Autonomous Robots
https://doi.org/10.1007/s10514-022-10073-6

Few-shot re-identification of the speaker by social robots

Pasquale Foggia[1] · Antonio Greco[1] · Antonio Roberto[1] · Alessia Saggese[1] · Mario Vento[1]

Received: 1 June 2021 / Accepted: 27 October 2022

Abstract

Nowadays advanced machine learning, computer vision, audio analysis and natural language understanding systems can be widely used for improving the perceptive and reasoning capabilities of the social robots. In particular, artificial intelligence algorithms for speaker re-identification make the robot aware of its interlocutor and able to personalize the conversation according to the information gathered in real-time and in the past interactions with the speaker. Anyway, this kind of application requires to train neural networks having available only a few samples for each speaker. Within this context, in this paper we propose a social robot equipped with a microphone sensor and a smart deep learning algorithm for few-shot speaker re-identification, able to run in real time over an embedded platform mounted on board of the robot. The proposed system has been experimentally evaluated over the VoxCeleb1 dataset, demonstrating a remarkable re-identification accuracy by varying the number of samples per speaker, the number of known speakers and the duration of the samples, and over the SpReW dataset, showing its robustness in real noisy environments. Finally, a quantitative evaluation of the processing time over the embedded platform proves that the processing pipeline is almost immediate, resulting in a pleasant user experience.

Keywords Social robot · Speaker re-identification · Few-shot learning

1 Introduction

Cognitive robots are intelligent machines able to perform tasks autonomously, without any human control. They have the capability of perception, information processing, decision-making and operation execution like humans (Liu et al., 2017). Cognitive robots have been applied in different application contexts, ranging from precision medicine to manufacturing. We are assisting to a great evolution of robotic applications thanks to the impressive progress of modern artificial intelligence algorithms; social robots have now the capability to perform complex and interactive tasks instead of simple and repetitive actions. The interaction with humans through speech commands or natural language is the main task of *social robots*, with which the humans feel comfortable relating and empathizing (Breazeal, 2002; Maxwell, 2007). Popular applications of social robots include museum guide (Vásquez & Matía, 2020), nurse (Ramachandran & Lim, 2021), autism treatment (Pennisi et al., 2016) and elderly care (Broekens et al., 2009).

While performing a task, it is commonly required for a cognitive robot to identify its interlocutor. This is particularly true for social robots, where it is strongly suggested to maintain a profile of the user (still guaranteeing the compliance with the privacy issues) in order to allow a proper and personalized interaction between the robot and the human. In fact, it has been shown that the users generally prefer that the dialogue with the robot is personalized (Cole et al., 2003).

A common application of social robotics that we may consider as a relevant example is the hotel concierge (López et al., 2013), namely a robot that proactively interacts with customers, recommending surrounding facilities (e.g. restaurants and attractions) and answering frequently asked questions. Within this context, *the ability of the robot to recognize the people who are interacting with itself is crucial*.

✉ Antonio Roberto
aroberto@unisa.it

Pasquale Foggia
pfoggia@unisa.it

Antonio Greco
agreco@unisa.it

Alessia Saggese
asaggese@unisa.it

Mario Vento
mvento@unisa.it

[1] Department of Computer and Electrical Engineering and Applied Mathematics (DIEM), University of Salerno, Fisciano, Italy

Published online: 07 November 2022

As an example, if we think about the recommendation task, knowing the interlocutor identity the robot can suggest facilities based on the stored profile. This means not only that the robot can recommend promising places to visit, but also that it can ask a feedback to incrementally improve the recommendation engine. Another example can be the possibility for the staff of the hotel to send commands to the robot; the recognition of the interlocutor allows to avoid accepting commands by not authorized people.

The main challenge of the robot concierge, and of social robots in general, is that *it has to recognize people met very few times* (often once); thus, there is the need to update the people known by the robot without updating the model (usually a heavy and long task) and with few interactions. The task of identifying a person who already interacted with the robot is called re-identification. In particular, when the number of samples (interactions) for all the people (or some of them) is limited, the problem is referred as few-shot re-identification. In the considered example, the robot concierge has to distinguish customers from the hotel staff and understand whether the person interacting with it is known (a profile is already available) or not. In case the profile is not available, it is then required to build a new profile for that specific interlocutor.

The re-identification task on robotics platform has been often addressed as a face re-identification through the use of RGB cameras (Kviatkovsky et al., 2012; Chen et al., 2015). Unfortunately, *the visual perception may be not reliable enough* when dealing with real scenarios commonly presenting brightness variations, scene occlusions and other disturbances; more importantly, the person interacting with the robot may not be in the field of view of the camera. For this reason, it is a common practice to treat the problem as a speaker re-identification task, i.e. to use the voice for the first identification of a new interlocutor or for the re-identification of a known interlocutor. The task is performed through the use of a microphone, available on social robots due the need to perform Automatic Speech Recognition (ASR) (Nassif et al., 2019). In (Krsmanovic et al., 2006) the STanford AI Robot (STAIR) identifies the speaker to personalize the conversation, improving the user experience (Cole et al., 2003); in case of tasks such as package delivery, it can answer the question ”Who sent you?”. Finally, STAIR is able to accept commands given by authorized people, that are recognized through their voice. In the same way, in (Pleva et al., 2017) and (Guo et al., 2020) the authors propose to use the vocal characteristics of the interlocutor for authentication purposes. Intuitively, the ability of identifying the person who is giving a command enhances the security of the system itself. It is worth to mention the article ”*Amazon's Alexa started ordering people dollhouses after hearing its name on TV*”, [1] in which the system damaged (in this case spending money for useless things) the user following the commands without any type of speaker identification. By using an authentication protocol (based on speaker identification), it would have been possible to avoid this kind of dangers.

The capability of a robot to identify the speaker can be also exploited to perform not perceptible but still not negligible tasks. For instance, with priors knowledge of the speaker identity, it is possible to enhance the speech audio signal focusing exclusively on its voice (suppressing other sounds) (Shi et al., 2020). In this way, it becomes possible to further improve the performance of ASR algorithms (Wang et al., 2019), which are crucial for the Human-Robot interaction (HRI) (Burger et al., 2011).

Speaker identification systems have been adopted in humanoids robots to improve the HRI from different points of view. In (Ji et al., 2007) the ”Wever” robot recognizes the speaker through multiple microphones to offer a natural and familiar interface. Martinson and Lawson (2011) proposed a multimodal audio-visual recognition system on board of the Octavia robot to effectively track the speaker in party-like conversation (i.e. multiple active speakers). In the same way, in (Churamani et al., 2017) the authors equipped the ”NICO” robot with a speaker and face recognition modules to personalize the conversation. They proved that when the robot converses in a personalized way, it is perceived by the interlocutors as more smart and likeable.

