Multimedia Tools and Applications (2022) 81:3351–3368
https://doi.org/10.1007/s11042-021-11112-7

1188: ARTIFICIAL INTELLIGENCE FOR PHYSICAL AGENTS

Optimization and improvement of a robotics gaze control system using LSTM networks

Jaime Duque Domingo[1] · **Jaime Gómez-García-Bermejo**[1,2] · **Eduardo Zalama**[1,2]

Received: 26 January 2021 / Revised: 5 May 2021 / Accepted: 28 May 2021 /
Published online: 8 July 2021

Abstract
Gaze control represents an important issue in the interaction between a robot and humans. Specifically, deciding who to pay attention to in a multi-party conversation is one way to improve the naturalness of a robot in human-robot interaction. This control can be carried out by means of two different models that receive the stimuli produced by the participants in an interaction, either an on-center off-surround competitive network or a recurrent neural network. A system based on a competitive neural network is able to decide who to look at with a smooth transition in the focus of attention when significant changes in stimuli occur. An important aspect in this process is the configuration of the different parameters of such neural network. The weights of the different stimuli have to be computed to achieve human-like behavior. This article explains how these weights can be obtained by solving an optimization problem. In addition, a new model using a recurrent neural network with LSTM layers is presented. This model uses the same set of stimuli but does not require its weighting. This new model is easier to train, avoiding manual configurations, and offers promising results in robot gaze control. The experiments carried out and some results are also presented.

Keywords Gaze control · Gaze engagement · Humanoid robot · ROS · HRI · Competitive network · Computer vision · Deep neural network · Recurrent neural network · RNN · LSTM

1 Introduction

Gaze control represents an important discipline in the development of intelligent social robots so as to achieve higher assessments of a robot's comprehension and naturalness in *Human–Robot Interaction* (HRI) [24]. When robots behave like a person, humans feel more comfortable. One approach to this problem is to use a competitive neural network that

✉ Jaime Duque Domingo
jaiduq@cartif.es

[1] CARTIF Foundation, División de Sistemas Industriales y Digitales, Parque Tecnológico de Boecillo, 47151, Valladolid, Spain

[2] ITAP-DISA, University of Valladolid, Pl. Santa Cruz 8, 47002, Valladolid, Spain

receives different stimuli and returns a stable determination of the focus of attention to be followed with the robot's eyes. One of the advantages of this approach is that the response of the competitive network is smooth and stable, avoiding erratic behavior. This approach also maintains, in the robot's memory, information about people who have previously interacted with it, regardless of whether they have left the *Robot's Field of View* (FOV). Different factors are taken into account: the human's gaze, who is speaking, pose, proxemics, *Visual Focus of Attention* (VFOA), hoarding conversation, habituation, etc. These factors produce stimuli that create dynamic behavior, giving interlocutors the feeling that they are talking to another human. One problem that arises, however, is the need to determine the importance of the stimuli in order to decide who should be the *Focus of Attention* (FOA). This importance has been modeled using a set of gain weights that are tuned by optimization from a set of experimental data.

These gain weights represent how each stimulus contributes to a particular output of the neural network. In our experimentation, training is performed by three people interacting with each other and with the robot. In this step, during an initial training, an external user (teacher) analyzes the sequence and establishes to whom the robot should pay attention. This information is manually labeled together with the recorded stimuli. After the initial training, the gain weights are computed and the robot must pay attention to people as learned in the experiments. To validate the proposed methodology, a robotic head with a projected face that simulates eye movement has been developed. The result of the competition causes the robot to fix attention on the person, using the combined movement of the neck and eyes.

A new robot gaze control architecture is also presented. This new architecture uses a *Recurrent Neural Network* (RNN), a type of neural network that integrates feedback loops, allowing information to persist for a few steps through connections from the outputs of the layers, which embed their results in the input data. When the input sequence is long, a problem known as gradient vanishing may appear. This is solved by incorporating *Long Short-Term Memory* (LSTM) layers that allow back-propagation through time by connecting events that appear far apart in the input data, without their weight being diluted between layers. Our new architecture makes use of LSTM layers, analyzing the sequence of 30 frames composed of stimuli of people who interact with the robot. The network receives the same set of stimuli used in the competitive network but does not require its weighting. This new architecture can be trained in a simple way and offers promising results.

The present paper is structured as follows: Section 2 explores the state-of-the-art of the technologies considered in this paper. Section 3 briefly explains how the gaze control works with the competitive network and presents the approach for computing the weights of the stimuli and the new architecture developed with LSTM layers. In Section 4, the different experiments and results are reported. Finally, Section 5 notes the conclusions of the presented work and suggests future developments.

2 Overview of the related work

Human–Robot Interaction is a discipline that allows robots that can communicate and respond to ongoing human communications and behavior to be improved [21]. Gaze control is an important topic in HRI. A recent survey of the state of the art in social gaze for HRI was presented by [2], distinguishing three different approaches to the problem: human-focused, centered on understanding the characteristics of human behavior during interactions with robots; design-focused, which studies how the design of a robot impacts

interactions with humans; and technology-focused, centered on researching how to build tools to guide robot's gaze in human interaction.

The main challenges in a conversation are the management of attention and turn-taking between partners, controlling the gaze and adopting the right conversational roles [2]. According to [24], when a robot is a listener in a conversation between multiple interlocutors and follows the conversation with its gaze, it promotes higher evaluations of its comprehension and naturalness than a robot performing random gazing between speakers. Furthermore, in [4], the authors showed how gaze control more effectively motivates users to repeatedly engage in therapeutic tasks. Moreover, as seen in [15], virtual agents who use turn-based gaze during conversations are evaluated as more natural and pleasant than others that use none, or a random gaze control in their communication. At the same time, robots with humanoid features positively influence people's behavior towards the machine and their expectations about its capabilities [37].

During the last few years, different techniques have been used for gaze control, such as that proposed in [3], where the authors used a circular array of microphones on a social robot, named Maggie, to determine in which direction the robot should look. An infrared laser was used to obtain the distance with respect to the person so that the robot could move forward/backward. In [33], a robot-assisted therapy method was presented, in which a robot perceived different stimuli (visual, auditory, and tactile) to follow a certain colored object, a face, or a directional voice. These two previous works did not address the problem of who to pay attention to in a natural way.

