Metin Dönüştürücü. In: 2017 25th Signal Processing and Communications Applications Conference, SIU 2017. Antalya, Turkey. https://doi.org/10.1109/SIU.2017.7960486.

Tombaloğlu, B., Erdem, H., 2016. MFCC-SVM Tabanli Türkçe Konuşma Tanima Sisteminin Geliştirilmesi. In: 2016 24th Signal Processing and Communication Application Conference, SIU 2016 - Proceedings. Zonguldak, Turkey. pp. 929–932, https://doi.org/10.1109/SIU.2016.7495893.

Turkish Alphabet Pronunciation, 2019. http://mylanguages.org/turkish_alphabet.php (Accessed 16 June 20).

Vafeiadis, A., Votis, K., Giakoumis, D., Tzovaras, D., Chen, L., Hamzaoui, R., 2020. Audio content analysis for unobtrusive event detection in smart homes. Eng. Appl. Artif. Intell. 89, https://doi.org/10.1016/j.engappai.2019.08.020.

Van Delden, S., Umrysh, M.A., 2011. Visual detection of objects in a robotic work area using hand gestures. In: ROSE 2011 - IEEE Int. Symp. Robot. Sensors Environ. Proc. pp. 237–242, https://doi.org/10.1109/ROSE.2011.6058529.

Wang, H., Ren, J., Li, X., 2016. Natural spoken instructions understanding for rescue robot navigation based on cascaded conditional random fields. In: Proceedings - 2016 9th International Conference on Human System Interactions, HSI 2016. IEEE, Portsmouth, UK. pp. 216–222, https://doi.org/10.1109/HSI.2016.7529634.

Wei, Y., Zhao, J., 2016. Designing robot behavior in human robot interaction based on emotion expression. Ind. Rob. 43, 380–389, https://doi.org/10.1108/IR-08-2015-0164.

Video of current work, 2020. https://drive.google.com/file/d/1lDh4iYxAY28cjtZwmm0ViUPIt6WCmPbR (Accessed 16 June 20).

Yang, C., Zeng, C., Liang, P., Li, Z., Li, R., Su, C.Y., 2018. Interface design of a physical human-robot interaction system for human impedance adaptive skill transfer. IEEE Trans. Autom. Sci. Eng. 15, 329–340, https://doi.org/10.1109/TASE.2017.2743000.

Yongda, D., Fang, L., Huang, X., 2018. Research on multimodal human–robot interaction based on speech and gesture. Comput. Electr. Eng. 72, 443–454, https://doi.org/10.1016/j.compeleceng.2018.09.014.

Zeng, C., Yang, C., Chen, Z., Dai, S.L., 2018. Robot learning human stiffness regulation for hybrid manufacture. Assem. Autom. 38, 539–547, https://doi.org/10.1108/AA-02-2018-019.

Zhang, H., Reardon, C., Parker, L.E., 2013. Real-time multiple human perception with color-depth cameras on a mobile robot. IEEE Trans. Cybern. 43, 1429–1441, https://doi.org/10.1109/TCYB.2013.2275291.

Zinchenko, K., Wu, C.Y., Song, K.T., 2017. A study on speech recognition control for a surgical robot. IEEE Trans. Ind. Inform. 13, 607–615, https://doi.org/10.1109/TII.2016.2625818.

Zoughi, T., Homayounpour, M.M., Deypir, M., 2020. Adaptive windows multiple deep residual networks for speech recognition. Expert Syst. Appl. https://doi.org/10.1016/j.eswa.2019.112840.