An important requirement for a robot is surely the real-time constraint. In the context of conversational social robots, *the interlocutor does not want to wait before receiving an answer from the robot* itself (Greco et al., 2019). Nowadays, modern robotics systems often rely on Deep Learning algorithms to perform their tasks, but they require heavy computation. A common choice is to use cloud-based services or high performance servers to distribute the computation. In (Chen et al., 2011) the ”Robert” and ”Davinci” senior companion robots perform speaker identification and speech recognition through cloud services. . However, the availability of a stable and fast internet connection is not guaranteed for this kind of applications (Du et al., 2017). Moreover, the communication over the network can be slowed down by several external factors (Tanwani et al., 2020). For these reasons, it is important that the computation is performed on board of the robot, by designing a solution that is a good trade-off between accuracy and computational costs (Foggia et al., 2019).

Starting from the above considerations, in this paper, we propose a social robot equipped with an audio processing architecture for few-shot re-identification of the speaker. As a case study for our application, we integrated this module into a real robotic concierge application, deployed on board of the Pepper robot. The proposed architecture has been designed and implemented through the Robotic Operating Systems

[1] https://www.theverge.com/2017/1/7/14200210/amazon-alexa-tech-news-anchor-order-dollhouse.

(ROS),[2] abstracting the specific platform used to acquire the audio stream. Moreover, the algorithm run on the NVIDIA Jetson Xavier embedded system directly mounted on board of the robot, avoiding network latency and removing the internet connection requirement. The proposed system is able to recognize in real time 30 speakers with an accuracy of 95% and 150 speakers with an accuracy of around 91% by considering just 3 voice samples for each speaker. The reference samples have a duration of few seconds and they are acquired during the conversation with the interlocutor. In order to validate the effectiveness of the proposed approach in the wild, we also conducted a robustness evaluation of the proposed systems in noisy environments. Also, we performed a quantitative evaluation of the user experience computed as the latency between the user utterances (i.e. voice samples) and the robot answer. The achieved results confirm the efficiency and the effectiveness of the proposed robotic platform when applied in real crowded environments.

The contribution of this paper can be summarized as follows. (i) We extended the method proposed by Kye et al. (2020) to make it suitable for a real application of few-shot speaker re-identification; in particular, we added the capability of working in an open set configuration by rejecting utterances spoken by unknown speakers and by updating the reference set in order to both improve the performance of the system over the time (for known speakers) and to include unknown speakers in the reference set. (ii) We extensively evaluated the accuracy of the proposed method by varying the number of known speakers, the number of samples per speaker, the duration of the utterance and the background noise, in order to design strategies allowing to maximize and improve its performance and robustness over the time in a social robot application. (iii) We integrated the proposed method on board of a social robot in the ROS framework and quantitatively evaluated the user experience as the time needed by the robot to recognize the user. (iv) We carried out a real experiment of the social robot inside a hotel hall, analysing the effectiveness of the proposed method in updating the reference set.

The paper is organized as follows: in Sect. 2 we details the algorithm, based on deep neural networks, proposed for speaker re-identification. Section 3 describes the robotic platform we have used in our application, and also the software architecture. Finally, the experimentation conducted is reported in Sect. 4, before concluding the paper in Sect. 5.

2 Speaker re-identification algorithm

The problem of identifying the speaker has historically been one of the main challenges of robotics and biometrics verification systems. In recent years, Deep Learning methods for speech analysis emerged, becoming as disruptive as they have been for image analysis. Deep neural networks achieved state-of-the-art performance over practically all large-scale benchmarks for the task of Speaker Identification (Nagrani et al., 2020; Jahangir et al., 2021).

Unfortunately, as mentioned before, a social robot has to recognize people after interacting with them very few times, usually once; therefore, a few speech samples (often only one) are available for each speaker. It is well-known that the training of a deep neural network over a dataset containing few samples for each class (for each speaker in our specific case) typically implies the specialization of the obtained model over the training data. Moreover, the training procedure of a deep neural network is very long and expensive and, therefore, cannot be performed on board of the robot. To address this problem, few-shot learning algorithms have been recently proposed with the aim of adapting the neural networks to samples of unseen classes represented by ”few” examples without the need of re-training (Wang et al., 2020; Vogt et al., 2018).

Starting from the above considerations, we exploited the few-shot learning procedure proposed in (Kye et al., 2020). The ResNet34 backbone (He et al., 2016) has been trained to extract discriminating feature from the speaker utterance. The neural network takes as input the Mel-Spectrogram, a time-frequency representation of the audio signal (Davis & Mermelstein, 1980) which applies a non-linear transformation of the frequency scale, namely the Mel Scale. In particular, the filters are mainly located in the regions of the audio spectrum corresponding to low frequencies, in which the main part of the speech energy is located (Greco et al., 2021b). In our setup, the Mel-Spectrogram is composed of 40 filters, while the sliding window needed for computing the time-frequency representation has a duration of 25 ms and a shift of 10 ms.

The output of a convolutional neural network has a shape proportional to the shape of its input and, therefore, proportional to the duration of the input utterance. For the aim of our system, it is crucial to obtain an utterance embedding of fixed length to compute a similarity function. For this reason, on top of the ResNet34 backbone, a Temporal Average Pooling (TAP) layer is stacked so as to collapse all the features along the feature map dimension (independent from the input size). Finally, the embedding is projected in a smaller vector space by an additional Fully Connected (FC) layer.

The embedding function has been optimized using two loss functions: a prototypical loss and a global classification loss. The former is responsible for reducing the distance between embeddings of samples belonging to the same class:

[2] https://www.ros.org/.

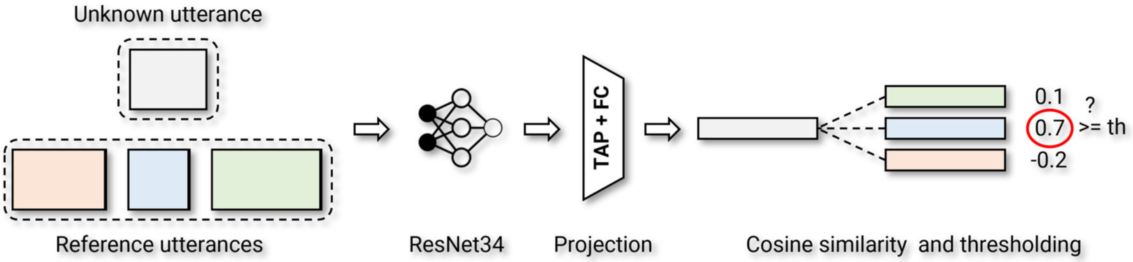


Fig. 1 Speaker re-identification procedure. The embedding of the unknown utterance and those of the reference utterances are computed through a ResNet34 combined with a Temporal Average Pooling (TAP) followed by a Fully Connected (FC) layer. The embedding of the unknown utterance is compared with all the embeddings of the reference set through the cosine similarity function. Finally, the higher similarity is compared with a threshold th to check whether the speaker is known ($\geq th$) or unknown ($< th$); the embedding of an unknown speaker is added to the reference set for a possible future re-identification

$$L_p^\tau = \frac{1}{|Q|} \sum_{(x,y)\in Q} -log\ p(y|x, S) \tag{1}$$

where (x, y) is a couple sample-label belonging to the query set Q in the current batch and S is the support set in the current batch used to compute the classes prototypes. Finally, $p(y|x, S)$ is the probability that a query sample x belongs to the class y given the support set S; the probability is computed applying the softmax function to the distances between the query sample and the classes prototypes. During the training, the support and query sets are randomly chosen within the batch. As suggested in (Kye et al., 2020), the query examples are audio samples of duration between 1 and 2 s, that allow to increase the robustness of the CNN in extracting good representations for short utterances.