Different authors have shown the benefits of fusing sensory information, such as [33, 40], or [42]. In [40], a person with a robotic head is localized by simultaneous processing of visual and audio data. However, these works do not focus on a conversation between multiple participants. In [42], the authors created a system to guide a robot's gaze to multiple interacting humans by adding different stimuli: social features, proxemic values, orientation, and a memory. This approach considered a limited number of stimuli and the maximum value could change abruptly and cause erratic changes in the focus of attention. Our proposed method based on a competitive neural network solves this problem by creating smooth transitions between participants.

The stimuli used by a robot can be diverse, but those based on computer vision, audio and memory represent some of the most commonly used. Visual information represents an important aspect of HRI [16]. When a robot is interacting with people, the detection of people is required. Some methods are able to detect human bodies, such as *Haar filters* [41], HoG [9], or *Deep Convolution Neural Networks* (DCNN) [31]. However, in our approach, a face recognition algorithm has been preferred, since the participants are assumed to look directly at the robot or have a slightly turned position to look at other people interacting with it. As indicated for human body detections, Haar classifiers, HoG detectors or Deep Learning based solutions are widely used. Haar classifiers detect faces at different scales, but do not deal with non-frontal faces or occlusions. It also returns a large number of false predictions. The HoG feature descriptor is fast, but does not work with small faces. The DLIB library [22] implements a CNN face detector using a Maximum-Margin Object Detector [12], which works for different face orientations and occlusions. Recognition can be implemented with Deep Residual Learning algorithms, which are very accurate. DLIB implements a ResNet network with 29 convolution layers and uses a pre-trained model which takes the 68 face landmarks obtained from an image [20]. In addition to face detection, lip activity is important to detect whether a person is speaking or not. When several people are situated in front of the robot and some audio is detected, it is likely that the

person who is moving the lips is currently speaking. Some works have explored different techniques to detect lip activity, such as [5], where the degree of disorder of pixel directions around the lips using the optical flow technique is measured; or [35], where a statistical algorithm using two detectors based on noise to characterize visual speech and silence in video sequences is created.

The *Robot's Field of View* (FOV) has to be considered to detect visually people situated in front of the robot's camera. When people are not situated in front of the robot, some information has to be kept in memory. A hypotheses generation was proposed by [34], inferring the people's position by using peripheral vision. Even though a person was not present in the foveal vision, the robot kept plausible hypotheses about the location. In [39], the authors proposed a dynamic visual memory to store information about objects from a moving camera on board a robot and created an attention system based on where to look to re-observe objects in memory and the need to explore new areas. The VFOA represents who or what people are looking at. As presented in [28], there is a relation between head poses and object locations. Audio represents an important stimulus, since microphone arrays can indicate the direction of the incoming sound. As explained previously, some works using microphone arrays, such as [3, 40], have been used with social robots.

LSTM networks have been widely used in recent years. Due to the ability of LSTM networks to remember in large data streams, they have been used for different problems, such as video sequence monitoring [27], *Human Action Recognition* (HAR) [7, 29, 38], speaker recognition [1] or even to extract business intelligence from sentiment analysis [27]. Regarding the new LSTM-based architecture presented, several related gaze control papers have been published. Recently, in [14], the authors have shown the effects of understanding human gaze communication by spatio-temporal graph reasoning, which allows us to see the real relationship between the behavior of people and the stimuli, showing that their interpretation and humans have the unique ability to infer others' intentions from eye gazes [13]. This approach with LSTM models is also followed by other authors [6, 8, 23].

Among the methods most similar to ours, it is worth noting the recently published by [26], where the authors present a system that uses *Reinforcement Learning* (RL) to focus its attention onto groups of people from its own audio-visual experiences. It also uses a pre-trained LSTM with simulated values. However, they start from raw information (audio/sound raw bitmaps) and perform gaze control in a generic way. The authors do not provide a direct result on behavior and only report the rewards obtained for face detection and speech. In our approach, all person's stimuli are individually obtained and they are treated in a single personal LSTM, rather than a general one. In addition, among other stimuli, the focus of the participants' gaze is also integrated in our model. Finally, in [30], the authors indicate the importance of a robot's head movement in nonverbal communication. In their case, they propose a cascaded LSTM model that first estimates the gaze from speech content and hand gestures performed by the partner. Their approach focuses on a very specific interaction with the hands, but highlights the importance of these networks in HRI.

3 Analysis of the system

Gaze control is usually implemented considering different stimuli [33, 40, 42]. In our approach for the stimuli integration, two types of neural works have been studied. Our previous approach [11], based on an on-center off-surround competitive model, needed the

adjustment of the weights of the stimuli to be done heuristically [10]. Competitive neural networks [17] can process different inputs and decide the winner in a dynamic and natural way. This kind of network can also have habituation capabilities. However, the problem is how to determine the importance of certain stimuli over others. This can be done by a set of weights. These weights have to be configured or learned to replicate human behavior when the same stimuli are produced. In this work, the calculation of the weights is explained by means of an optimization process. Alternatively, a new model based on *Recurrent Neural Networks* is proposed. This new architecture makes use of LSTM layers and does not directly require weighting of the weights and the manual adjustment of the parameters of the competitive network.

For both approaches, different stimuli are considered. These stimuli are the inputs of the competition at an instant of time t, I_{tkx}, where k is the number of persons who interact and x the number of stimuli. They have a binary value, indicating whether they are present or not, and are balanced by a weight, w_x. In some stimuli, such as speech detection, the value is activated if the different indicators exceed a threshold.

- I_{tk1} shows if a person k is situated in the robot's FOV at the instant of time t. People situated in front of the robot's FOV are usually candidates to interact with the robot. w_1 is the corresponding weight associated to that stimulus.
- I_{tk2} shows that a person k is considered to be speaking. This stimulus is obtained by performing lip movement detection, based on mouth landmarks. In addition, incoming audio has to be detected in the direction of the person k.
- I_{tk3} shows that a person k is gazing directly at the robot. The pose of a person is used to verify if that person is visually interacting with the robot. Engagement in an interaction is increased by the *mutual gaze*, a kind of *shared looking* [36].
- I_{tk4} shows that a person k is continuously moving. In a conversation with several people, an individual tends to look at another restless person. This stimulus requires the individual to be situated in the FOV of the robot. If the sum of the differences in a person's position between several frames is over a given threshold, the person is assumed to be restless.
- I_{tk5} shows that a person k is not situated in the robot's FOV, but for whom incoming audio could have been detected. When the robot does not see a person who has previously interacted, it keeps the previous position of the person in its memory and, if incoming audio is detected in his/her direction, the stimulus for that person k is activated. In a conversation between humans, when someone is speaking at their left/right side, a person tends to turn their head in the direction of the person who is speaking.
- I_{tk6} shows that a person k is in the VFOA of other people, but is not situated in the robot's FOV (see Fig. 1). When two or more people are looking at another person in a conversation, a stimulus is given to people in the direction of the gaze. Since the focus of attention is given to a concrete person, the corresponding stimulus is increased.
- I_{tk7} indicates the proxemics of a person. People situated at a certain distance are likely to be interacting with the robot [3, 42]. In addition to the weight w_7, this stimulus is balanced by a tuning factor, depending on the distance between the robot and the person.