The global classification loss helps the network in learning embeddings that are substantially distant for samples of different speakers:

$$L_g^\tau = \frac{1}{|Q| + |S|} \sum_{(x,y)\in Q\cup S} -log\ p(y|x, w) \tag{2}$$

In this case, all the samples from the query and the support set are classified with respect to a global set of prototypes w.

The network has been trained using query samples of very short length (i.e. 1 or 2 s) to enhance the capability of the embedding function in finding discriminating features even during a conversation.

Figure 1 shows how the proposed systems works at inference time. Given as input the sentence, the algorithm computes the fixed-length embedding of the unknown speech sample. Then, the cosine distance similarities between the unknown embedding and all the identities in the reference set is considered. The cosine similarity measure is computed as the cosine of the angle between the two vectors to compare:

$$cos(\theta) = \frac{A \cdot B}{||A|| \cdot ||B||} \tag{3}$$

The output values are in the range [-1,1], where 1 represents the maximum similarity, while -1 indicates that the vectors are literally opposite. We adopted the cosine similarity since it is invariant to module variations; this property makes it very suitable for those cases in which there are uniform transformations as, for instance, loudness changes with respect to reference acquisitions.

At this point, we take the maximum mean similarity, in case of more samples for identity, and we compare it with a threshold th in order to decide if the person is already known or not.

We store all the embeddings of known identities in the system memory to avoid useless and heavy computation; each time a new person talks with the robot, the reference set is updated with the new identity. During a conversation, the proposed method extends the reference set by adding samples for the current speaker until reaching three utterances. Furthermore, given the experimental evidence we will detail in Sect. 4.2, the longer is the duration of the utterance, the higher is the accuracy of the system. Thus, once the limit of three utterances per speaker has been reached, the system continuously updates the reference set if an utterance longer than those stored for the current speaker is listened; indeed, in this case the shortest one is replaced by the new one. Also, the reference set is dynamically updated in real time with samples from unknown speakers. Depending on the specific applications at hand, the oldest speakers will be then removed. In our case study of a social robot as virtual concierge, we only maintain the employees as fixed speaker, giving the admin the possibility to manually remove those who are no more working in the hotel, and the other speakers (corresponding to the guests of the hotel) for a configurable period of time (3 days in our case). In the assumption of a hotel with 75 rooms and an average occupancy of 2 people per room, we can estimate 150 people in the database. Following this hypothesis, we can estimate the memory required to store: for each customer, 3 embeddings having 256 elements of type float32 are

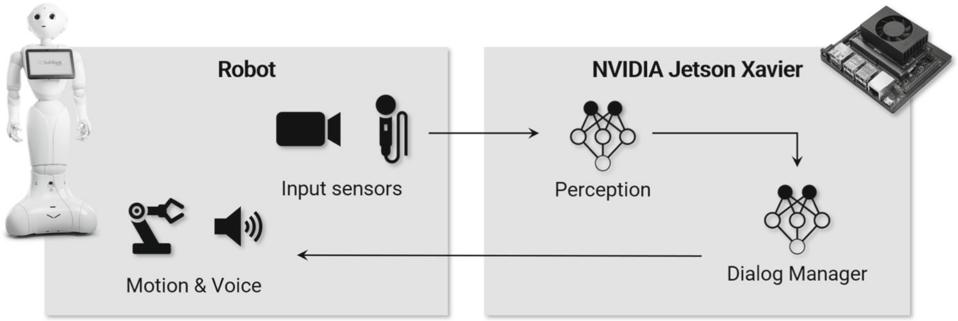


Fig. 2 Overall system architecture. The robot is equipped with sensors, including camera and microphone, and actuators, which allow to speech and move. The raw data from sensors are processed over the NVIDIA Jetson Xavier NX embedded system, mounted on board the robot, to perceive the robot surrounding and to decide the next action to perform

necessary, namely $150*3*256*4B = 450KB$, which is negligible with respect to the memory of the embedded systems available on the market. We can also highlight that, from a computational load point of view, the cosine distance (being a scalar product) is easily parallelizable over GPU architectures. It implies that even the computation of the similarity between the reference set and the speaker embedding can be efficiently managed.

3 Robotic platform setup

In order to evaluate the proposed system, we have designed a robotic platform, which architecture is summarized in Fig. 2. It is devised as a robotic concierge system able to interact with people through natural language and movements. The robot is responsible for exploiting both audio and visual information with the aim of recommending facilities to the hotel's customers. For instance, the robot uses soft-biometrics like emotions to better understand the feedback of the interlocutor and thus, combined with its identity and its stable traits (such as gender and age), to update the related user profile.

In more details, the audio stream from the microphone is analyzed with the aim of identifying the interlocutor (with the algorithms described in the previous section), his position with respect to the robot and the spoken words. On the other hand, the video stream is exploited to more precisely locate the interlocutor face and to recognize its soft-biometrics, both stable (age and gender) and temporary (emotions). Once engaged a conversation with a person, the dialog manager is in charge of processing the recognized sentences with the aim of understanding the intent of the interlocutor and to answer/act accordingly.

The robotic platform adopted is Pepper from SoftBank Robotics. Pepper is a humanoid robot which has been designed for social interaction, with the aim to intrigue and attract people and customers. In the last years, Pepper became a reference platform for both academia and industry, and has been deployed in thousands of homes and schools (Pandey & Gelin, 2018). Also, we equipped Pepper with a RGB-D camera and a microphone array mounted on top of the head, so as to allow the robot to perceive the surrounding environment and people in the scene.

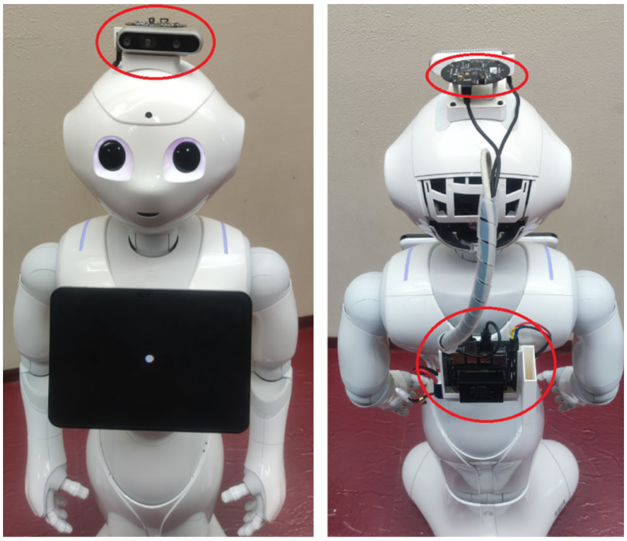


Fig. 3 The proposed robotic system. Pepper has been equipped with a RGB-D camera (circled in red in the image to the left) and a microphone array (circled in red in the image to the right, top mounted on top of the head. The NVIDIA Jetson Xavier NX (circled in red in the image to the right, bottom) is mounted on the back of the robot through a 3D printed support and powered by a lithium battery (Color figure online)

All the smart control of the robot is performed on board of the NVIDIA Jetson Xavier NX embedded system. This choice allows to avoid the use of cloud services and, consequently, the need of an internet connection, which may be sometimes unavailable or more in general not fast enough. The board is connected to the robot in a point-point manner through a RJ45 Ethernet cable. The embedded system communicates with the robot using the official libqi-python library[3] to read data from sensors and to send movement and voice commands. As for the actuators, Pepper allows to use its on-board multilingual speech synthesis system in combination with coordinated human-like movements to interact with people.

The final setup of the robotic system is shown in Fig. 3.

[3] https://github.com/aldebaran/libqi-python.

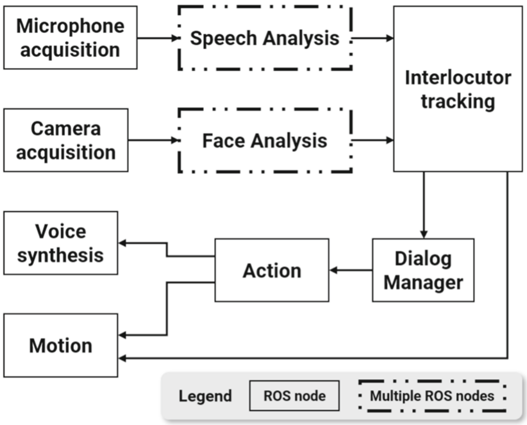


Fig. 4 Software architecture of the proposed cognitive robot, based on ROS. The raw audio-visual data are acquired and analyzed to perform the profiling of the interlocutor (identification, localization, speech recognition). For the sake of readability, all the nodes involved in speech and face analysis have been aggregated into a single software module. The Interlocutor Tracking node filters sentences spoken by other people and send commands to track the interlocutor. The Dialog Manager predicts the next action (speech/motion) to perform according to the intent of the interlocutor, which is finally executed by the Action node

3.1 Software architecture

The software has been designed and developed within the Robotic Operating System (ROS) framework and has been devised and optimized for running on board of the NVIDIA Jetson Xavier NX. Also, each software module has been designed as a ROS node; the architecture is depicted in Fig. 4.

A specific ROS node is dedicated to the acquisition from each of the sensors (i.e. microphone and camera). The data streams are then analysed by independent pipelines to extract high level information about the interlocutor from images and speech. In particular, the analysis of speech signals is comprehensive of the ROS nodes in charge of identify the speaker, locate its position based on the time-difference of arrival of the voice on the microphone array, and recognize the spoken words. In the same way, the ROS nodes responsible for face analysis adopt modern deep learning based face-detectors to identify the exact position of the interlocutor and to recognize its soft-biometrics (Saggese et al., 2019; Greco et al., 2020, 2021c). Based on the information coming from sensors, the Interlocutor Tracking node is able to filter out utterances spoken by other people in the hall and send commands to the robot in order to follow the interlocutor. The Interlocutor Tracking node is crucial to improve the user experience, since it must give the human the feeling of a conversation with a conscious system. The Dialog Manager node receives from the Interlocutor Tracking the sentences spoken by the interlocutor and, based on them, predicts which is the next sentence that the robot has to pronounce. The Dialog Manager exploits the user's attributes (i.e. age, gender,emotion and identity) to improve the recommendation from both the point of view of accuracy and number of questions needed to suggest an useful facility. Finally, the Action node is responsible for voice synthesis and motion, giving the possibility to perform more complex routines with respect to those ones made available by the standard robot library.

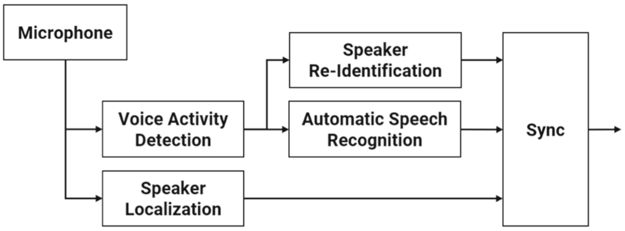


Fig. 5 Nodes of the ROS architecture performing speech analysis. The audio stream is analyzed with the aim to detect the voice and the position of the speaker with respect to the robot. The Speaker Re-Identification and the Automatic Speech Recognition nodes take as input the detected sentence and are able to identify the speaker and transcribe the sentence, respectively. The inferred information are then synchronized and made available to the Interlocutor Tracking node

All the nodes communicate according to the ROS publisher/subscriber paradigm, in order to make available the perceived information as soon as they are acquired. This choice also allows to easily add new ROS nodes without modifying the data flow. In addition, the nodes are able to discard messages if not needed; this is the case, for instance, of nodes that work at a different rate with respect to sensors.

In this paper, as mentioned, we will focus on the Speech Analysis software modules. Its architecture is represented in Fig. 5. Since the focus of this node is on the speech, we may refer indiscriminately to the interlocutor with the term speaker, namely the person who is talking with the robot. The audio stream is acquired and made available to the Voice Activity Detection and to the Speaker Localization nodes. The former allows to filter only sounds attributable to the human voice, in order to provide such spoken sentences to the Speaker Re-Identification node, which identifies the speaker, and to the Automatic Speech Recognition nodes, which recognize the spoken words; the latter is able to localize the direction of arrival of the voice. All these information related to the speaker are synchronized in order to make them simultaneously available for the dialog manager node.

The details about the algorithm considered for the Speaker Re-Identification node has been the subject of the previous section.

4 Experiments

In order to validate the performance of the proposed system we conducted three experiments, each one with a different aim. First, (i) we analyzed the speaker re-identification capability of the system by varying the number of samples per speaker, the number of known speakers and the duration of the voice samples; (ii) we evaluated the robustness in different crowded environments characterized by an increasing noise level; (iii) we performed a quantitative evaluation of the user experience measured as the time needed to the system to identify the speaker once the utterance has been detected.

We trained the CNN architecture using the Stochastic Gradient Descent algorithm with the Nesterov momentum as suggested in (Kye et al., 2020). The learning rate has been set to 0.1 and decreased by a factor 10 when the loss on the validation set does not improve for three consecutive epochs. Furthermore, the training has been regularized through a weight decay equal to 0.0001.

In the following sections we describe the reference datasets and report the main results from the three experiments.

4.1 Dataset

We performed the quantitative evaluation of the system on the VoxCeleb1 (Nagrani et al., 2017) speaker identification dataset. It is one of the most used datasets, characterized by uncontrolled recording conditions, commonly said ”in-the-wild”. VoxCeleb 2 dataset instead has been used for training the neural network. Of course, the test set of the VoxCeleb1 dataset does not include samples of speakers used for training the proposed network.