Next, the optimization of the control system of a robotic head, previously developed by the authors [11], and a new model of the neural network are presented. This new model replaces both the previous competitive network and the weight optimization process itself.

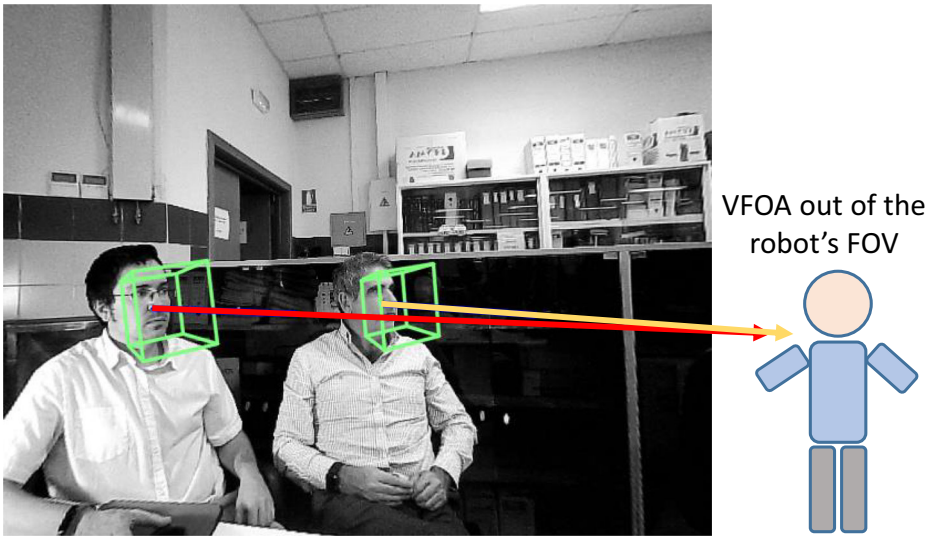

Fig. 1 Two people with their VFOA in another person

It allows easy training in an automatic way and avoids both the manual adjustment of the model parameters and the computation of the weights. Although the preliminary results are slightly lower than those of previous model, the ease of training is important to us. The new model is an end-to-end network that is trained with the input stimuli along a sequence and as output returns the winner during that sequence. The optimization of the weights of the previous model is first explained (Section 3.1), followed by the presentation of the new model (Section 3.2). Both methods use the same group of input stimuli.

3.1 Weight stimuli optimization

The different stimuli that influence a gaze control system must be weighted in such a way that it allows the system to behave similarly to humans. The gaze control system is implemented through a competitive neural network, where each stimulus competes with the others to provide a smooth dynamic result. However, the input of the competitive network, composed by the relation of stimuli and their weights, must be correctly assigned for the robot to have a realistic behavior. Figure 2 shows the system scheme, in which an optimization problem allows the weights that will be used in the gaze control system to be obtained. This optimization requires the data to be split into winners and losers.

The competitive network receives of a set of inputs, where each input I_{tk} is obtained by adding the stimuli of the person k, I_{k1},..., I_{k7}, balanced by their respective weights, at time t. Thus, the input of the network produces an output O_{tk} that represents if a person k is the winner. The output with the highest value corresponds to the winner. The configuration of the weights, w_x, is not initially known, so a mechanism has been implemented to obtain it. The competitive network keeps information about preceding states and dynamically adapts the outputs depending on its configuration. At a given time t, there is an input vector of the network, I_t, composed by the value associated to each person (I_{t1}, ..., I_{tn}). I_{tk} is computed as shown in (1).

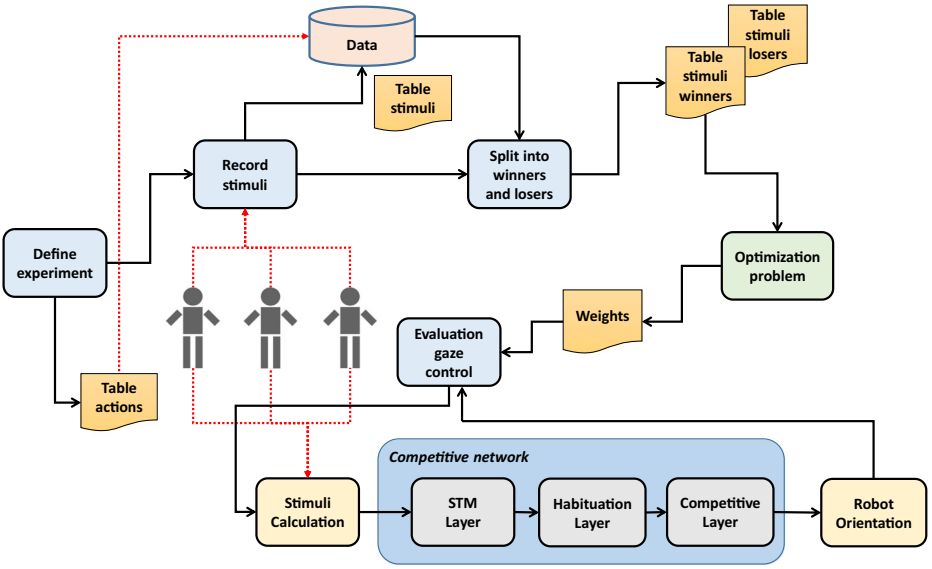

Fig. 2 Scheme of the training system

$$I_{tk} = \sum_{x=1}^{7} w_x \cdot I_{tkx} \tag{1}$$

The output of the network depends on the input vectors obtained previously, as seen in (2), where C, H and S are, respectively, the layers associated to the competitive, habituation and short time memory (STM) operations.

$$O_t = C(H(S(I_1, ..., I_t))) \tag{2}$$

The STM, habituation and competitive layers are based on the model of Grossberg [17] and use the differential (3), (4) and (5), respectively.

$$\frac{dx_i}{dt} = -A_1 x_i + C_1(B_1 - x_i)[I_i w_i] \tag{3}$$

$$\frac{dg_i}{dt} = E(1 - g_i) - F I_i' g_i \tag{4}$$

$$\frac{dy_i}{dt} = -A_2 y_i + C_2(B_2 - y_i)\left[I_i'' + D{y_i}^2\right] - y_i \sum_{i \neq j} D{y_i}^2 \tag{5}$$

The STM layer, where A_1 corresponds to the decay rate, B_1 to the saturation and C_1 to the growth rate, receives the input stimuli, I_i, and produces an output where the duration of the stimuli is increased. The STM output is the input of the habituation layer, I_i', where the permanent stimuli lose interest through time. In the habituation layer, g_i is the gain for the stimuli. When a stimulus is active, the habituation gain decreases from a maximum value, 1, to the value given by $E/(E + F I_i')$, where E and F correspond to the charge and discharge rates. The output of the habituation layer is the input of the competitive layer,

I_i'', where A_2 is the decay rate, B_2 the saturation value and C_2 marks the growth rate. D balances the parabolic function ${y_i}^2$, reinforcing the winner against the rest, which represents a lateral inhibition (off-surround). The output of the competitive layer shows the winner of the competition, as explained previously.

The output of the network takes into account the dynamic nature of previous states by filtering spurious stimuli and producing a smooth change of the winning person focus of attention. However, the input of the network, I_i, corresponding to the stimuli, has to be optimized to obtain a group of weights which are optimal in the gaze control process. To this optimization, the training has to be carried out in a sequential way, following a list of steps previously recorded. The training is carried out by three people. At the same time, an external user (teacher) observes the interaction and annotates the time instants when a person should be the focus of attention.

When all data have been obtained, stimuli from expected winners and losers are separated based on the said manual annotation. At an instant t, there is an input value for the winner, $I_{t,winner}$, and an input value for every k loser, $I_{t,k}$. The process is modeled as an optimization problem, where the aim is to obtain the optimal weights that maximize the sum of the distances between the winners and the losers at each time instant t, as shown in (6). This procedure ensures that the weights are optimal to make the selected persons winners and separate them from the losers.

$$\begin{aligned} &\max \sum_{t=1}^{m} \left(\sum_{k \in losers} I_{t,winner} - I_{tk} \right) \\ &= \max \sum_{t=1}^{m} \left[\sum_{k \in losers} \left(\sum_{x=1}^{7} w_x \cdot I_{t,winner,x} - \sum_{x=1}^{7} w_x \cdot I_{t,k,x} \right) \right] \end{aligned} \tag{6}$$

Some constraints have to be considered. First of all, the sum of all weights has to be equal to 1, as shown in (7). In addition, the weights range between 0 and 1, being bigger than 0 (8).

$$\sum_{x=1}^{7} w_x = 1 \tag{7}$$

$$\forall x \in [1...7] : w_x \in [0, 1] \wedge w_x > 0 \tag{8}$$

Secondly, there is a constraint for each case evaluated. The input of the winner, $I_{t,winner}$, is bigger than or equal to the losers at an instant t, as shown in (9). With this constraint, the problem is forced to behave as annotated during the training phase.

$$\forall t \in [1..m] \wedge k \in losers \; at \; t : I_{t,winner} \geq I_{t,k} \tag{9}$$

Another consideration is related to the proxemic stimulus. This is balanced by a factor depending on the distance between the person and the robot during the gaze control operation. During the training phase, a fixed value of 1 is assigned. The training is carried out by people who are situated at the *far phase* of the personal space (0.76m to 1.22m) or at the

close phase of the social space (1.22m to 2.10m) [18], a region where this distance factor is 1. Beyond 2.10 meters, this factor decreases and is not significant for the calculation of weights because, during our training, the participants have been situated within these distances. During normal operation time, an adjustment factor balances the weight of people situated beyond 2.10 meters.

3.2 Towards the use of deep neural networks

During our experiments, using habituation and competitive layers in our previous model [11], a different approach was started. This another method allows us to train the system in a simpler way and also responds to situations that may be out of the ordinary by not relying on such dynamic models as those of the competitive network.

The new model is based on the use of a *Deep Neural Network* (DNN) whose inputs represent the set of stimuli that have been detected for each person along a set of f frames. Therefore, there will be stimuli, e_{ijk}, where i represents the person $(1...p)$, j represents the frame $(1...f)$ and k represents the stimulus (1...7). The output of the net will activate the person on which the robot must put the focus of attention. All the inputs will be worth 1 or 0 depending on whether the stimulus has been detected or not, except in the case of the proxemic, where the stimulus will depend on the distance.

Figure 3 shows this new model, where the data coming from the robotic head, video frames together with audio direction, are processed using the position tracking of the people who have previously interacted with the robot in order to associate each of the seven computed stimuli to each of the persons. Then, the stimulus input corresponding to each person is diverted to a different LSTM. Among the methods based on neural networks, RNNs, and specifically the use of a LSTM networks [19], are capable of learning long-term order dependences in sequence prediction problems. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. Basically, the cell can process data sequentially and keep its hidden state through time.