The dataset, which details are reported in Table 1, contains around 153k utterances acquired from 1251 speakers. The voice samples are acquired with an automated pipeline on YouTube videos and, therefore, without any bias due to environmental background and recording devices. All the audio samples have been re-sampled at 16 KHz.

To evaluate the robustness of the proposed systems in real environments, we performed an additional evaluation over the Speaker Recognition in the Wild (SpReW) dataset (Roberto et al., 2019). SpReW is a voice dataset acquired in four different environments (C00, C01, W01, W02) characterized by different noise levels. It is particularly suitable to evaluate speaker identification systems for social robotics applications, since the utterances have been acquired in very crowded environments. As proposed in (Roberto et al., 2019), the data of the scenario C00 are used as reference samples, while the ones of C01, W01 and W02 for testing purposes. Therefore, there are at most 10 reference samples for each speaker to recognize, in order to simulate realistic operating conditions. The details of the dataset are reported in Table 2.

Table 1 Statistics of the VoxCeleb 1 dataset, in terms of number of speakers, number of male speakers, number of utterances, average number of utterances per speaker and average duration of the utterances

VoxCeleb 1	
Speakers	1251
Male speakers	690
Utterances	153,516
Avg utterances per speaker	116
Avg duration of utterances (s)	8.2

Table 2 Statistics of the SpReW dataset, detailed for each scenario, in terms of number of speakers, number of utterances, average number of utterances per speaker and average noise level

SpReW	Scenario			
	C00	C01	W01	W02
Speakers	20	20	20	20
Utterances	200	104	94	95
Avg utterances per speaker	10	5	5	5
Avg noise level [dB]	−35	−30	−20	−17

Table 3 One-Shot Speaker Re-Identification accuracy in the open-set configuration by varying the number of known speakers and the duration of the utterances (1 s, 3 s, 5 s)

Known speakers	Accuracy (%)		
	1 s	3 s	5 s
30	81.03	90.63	92.47
60	76.60	88.09	90.42
90	73.40	86.36	89.06
120	70.72	84.87	87.86
150	68.29	83.62	86.87

4.2 Speaker re-identification results

In this section we report the results of the proposed speaker re-identification algorithm over VoxCeleb1. To reproduce the real conditions in which the robot may interact with an unknown speaker, we setup the experiment in an open-set configuration; the speaker to identify may not be known and a threshold is defined to reject unknown speakers. The evaluation has been performed by varying the number of samples per speaker (one-shot and three-shot), the number of known speakers (30, 60, 90, 120, 150) and the duration of the utterances (1 s, 3 s, 5 s).

We reported in Tables 3 and 4 the performance of the proposed method in terms of accuracy, i.e. the ratio between the number of correctly identified speakers and the total number of speakers to identify. The threshold used for rejecting unknown speakers has been computed in order to obtain the Equal Error Rate (EER), i.e. the threshold which achieves a False Acceptance Rate (percentage of not rejected unknown people), equal to the False Rejection Rate (percentage of rejected known people).

Table 4 Three-Shots Speaker Re-Identification accuracy in the open-set configuration by varying the number of known speakers and the duration of the utterances (1 s, 3 s, 5 s)

Known speakers	Accuracy (%)		
	1 s	3 s	5 s
30	84.28	93.48	95.00
60	80.53	91.70	93.66
90	77.85	90.47	92.76
120	75.61	89.37	91.92
150	73.61	88.46	91.24

Table 5 One-Shot Speaker Identification Equal Error Rate (EER) and MisClassification rate (MC) in the open-set configuration by varying the number of known speakers and the duration of the utterances (1 s, 3 s, 5 s)

Known speakers	EER (%)			MC (%)		
	1 s	3 s	5 s	1 s	3 s	5 s
30	18.48	9.24	7.45	4.90	1.31	0.85
60	22.03	11.54	9.34	6.87	1.87	1.17
90	24.15	12.94	10.49	8.16	2.33	1.49
120	25.61	14.05	11.44	9.18	2.70	1.75
150	26.69	14.86	12.13	10.04	3.05	1.99

Let us firstly focus on the one-shot setup (Table 3), to evaluate the impact of the number of known speakers and the duration of the utterance on the performance of the proposed system. It is possible to note that the accuracy of the method is inversely proportional to the number of known speakers; by adding 30 known speakers to the reference set, we can note a linear decrease of the accuracy with a factor of around 3% with utterances of 1 s (accuracy from 81 to 68% with 30 and 150 known speakers) and about 1% with longer utterances (accuracy from 90 to 83% and from 92 to 87% with utterances of 3 and 5 s). This result gives an idea of the good scalability of the proposed method with respect to the number of speakers, but also suggests that the collection of longer utterances in the reference set can further reduce the decrease of the accuracy. Indeed, we notice that with utterances longer than 1 s the accuracy increases of more than 10% (from 81 to 91% and 93% with 30 known speakers, from 68 to 83% and 87% with 150 known speakers). This means that the neural network is able to extract more discriminant features with longer samples of the voice of the speaker. According to these results and considering that the size of the embedding is independent on the duration of the utterance, in our application we decided to always store the longest utterances of the speaker in the reference set; this choice allows to improve the performance of the system over the time.

Another way to evolve the system is the collection of more voice samples for each speaker during the conversation. We can see from Table 4 that with just three utterances for each known speaker we are able to substantially increase the accuracy and the robustness of the proposed method. In fact, in this case we can note an absolute 2–5% increase of the accuracy and a better scalability with respect to the number of known speakers, namely a factor of less than 3% with utterances of 1 s (accuracy from 84 to 73% with 30 and 150 known speakers) and less than 1% with longer utterances (accuracy from 93 to 88% and from 95 to 91% with utterances of 3 and 5 s). The best accuracy of 95% is achieved with 30 known speakers and utterances of 5 s. Starting from these observations, in our application we decided to collect the longest three utterances for each speaker, when available.

For an in-depth analysis, we also analysed the errors of the method in terms of EER and misclassification rate (MC), namely the percentage of known people that have been recognized with a wrong identity label; these results are reported in Tables 5 and 6.

The results of this experiment allows to renew the considerations about the impact of the number of known speakers and the duration of the utterances on the performance. In particular, with utterances longer than 1 s we appreciate a reduction of the EER of at least 9% (with a peak of around 18% in the one-shot setup with 30 known speakers); this evidence confirms the validity of the choice to store the longest utterances for each speaker. Similarly, the MC decreases reaching a minimum value of 0.85% in the same setup. From both the one-shot and three-shot settings results it is evident that the task of determining if a voice has been already heard is harder than distinguish between known people. Nevertheless, considering a three-shot setup, the EER decreases from 23.05 to 10.82% and 8.34% increasing the duration of the utterances to 3 and 5 s respectively, when dealing with a reference set of 150 speakers. The comparison of these results with those ones obtained in the one-shot setup confirms that the use of more reference utterances increases the robustness of the method. In particular, the three-shot configuration improves the performance of around 2–5% and 0.5–2% considering the EER and the MC respectively.