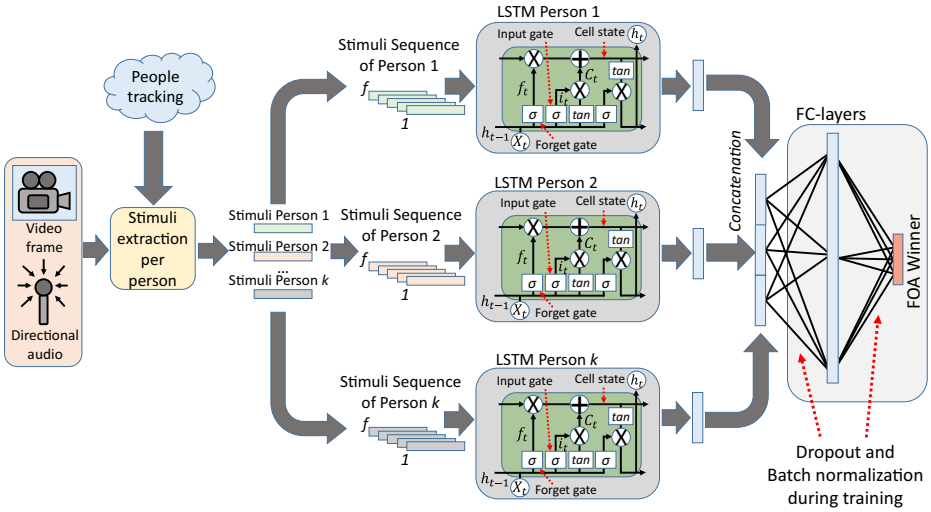

Fig. 3 Model based on DNN with LSTM layers

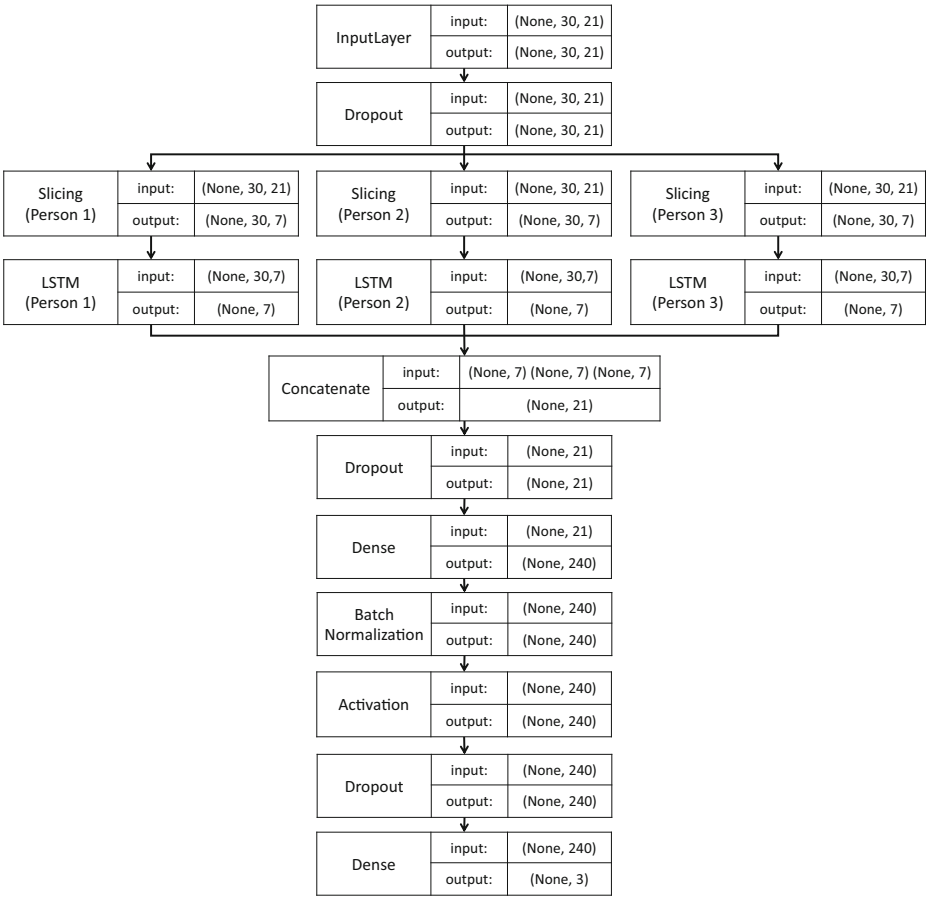

Fig. 4 Architecture of the model

For hence, in our new model, the LSTM seeks to analyze the behavior of each person over a period of time. The output of each LSTM reduces the data $(30 \cdot 7)$ to a set of 7 elements that allow to distinguish between different sequences. These 7 values obtained for a person are concatenated with those of the rest of people to be connected with a Fully-Connected (FC) layer. This layer is in turn connected to a classification layer (*softmax*) with one output for each person. This output shows who the winner is and should therefore be the new *Focus of Attention the Robot* (FOA).

An important aspect in this model is that each of the LSTM layers, corresponding to each person, should behave in an analogous way. To this end, the inputs are permuted during the training so that the system can recognize the same stimuli for each LSTM layer. Our project has followed a cyclical methodology, gradually improving the results based on small adjustments to the model: modifications of the input data configuration as well as modifications in number of hidden layers, neurons per layer, dropout percentage, batch normalizations, activation functions, learning rate/decay/momentum parameters, optimizers, etc. Figure 4 shows the implementation of the new architecture, including the hidden neurons. After a battery of training sessions, a single layer composed of 240 hidden neurons obtained the

best results. In addition, two types of regularization are carried out. There is a batch normalization of the FC layer to avoid values to be out-of-range. Moreover, dropout is used to avoid overfitting. The dropout is also applied to the input vector of the neural network.

Dropout was used in different points of the model: after the input layers, a dropout of 5%; after LSTM layers, a dropout of 20%; and after the FC layer, a dropout of 20% was included. The models were trained using an Adam optimizer with a learning rate of 10^{-3}. A metric *accuracy* was added during the training.

An important aspect of this new approach is how it works in real time. Although we could propose a sliding window model where the robot would process the current frame together with the 29 preceding ones, it is not operational according to our experiments. The 30 processed frames approximately represent one second of execution time. More than one head movement per second is excessive and gives an unnatural feeling. In addition, it causes mechanical problems and blurring images received by the camera. Therefore, each time the robot's head is moved using this new model, the stimulus processing and the set of frames to be processed starts again. The robot head processes the new stimuli and will produce the next possible movement in other 30 frames. This new technique also offers much easier training since the manually annotated training data is recorded with the head stationary.

4 Experiments and results discussion

Two different experiments have been carried out. On the one hand, the optimization algorithm has been used to obtain the stimulus weights at the input of the previous competitive network [11] (Section 4.1). On the other hand, the new model has been tested using an end-to-end network that does not require the manual adjustments of the previous model and allows a direct training of the received stimuli along a time sequence as well as the expected output (Section 4.2).

Table 1 Behavior sequences of three people

State / Stimuli	Person 1							Person 2							Person 3							Winner
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	
1	G						G	R		R				R								2
2	G		G				G	R		R				R	G						G	2
3	R		R	R			R	G		G				G	G		G				G	1
4	G		G				G	R	R	R	R			R	G		G				G	2
5	G		G				G	G		G				G	R			R			R	3
6	R		R	R			R	G		G				G	G						G	1
7	G						G	R	R	R				R	G		G				G	2
8	R		R				R	G						G	G	G					G	1
9	G						G	G		G				G	R						R	3
10	G		G				G	R		R				R								2
11	G						G	G		G				G					R	R		3
12	R	R	R				R	G						G						G		1

4.1 Weight stimuli optimization

The experiment consisted in obtaining the optimal weights during training with three people interacting with the robot. The weights were not initially known and the persons followed a list of actions previously established, as shown in Table 1. The complete list of actions had 726 states. The table shows the losers in green and the winners in red. Only 12 out of the 726 states are shown, but all of them were evaluated in the maximization problem.

The optimization problem was solved using the SLSQP algorithm [25], which obtained the results in 18 iterations and 0.23 seconds (in an intel i9-9900K, with 32Gb of RAM). The obtained results were $w_1 = 0.06$, $w_2 = 0.25$, $w_3 = 0.06$, $w_4 = 0.16$, $w_5 = 0.16$, $w_6 = 0.25$ and $w_7 = 0.06$.

These weights were evaluated in the previously self-developed robotic head (see Fig. 5) [11], where the competitive network had been deployed. The network created smooth transitions between the focus of attention, resulting in a natural behavior but, at the same time, considering properly which stimuli were more important according to the obtained weights.

Although the system has been trained with 3 people, the stimuli generated during the training are independent of the person. The methods of obtaining stimuli that have been used are generic and suitable for any person. The 726 cases used allow to refine the weights to a very wide range of situations. Training with more people may slightly vary these weights but not in a significant way. At normal operation time, the system works with 3 or more people, having even performed some test with more than 8 participants. Beyond the increase in computing time, the system was able to correctly gaze at the participants who were winning the competition. For performance reasons, our system has been limited to 10 participants.

When a participant disappears from the robot's FOV, the head is able to estimate the position using a Kalman filter [32]. If the robot interacts with 3 participants and one disappears, either because the person moves or because the head turns, when audio is detected in his direction or the other participants are looking at him (VFOA), stimuli will be provided

Fig. 5 Self-developed robot head

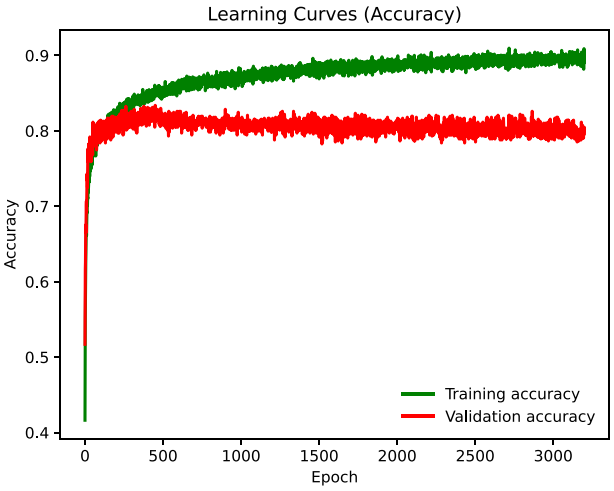

Fig. 6 Training curve

to that participant and the head will gaze at this person in case of being winner, using the estimated Kalman position. When no stimuli are received from a participant for more than 20 seconds, the participant is removed. When a new person arrives on the scene, regardless of whether the robot is able to detect him in the FOV or not, if audio is detected in his direction or if this person is in the VFOA of other participants, the new participant will enter the competition. If this new person becomes a winner, the head will turn to look for him. If no one is detected, it will go back to the old winner.

Although the main aim of this work has been to obtain the weights of our neural network, the head has quantitatively behaved as we expected in 85.0% of cases, higher compared to other works, such as [42], where the authors obtained results between 75.2% and 89.4%,

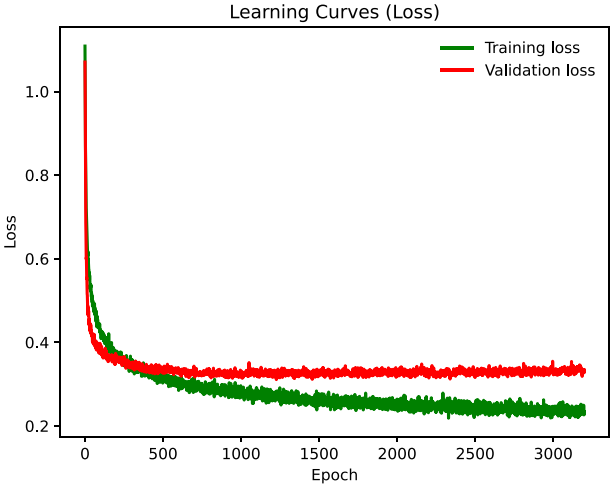

Fig. 7 Loss curve

Table 2 Confusion Matrix of the new model

Expected \ Estimated	Person 1	Person 2	Person 3
Person 1	0.92	0.07	0.01
Person 2	0.03	0.90	0.07
Person 3	0.14	0.02	0.84

depending on the saccadic and non-saccadic movements. However, their experimental conditions were very different. In their work, the interaction was performed between two people at some distance prioritizing gestural and postural acts. They used an RGB-D Kinect and a DIK-ABLIS eye-tracking system.

4.2 Experiments with the model using LSTM

The training of our model based on LSTM has been carried out from 726 situations previously recorded on 3 people. These situations have been transformed into 696 different windows given that a sliding window sized 30 frames has been chosen for each sequence ($696 = 726 - 30$). These situations have been permuted as explained previously to obtain a dataset of 4,176 sequences. For each situation, the obtained stimuli for each person along 30 frames and the winner have been labeled. This dataset has been split into training (72% $\rightarrow$ 3,007 sequences), validation (18% $\rightarrow$ 752 sequences) and test (10% $\rightarrow$ 417 sequences). Figures 6 and 7 shows the training and loss curve of the model, including the validation accuracy. The model has been trained in 60 minutes using a server i9-10900K with 128Gb RAM and 2 GPU RTX-3090 with 24GB GDDR6X. The accuracy of the model has been 89.91%, validation accuracy 83.38%, and test accuracy 82.80%. The loss in training accuracy has been 0.11 while the loss of validation accuracy has been 0.20. Table 2 shows the Confusion Matrix of the model for the 3 people who have participated in the dataset preparation.

Our model has been evaluated using different techniques, as shown in Table 3. A *Multi-Class Precision-Recall* curve, with all classes, is displayed in Fig. 8. In addition, a *Precision-Recall* curve with the average precision score, micro-averaged over all classes, is displayed in Fig. 9. The evaluation of the model shows a good performance leading the *Area Under the Curve* (AUC) close to 1 in all cases.