To prove the effectiveness of our method we reproduced the one-shot setup with 150 speakers and utterances of duration equal to 1 s using the well-known x-vectors (Snyder et al., 2018). The x-vector model obtained an EER equal to 44.73%, i.e. 18.04% higher w.r.t. the proposed model, while using embeddings of double size (i.e. 512 hidden units). This result can be motivated by the fact that the proposed method has been optimized to deal with short and variable-length utterances. This further analysis confirms the effectiveness of the proposed approach, even when compared with state of the art algorithms.

Table 6 Three-Shot Speaker Identification Equal Error Rate (EER) and MisClassification rate (MC) in the open-set configuration by varying the number of known speakers and the duration of the utterances (1 s, 3 s, 5 s)

Known speakers	EER (%)			MC (%)		
	1 s	3 s	5 s	1 s	3 s	5 s
30	15.41	6.46	4.97	3.05	0.56	0.32
60	18.59	8.13	6.24	4.40	0.86	0.49
90	20.57	9.21	7.06	5.27	1.06	0.61
120	21.98	10.12	7.78	6.03	1.27	0.74
150	23.05	10.82	8.34	6.68	1.44	0.83

Table 7 One-Shot and Three-Shot speaker identification accuracy on the SpReW dataset

Scenario	Accuracy (%)	
	1-shot	3-shot
C01	98.65	100.00
W01	98.08	100.00
W02	97.47	99.78
Total	98.08	99.93

The results have been computed in different three environments, namely, C01, W01 and W02, characterized by an increasing noise level

4.3 Robustness evaluation in noisy environments

In real scenarios the robot must identify the speaker in different environmental conditions, which may be characterized by various sources of background noise (Roberto et al., 2019; Greco et al., 2021a). In order to evaluate the robustness of the proposed method in such challenging conditions, we perform an experiment over the SpReW dataset, which includes in the test scenarios (C01, W01, W02) samples acquired in scenarios characterized by different noise levels. The results of this experiment are reported in Table 7.

The proposed method demonstrated an impressive robustness in these noisy environments. In fact, the overall accuracy is over 98% for the one-shot setup and almost 100% in the three-shot setup. The very small variance between the worst scenario W02 (97.47% and 99.78%) and the best scenario C01 (98.65% and 100%) is a clear evidence of the generalization capability of the proposed method in noisy environments.

4.4 Quantitative user experience evaluation

The quality of the user experience does not depend only on the re-identification accuracy of the algorithm, but also on the responsiveness of the robot; the speaker would receive an answer as soon as possible and this response time may perceived by the speaker as a possible delay in the interaction. To this aim, we evaluated the processing time required by the algorithm over various NVIDIA Jetson embedded systems that may be equipped by the robot, namely the models Nano, TX2 and Xavier NX. In particular, we measured the average time needed by the robot to recognize the speaker after the acquisition of the first utterance, by varying the duration of the utterance; this is necessary since the processing time increases proportionally with the duration of the utterance. The results of this experiment are reported in Table 8.

Table 8 Processing time in seconds required by the proposed algorithm over three different NVIDIA Jetson embedded devices: Nano, TX2 and Xavier NX

Device	Processing time (s)		
	1 s	3 s	5 s
Nano	0.272	0.924	1.536
TX2	0.124	0.442	0.714
Xavier NX	0.098	0.238	0.384

The processing time increases proportionally with the duration of the utterance (1 s, 3 s, 5 s), but it is around 1.5 s in the worst case

The Xavier NX model, chosen for our setup, allows to receive a response in less than a half second in the worst case (0.384 s); it means that the speaker is recognized immediately even if the spoken utterance is quite long. The TX2 and the Nano models have less processing power, but the response time is in the worst case 0.715 s and 1.536 s, respectively; this is an interesting result, since a fast response can be obtained even with cheaper embedded devices. We can conclude that the proposed method allows to obtain a responsive system.

4.5 HRI evaluation

A further experiment has been conducted in a real environment: indeed, our social robot was placed into a hotel hall and interacted with 20 people about topics related to the hotel services (for instance, the breakfast time or the wifi password). The interactions between the robot and the customers have been recorded in different hours in order to avoid a bias in the obtained sentences. We collected in total 240 sentences.

Starting from these samples, (i) we computed different statistics about the duration of the input utterances (such as minimum and average duration) and (ii) we validated the performance of the system in terms of accuracy of the predictions and percentage of corrupted prototypes in the reference set (i.e. the prototype belonging to another person, both known or unknown).

As for (i), we have computed the density distribution through the gaussian kernel density estimation method. The normalized histogram of the duration of the input utterances and the related density distribution (represented through the solid green line) are depicted tn Fig. 6.

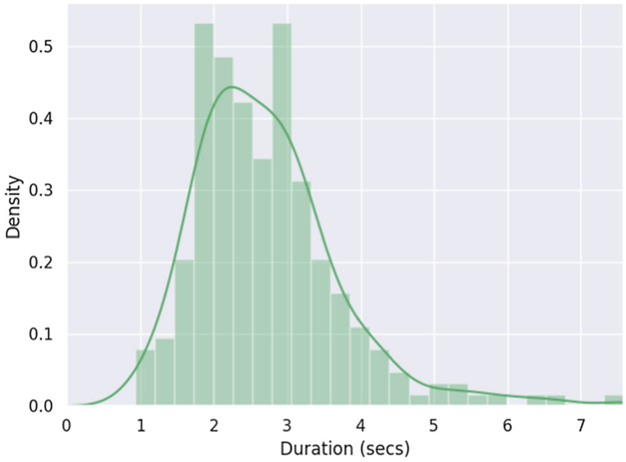


Fig. 6 Normalized histogram of the duration of the utterances. The solid green line represents the estimated density distribution of the audio length for the considered application context (Color figure online)

We can note that the minimum duration of the audio samples is equal to 0.95 s. It means that the performance corresponding to 1-s utterances (reported in Table 4) represents the lower bound of the proposed system in the considered application context. Moreover, we can also observe that 50% of the utterances have a duration in the range (2.0, 3.2) s, with a mean of 2.73 s. This result further validates the benchmark carried out in previous sections.

The proposed system achieved an accuracy of 97.08%, corresponding to 7 errors over 240 interactions. Among these errors, 4 utterances have been wrongly rejected while the remaining sentences have been misclassified. The first type of error did not affect the perceived performance, since the robot can ask the name of the user through the dialogue manager module and, therefore, recover the person, even if already known. On the other hand, the latter requires the user to repeat the sentence if the conversation started correctly.

Since we store 3 prototypes per speaker, during the experiment we started from 60 prototypes and then we updated the reference set through the proposed policy. This update happened 84 times, and just once the reference set has been wrongly updated with a misclassified sample. Anyway, this wrong prototype did not affect the following predictions (for both known and unknown people), thanks to the fact that we use three prototypes per speaker and that the prediction is based on the mean of the cosine similarities w.r.t. each prototype (as described in Sect. 2).