Although the obtained results are slightly lower than the competitive method, this end-to-end model is much easier to be trained and to be integrated into the robotic head. It does

Table 3 Evaluation of the model with different metrics (%)

Metric	Value
Accuracy	0.8878
Precision	0.8878
Recall	0.8878
F1 score	0.8878
Cohens kappa	0.8309
ROC AUC	0.9874

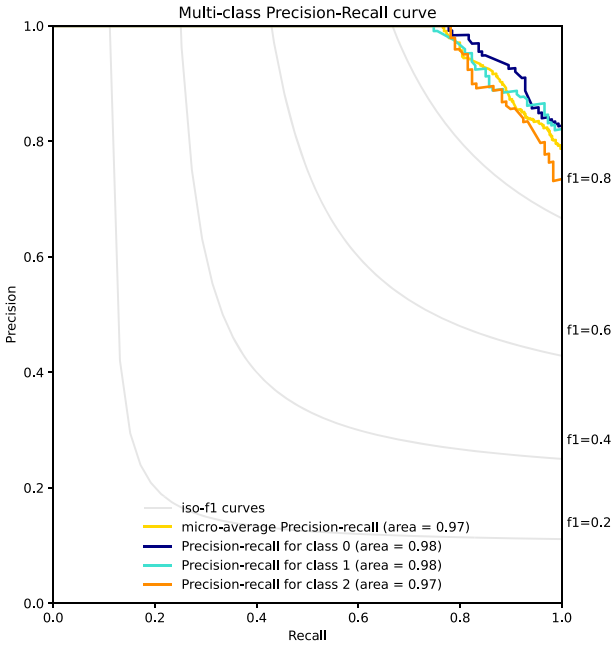

Fig. 8 Multi-Class Precision-Recall curve

not require manual adjustments to the competitive network equations and the optimization problem is solved by the training. Table 4 shows a comparison with other similar works.

The proposed system reflects a success rate of 82.80% in relation to the behavior a person would have in the same situation. The errors are mainly due to disturbances in the perception of the stimuli, such as external noises not related to the interaction, and blurred images due to the effect of the movement of the people and the robot itself. It is difficult to make a comparison of the results with other authors, as there are no common datasets to compare with.

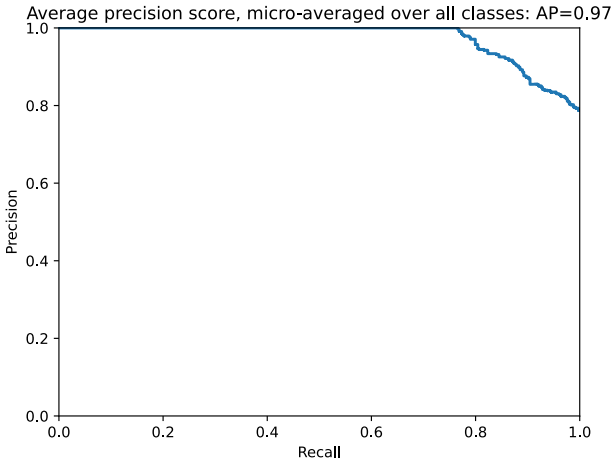

Fig. 9 Precision-Recall curve with the average precision score, micro-averaged over all classes

Table 4 Comparison with other similar works

Paper	Success rate	Technique
[11]	85.0%	Competitive neural network [17] that requires manual adjustments and problem optimization.
[26]	Not provided	They provide face/speak rewards using *Reinforcement Learning* (RL) to focus its attention onto groups of people from its own audio-visual experiences.
[42]	75.2% to 89.4%	Attention mechanism maximizing the sum of different elements: social features, proxemics values, orientation, and a memory component. Interaction between two people
Proposed	82.80%	Multiple LSTM network to train gaze control based on previous experiences. Much easier to be trained and incorporated into the robotic head.

The closest research to the proposal corresponds to [42], which obtains results with accuracy between 75.2% and 89.4% depending on the saccadic and non-saccadic movements. However, their experimental conditions were different as has been previously discussed. In [26], the authors do not provide a direct result on behavior and only report the rewards obtained for face detection and speech. Although with a different technique, the present work is based on a principle similar to our previous paper [11], where stimuli gradually increased the output value of a competitive network until the robot changed its focus of attention. In that work we obtained 85.0%, although it required manual choice of certain parameters and a much more complex parameter optimization problem than the training presented in this paper.

5 Conclusions

This work presents an optimization process for a robotic gaze control system and a new architecture of the whole system. The gaze control system uses a competitive network which receives a large number of visual, auditory or presence stimuli. It allows a smooth transition, changing the focus of attention between participants, avoiding erratic movements. In addition, it has habituation capabilities to avoid someone from hoarding the conversation. The weights of each stimulus are not known a priori and a strategy based on an optimization problem has been developed to obtain them through experiential learning. The computed weights were integrated into the gaze control system of a self-developed robotic head with combined body and eye movement, and the robot showed a natural interaction behavior similar to those annotated during the learning phase. The proposed method significantly improved the sensation of naturalness and realism of the robotic head. The movement of the robot joints and the expressions of the virtual agent projected on its 3D facial display were controlled by the system integrated in a ROS-based architecture. The design of the robot and the gaze control system creates a more realistic HRI system, which is more acceptable to interlocutors than other not-so-human robots. In addition, a new architecture that makes use of LSTM layers has been described. This new architecture offers promising results, compared to other models. It uses the same set of stimuli of the previous model but does not require its weighting. Also, it is much easier to be trained and incorporated into the robotic

head. This new method replaces both the competitive network itself and the weight optimization method, and represents an improvement in terms of being able to operate the head in a simpler way.

The future objectives of the project will consist in integrating speech capabilities into the ROS architecture, offering a low-cost, intelligent robot with human-like behavior.

Acknowledgements The present research has been partially financed by "Programa Retos Investigación del Ministerio de Ciencia, Innovación y Universidades (Ref. RTI2018-096652-B-I00)" and by "Programa de Apoyo a Proyectos de Investigación de la Junta de Castilla y León (Ref. VA233P18)", cofinancied with FEDER funds.

References

1. Abd El-Moneim S, Nassar M, Dessouky MI, Ismail NA, El-Fishawy AS, Abd El-Samie FE (2020) Text-independent speaker recognition using lstm-rnn and speech enhancement. Mult Tools Appl 79(33):24,013–24,028
2. Admoni H, Scassellati B (2017) Social eye gaze in human-robot interaction: a review. J Human Robot Interact 6(1):25–63
3. Alonso-Martín F, Gorostiza JF, Malfaz M, Salichs MA (2012) User localization during human-robot interaction. Sensors 12(7):9913–9935
4. Andrist S, Mutlu B, Tapus A (2015) Look like me: matching robot personality via gaze to increase motivation. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems, pp 3603–3612. ACM
5. Bendris M, Charlet D, Chollet G (2010) Lip activity detection for talking faces classification in tv-content. In: International conference on machine vision, pp 187–190
6. Benrachou DE, dos Santos FN, Boulebtateche B, Bensaoula S (2015) Online vision-based eye detection: Lbp/svm vs lbp/lstm-rnn. In: CONTROLO'2014–proceedings of the 11th Portuguese conference on automatic control, pp 659–668. Springer
7. Carrara F, Elias P, Sedmidubsky J, Zezula P (2019) Lstm-based real-time action detection and prediction in human motion streams. Multimed Tools Appl 78(19):27,309–27,331
8. Chen Y, Liu C, Shi BE, Liu M (2020) Robot navigation in crowds by graph convolutional networks with attention learned from human gaze. IEEE Robot Auto Lett 5(2):2754–2761
9. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection
10. Domingo JD, Gómez-García-Bermejo J, Zalama E (2020) Optimization of a robotics gaze control system. In: Workshop of physical agents, pp 213–226. Springer
11. Duque-Domingo J, Gómez-García-Bermejo J, Zalama E (2020) Gaze control of a robotic head for realistic interaction with humans. Front Neurorobot 14:34
12. King E (2015) D.: Max-margin object detection. arXiv:1502.00046
13. Emery NJ (2000) The eyes have it: the neuroethology, function and evolution of social gaze. Neurosci Biobehav Rev 24(6):581–604
14. Fan L, Wang W, Huang S, Tang X, Zhu SC (2019) Understanding human gaze communication by spatio-temporal graph reasoning. In: Proceedings of the IEEE international conference on computer vision, pp 5724–5733
15. Garau M, Slater M, Bee S, Sasse MA (2001) The impact of eye gaze on communication using humanoid avatars. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp 309–316. ACM

16. Gergle D, Kraut RE, Fussell SR (2013) Using visual information for grounding and awareness in collaborative tasks. Human Comput Interact 28(1):1–39
17. Grossberg S (1982) Contour enhancement, short term memory, and constancies in reverberating neural networks. In: Studies of mind and brain, pp 332–378. Springer
18. Hall ET, Birdwhistell RL, Bock B, Bohannan P, Diebold JrAR, Durbin M, Edmonson MS, Fischer J, Hymes D, Kimball ST et al (1968) Proxemics [and comments and replies]. Curr Anthropol 9(2/3):83–108
19. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
20. Kazemi V, Sullivan J (2014) One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1867–1874
21. Kiesler S, Hinds P (2004) Introduction to this special issue on human-robot interaction. Human Comput Interact 19(1-2):1–8
22. King DE (2009) Dlib-ml: A machine learning toolkit. J Mach Learn Res 10(Jul):1755–1758
23. Koochaki F, Najafizadeh L (2019) Eye gaze-based early intent prediction utilizing cnn-lstm. In: 2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC), pp 1310–1313. IEEE
24. Kousidis S, Schlangen D (2015) The power of a glance: Evaluating embodiment and turn-tracking strategies of an active robotic overhearer. In: 2015 AAAI Spring symposium series
25. Kraft D, Schnepper K (1989) Slsqp—a nonlinear programming method with quadratic programming subproblems. DLR Oberpfaffenhofen
26. Lathuilière S, Massé B, Mesejo P, Horaud R (2019) Neural network based reinforcement learning for audio–visual gaze control in human–robot interaction. Pattern Recogn. Lett. 118:61–71
27. Liu F, Chen Z, Wang J (2019) Video image target monitoring based on rnn-lstm. Multimed Tools Appl 78(4):4527–4544
28. Massé B (2018) Gaze direction in the context of social human-robot interaction. Ph.D thesis
29. Meng B, Liu X, Wang X (2018) Human action recognition based on quaternion spatial-temporal convolutional neural network and lstm in rgb videos. Multimed Tools Appl 77(20):26,901–26,918
30. Nguyen DC, Bailly G, Elisei F (2018) Comparing cascaded lstm architectures for generating head motion from speech in task-oriented dialogs. In: International conference on human-computer interaction, pp 164–175. Springer
31. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99
32. Rosales R, Sclaroff S (1998) Improved tracking of multiple humans with trajectory prediction and occlusion modeling. Tech. rep. Boston University Computer Science Department
33. Saldien J, Vanderborght B, Goris K, Van Damme M, Lefeber D (2014) A motion system for social and animated robots. Int J Adv Robot Syst 11(5):72
34. Shiomi M, Kanda T, Miralles N, Miyashita T, Fasel I, Movellan J, Ishiguro H (2004) Face-to-face interactive humanoid robot. In: 2004 IEEE/RSJ International conference on intelligent robots and systems (IROS)(IEEE Cat. No. 04CH37566), vol 2. IEEE, pp 1340–1346
35. Siatras S, Nikolaidis N, Krinidis M, Pitas I (2008) Visual lip activity detection and speaker detection using mouth region intensities. IEEE Trans Circ Syst Video Technol 19(1):133–137
36. Sidner CL, Kidd CD, Lee C, Lesh N (2004) Where to look: a study of human-robot engagement. In: Proceedings of the 9th international conference on Intelligent user interfaces, pp 78–84. ACM
37. Thrun S (2004) Toward a framework for human-robot interaction. Human Comput Int 19(1):9–24
38. Ullah A, Ahmad J, Muhammad K, Sajjad M, Baik SW (2017) Action recognition in video sequences using deep bi-directional lstm with cnn features. IEEE Access 6:1155–1166
39. Vega J, Perdices E, Cañas J (2013) Robot evolutionary localization based on attentive visual short-term memory. Sensors 13(1):1268–1299
40. Viciana-Abad R, Marfil R, Perez-Lorenzo J, Bandera J, Romero-Garces A, Reche-Lopez P (2014) Audio-visual perception system for a humanoid robotic head. Sensors 14(6):9522–9545
41. Viola P, Jones M et al (2001) Rapid object detection using a boosted cascade of simple features. CVPR (1) 1:511–518
42. Zaraki A, Mazzei D, Giuliani M, De Rossi D (2014) Designing and evaluating a social gaze-control system for a humanoid robot. IEEE Trans Human Mach Syst 44(2):157–168