5 Conclusions

The proposed social robot, equipped with a microphone sensor and a smart deep learning algorithm for few-shot speaker re-identification, demonstrated a remarkable re-identification accuracy, a notable robustness to strong background noise and the capability to run in real time over an embedded platform. The experiments performed over the VoxCeleb1 dataset by varying the number of samples per speaker, the number of known speakers and the duration of the voice samples confirmed the validity of the design choices; the method is effective and its performance may improve over the time, by collecting new data from the speakers. Moreover, the efficiency demonstrated on embedded devices with limited resources and cost makes the proposed solution ready for a real social robotics application.

Funding Open access funding provided by Universitá degli Studi di Salerno within the CRUI-CARE Agreement.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Breazeal, C. (2002). *Designing sociable robots*. MIT Press.

Broekens, J., Heerink, M., & Rosendal, H. (2009). Assistive social robots in elderly care: A review. *Gerontechnology, 8*(2), 94–103. https://doi.org/10.4017/gt.2009.08.02.002.00.

Burger, B., Ferrané, I., Lerasle, F., & Infantes, G. (2011). Two-handed gesture recognition and fusion with speech to command a robot. *Autonomous Robots, 32*(2), 129–147. https://doi.org/10.1007/s10514-011-9263-y.

Chen, D., Yuan, Z., Hua, G., Zheng, N., & Wang, J. (2015). Similarity learning on an explicit polynomial kernel feature map for person re-identification. In *2015 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1565–1573). IEEE. https://doi.org/10.1109/cvpr.2015.7298764

Chen, Y. Y., Wang, J. F., Lin, P. C., Shih, P. Y., Tsai, H. C., & Kwan, D. Y. (2011). Human-robot interaction based on cloud computing infrastructure for senior companion. In *TENCON 2011–2011 IEEE region 10 conference* (pp. 1431–1434). IEEE.

Churamani, N., Anton, P., Brügger, M., Fließwasser, E., Hummel, T., Mayer, J., Mustafa, W., Ng, H. G., Nguyen, T. L. C., & Nguyen, Q. et al. (2017) The impact of personalisation on human-robot interaction in learning scenarios. In *Proceedings of the 5th international conference on human agent interaction* (pp. 171–180).

Cole, R., Vuuren, S. V., Pellom, B., Hacioglu, K., Ma, J., Movellan, J., et al. (2003). Perceptive animated interfaces: First steps toward a new paradigm for human-computer interaction. *Proceedings of the IEEE, 91*(9), 1391–1405. https://doi.org/10.1109/jproc.2003.817143.

Davis, S., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing, 28*(4), 357–366. https://doi.org/10.1109/tassp.1980.1163420.

Du, Z., He, L., Chen, Y., Xiao, Y., Gao, P., & Wang, T. (2017). Robot cloud: Bridging the power of robotics and cloud computing. *Future Generation Computer Systems, 74,* 337–348. https://doi.org/10.1016/j.future.2016.01.002.

Foggia, P., Greco, A., Percannella, G., Vento, M., & Vigilante, V. (2019). A system for gender recognition on mobile robots. In *Proceedings of the 2nd international conference on applications of intelligent systems—APPIS '19* (pp. 1–6). ACM Press. https://doi.org/10.1145/3309772.3309781

Greco, A., Roberto, A., Saggese, A., Vento, M., Vigilante, V. (2019). Emotion analysis from faces for social robotics. In *2019 IEEE international conference on systems, man and cybernetics (SMC)* (358–364). IEEE. https://doi.org/10.1109/smc.2019.8914039

Greco, A., Saggese, A., Vento, M., & Vigilante, V. (2020). A convolutional neural network for gender recognition optimizing the accuracy/speed tradeoff. *IEEE Access, 8,* 130771–130781. https://doi.org/10.1109/access.2020.3008793.

Greco, A., Roberto, A., Saggese, A., & Vento, M. (2021). Denet: A deep architecture for audio surveillance applications. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-020-05572-5.

Greco, A., Roberto, A., Saggese, A., Vento, M. (2021b) Which are the factors affecting the performance of audio surveillance systems? In *2020 25th international conference on pattern recognition (ICPR)* (pp. 7876–7883). IEEE. https://doi.org/10.1109/icpr48806.2021.9412573.

Greco, A., Saggese, A., Vento, M., & Vigilante, V. (2021). Effective training of convolutional neural networks for age estimation based on knowledge distillation. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-021-05981-0.

Guo, Y., Xu, W., Pradhan, S., Bravo, C., & Ben-Tzvi, P. (2020). Integrated and configurable voice activation and speaker verification system for a robotic exoskeleton glove. In *International design engineering technical conferences and computers and information in engineering conference, American Society of Mechanical Engineers* (Vol. 83990, p. V010T10A043).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 770–778). IEEE. https://doi.org/10.1109/cvpr.2016.90

Jahangir, R., Teh, Y. W., Nweke, H. F., Mujtaba, G., Al-Garadi, M. A., & Ali, I. (2021). Speaker identification through artificial intelligence techniques: A comprehensive review and research challenges. *Expert Systems with Applications, 171,* 114591. https://doi.org/10.1016/j.eswa.2021.114591.

Ji, M., Kim, S., Kim, H., Kwak, K. C., & Cho, Y. J. (2007). Reliable speaker identification using multiple microphones in ubiquitous robot companion environment. In *RO-MAN 2007-The 16th IEEE international symposium on robot and human interactive communication* (pp. 673–677). IEEE.

Krsmanovic, F., Spencer, C., Jurafsky, D., Ng, A. Y. (2006). Have we met? MDP based speaker ID for robot dialogue. In *INTERSPEECH 2006—ICSLP, ninth international conference on spoken language processing, Pittsburgh, PA, USA*, September 17–21, 2006, ISCA. http://www.isca-speech.org/archive/interspeech_2006/i06_1193.html.

Kviatkovsky, I., Adam, A., & Rivlin, E. (2012). Color invariants for person reidentification. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(7), 1622–1634. https://doi.org/10.1109/tpami.2012.246.

Kye, S. M., Jung, Y., Lee, H. B., Hwang, S. J., & Kim, H. (2020). Meta-learning for short utterance speaker recognition with imbalance length pairs. In *Interspeech 2020, ISCA* (pp. 2982–2986). https://doi.org/10.21437/interspeech.2020-1283

Liu, Y., Tian, Z., Liu, Y., Li, J., Fu, F., & Bian, J. (2017). Cognitive modeling for robotic assembly/maintenance task in space exploration. In *Advances in neuroergonomics and cognitive engineering* (pp. 143–153). Springer. https://doi.org/10.1007/978-3-319-60642-2_13

López, J., Pérez, D., Zalama, E., & Gómez-García-Bermejo, J. (2013). BellBot: A hotel assistant system using mobile robots. *International Journal of Advanced Robotic Systems, 10*(1), 40. https://doi.org/10.5772/54954.

Martinson, E., & Lawson, W. (2011). Learning speaker recognition models through human-robot interaction. In *2011 IEEE international conference on robotics and automation* (pp. 3915–3920). IEEE.

Maxwell, B. A. (2007). Building robot systems to interact with people in real environments. *Autonomous Robots, 22*(4), 353–367. https://doi.org/10.1007/s10514-006-9020-9.

Nagrani, A., Chung, J. S., Zisserman, A. (2017). VoxCeleb: A large-scale speaker identification dataset. In *Interspeech 2017, ISCA*. https://doi.org/10.21437/interspeech.2017-950

Nagrani, A., Chung, J. S., Huh, J., Brown, A., Coto, E., Xie, W., McLaren, M., Reynolds, D. A., & Zisserman, A. (2020). Voxsrc 2020: The second voxceleb speaker recognition challenge. Preprint arXiv:2012.06867

Nassif, A. B., Shahin, I., Attili, I., Azzeh, M., & Shaalan, K. (2019). Speech recognition using deep neural networks: A systematic review. *IEEE Access, 7,* 19143–19165. https://doi.org/10.1109/access.2019.2896880.

Pandey, A. K., & Gelin, R. (2018). A mass-produced sociable humanoid robot: Pepper: The first machine of its kind. *IEEE Robotics & Automation Magazine, 25*(3), 40–48. https://doi.org/10.1109/mra.2018.2833157.

Pennisi, P., Tonacci, A., Tartarisco, G., Billeci, L., Ruta, L., Gangemi, S., & Pioggia, G. (2016). Autism and social robotics: A systematic review. *Autism Research, 9*(2), 165–183. https://doi.org/10.1002/aur.1527.

Pleva, M., Juhar, J., Cizmar, A., Hudson, C., Carruth, D. W., & Bethel, C. L. (2017). Implementing english speech interface to jaguar robot for swat training. In *2017 IEEE 15th international symposium on applied machine intelligence and informatics (SAMI)* (pp. 000105–000110). IEEE.

Ramachandran, B. R. N., & Lim, J. C. (2021). User validation study of a social robot for use in hospital wards. In *Companion of the 2021 ACM/IEEE international conference on human-robot interaction* (pp. 215–219). ACM. https://doi.org/10.1145/3434074.3447162.

Roberto, A., Saggese, A., & Vento, M. (2019). A challenging voice dataset for robotic applications in noisy environments. In *Computer analysis of images and patterns* (pp. 354–364). Springer. https://doi.org/10.1007/978-3-030-29891-3_31

Saggese, A., Vento, M., & Vigilante, V. (2019). MIVIABot: A cognitive robot for smart museum. In *Computer analysis of images and patterns* (pp. 15–25). Springer. https://doi.org/10.1007/978-3-030-29888-3_2

Shi, Y., Huang, Q., & Hain, T. (2020). Speaker re-identification with speaker dependent speech enhancement. In *Interspeech 2020, ISCA*. https://doi.org/10.21437/interspeech.2020-1772

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-vectors: Robust dnn embeddings for speaker recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 5329–5333). IEEE.

Tanwani, A. K., Anand, R., Gonzalez, J. E., & Goldberg, K. (2020). RILaaS: Robot inference and learning as a service. *IEEE Robotics and Automation Letters, 5*(3), 4423–4430. https://doi.org/10.1109/lra.2020.2998414.

Vásquez, B. P. E. A., & Matía, F. (2020). A tour-guide robot: Moving towards interaction with humans. *Engineering Applications of Artificial Intelligence, 88,* 103356. https://doi.org/10.1016/j.engappai.2019.103356.

Vogt, D., Stepputtis, S., Jung, B., & Amor, H. B. (2018). One-shot learning of human–robot handovers with triadic interaction meshes. *Autonomous Robots, 42*(5), 1053–1065. https://doi.org/10.1007/s10514-018-9699-4.

Wang, Q., Muckenhirn, H., Wilson, K., Sridhar, P., Wu, Z., Hershey, J. R., Saurous, R. A., Weiss, R. J., Jia, Y., & Moreno, I. L. (2019). VoiceFilter: Targeted voice separation by speaker-conditioned spectrogram masking. In *Interspeech 2019, ISCA* (pp. 2728–2732). https://doi.org/10.21437/interspeech.2019-1101

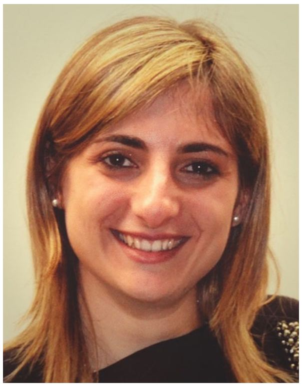
Wang, Y., Yao, Q., Kwok, J. T., & Ni, L. M. (2020). Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys, 53*(3), 1–34. https://doi.org/10.1145/3386252.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Pasquale Foggia Pasquale Foggia received the Ph.D. degree in electronic and computer engineering from University of Naples Federico II, Italy, in 1999. He is currently Full Professor of computer science with the University of Salerno, Italy. His current research interests include basic methodologies and applications in the fields of computer vision and pattern recognition. He is the author of several research papers on these subjects. Dr. Foggia has been a member of the International Association for Pattern Recognition (IAPR) and has been involved in the activities of the IAPR Technical Committee 15 (Graph-Based Representations in Pattern Recognition) since 1997. In 2016 he is elected chairman of the IAPR technical committee 15 on graph-based representations in pattern recognition.

Antonio Greco Antonio Greco received the Ph.D. degree in computer science and computer engineering from the University of Salerno in 2018. He is currently an Assistant Professor with the University of Salerno. His research interests include computer vision and machine learning techniques for video surveillance applications. He serves as a referee for many journals and international conferences.

Antonio Roberto Antonio Roberto received the degree (cum laude) in computer engineering from the University of Salerno in December 2018, where he is currently pursuing the Ph.D. degree. His research interests include audio analysis and machine learning techniques for intelligent robotics and spoken dialogue systems.



Alessia Saggese Alessia Saggese (Member, IEEE) received the Ph.D. degree in computer engineering from the University of Salerno, Italy, and the University of Caen, France, in 2014. She is currently an Associate Professor with the University of Salerno. Her research interests include basic methodologies and applications in computer vision and pattern recognition. She has been a member of the International Association for Pattern Recognition Technical Committee 15 on Graph-Based Representations in Pattern Recognition since 2012.

Mario Vento Mario Vento received the Ph.D. degree in computer engineering from the University of Napoli ”FedericoII” in 1989. He is currently the Vice Rector of the University of Salerno, Italy, where he is also a Full Professor of computer engineering and artificial intelligence. He is also the Coordinator of the Artificial Vision Laboratory. His research interests include real-time video analysis and interpretation for video surveillance applications and cognitive robotics, classification techniques, either statistical, syntactic and structural, and exact and inexact graph matching. Dr. Vento is the Fellow Scientist of the International Association Pattern Recognition (IAPR). He has served as the Chairman of the IAPR Technical Committee 15 on Graph-Based Representation in Pattern Recognition from 2002 to 2006. He is also an Associate Editor of the Pattern Recognition journal.