2020 17th International Conference on Ubiquitous Robots (UR)
June 22-26, 2020. Kyoto, Japan

Learning from Humans to Generate Communicative Gestures for Social Robots

Nguyen Tan Viet Tuyen, Armagan Elibol, and Nak Young Chong

Abstract— **Non-verbal behaviors play an essential role in human-human interaction, allowing people to convey their intention and attitudes, and affecting social outcomes. Of particular importance in the context of human-robot interaction is that the communicative gestures are expected to endow social robots with the capability of emphasizing its speech, describing something, or showing its intention. In this paper, we propose an approach to learn the relation between human behaviors and natural language based on a Conditional Generative Adversarial Network (CGAN). We demonstrated the validity of our model through a public dataset. The experimental results indicated that the generated human-like gestures correctly convey the meaning of input sentences. The generated gestures were transformed into the target robot's motion, being the robot's personalized communicative gestures, which showed significant improvements over the baselines and could be widely accepted and understood by the general public.**

I. INTRODUCTION

Non-verbal behaviors play an essential role in human-human interaction. Psychological studies have shown that people tend to use facial and bodily expressions during the conversation to signal their intention and attitudes [1], which influence social outcomes. It is convinced that social human-robot interaction should be treated in the same way the interaction occurs between people [2]. By adding the robot's social cues to its interaction behavior, the robot could help improve the interacting partner's perception and makes the social interaction outcomes enhanced [3], [4].

Toward understanding the effect of social cues, generating communicative gestures has been received increasing attention in the social robotics domain. In [5], the authors proposed Behavior Expression Animation Toolkit (BEAT), which receives the input text to be spoken and releases the non-verbal behaviors. In the BEAT toolkit, the mapping from text to gesture is based on a set of rules derived from state of the art in the non-verbal conversational behavior research. Although this approach can produce various gestures, the fundamental motions must be designed manually. The model proposed in [6] accepts both the lexical content of utterances and audio signals as the inputs to generate the non-verbal behaviors for virtual agents. Similar to the BEAT toolkit, the basic behaviors must be designed in advance. In contrast with the rule-based approach [5], [6] in which the robots' gestures are limited to a set of rules, the data-driven approach [7], [8] endows robots with the capability of learning social gestures through human data. In [7], the authors proposed the 3D pose generation model receiving speech signal and/or text input to generate the gestures corresponding to certain specific words. Afterward, it is converted to the target robot joint angles. Instead of generating robots' gestures to convey the meaning of specific words, the bidirectional relation between the human body motion and natural language was investigated in [8]. The authors demonstrated the capability of their approach to generating text descriptions for a variety of human body motions. Conversely, given the text description input, the model produces the gestures displayed on the Master Motion Map (MMM) model. Since the generated actions are defined in joint space with respect to the MMM joint configuration, it is difficult to utilize this approach on the other robots whose kinematic structures are different from the MMM framework.

In this paper, we aim at generating communicative gestures for social robots taking into account the meaning of the whole sentences uttered by robots. Indeed, the generated gestures are defined in 3D Cartesian space, allowing them could be effectively implemented into a variety of social robot platforms. In order to attain this objective, the proposed approach is based on Conditional Generative Adversarial Network (CGAN) [9], an extension of Generative Adversarial Networks (GANs) [10] with an additional input condition, to generate communicative gestures when synthesizing the verbal content of speech. Recently, GANs have been successfully applied to a variety of domains, especially for image generation tasks [11]. GANs include the Generator and Discriminator networks which are simultaneously trained and updated. The Generator tries to create the samples imitating the training data distribution, while the Discriminator tries to distinguish between generated samples and real data of the training set. Although GANs have received considerable attention across different disciplines, generating robot motions with GANs is seldom explored [12] since this problem often involves high-dimensional data with complex dynamics. This paper extends the application of GANs for generating social robots' non-verbal actions when synthesizing their verbal content of speech.

The rest of the paper is organized as follows. In Section II, the proposed model of generating communicative robot gestures is described in detail. In Section III, our approach is validated on the publicly available dataset, which is further supported by experiments with a real robot. Finally, we draw some conclusions and describe the direction of our future work in Section IV.

The authors are with the School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa, Japan {ngtvtuyen, aelibol, nakyoung}@jaist.ac.jp

978-1-7281-5715-3/20/$31.00 ©2020 IEEE

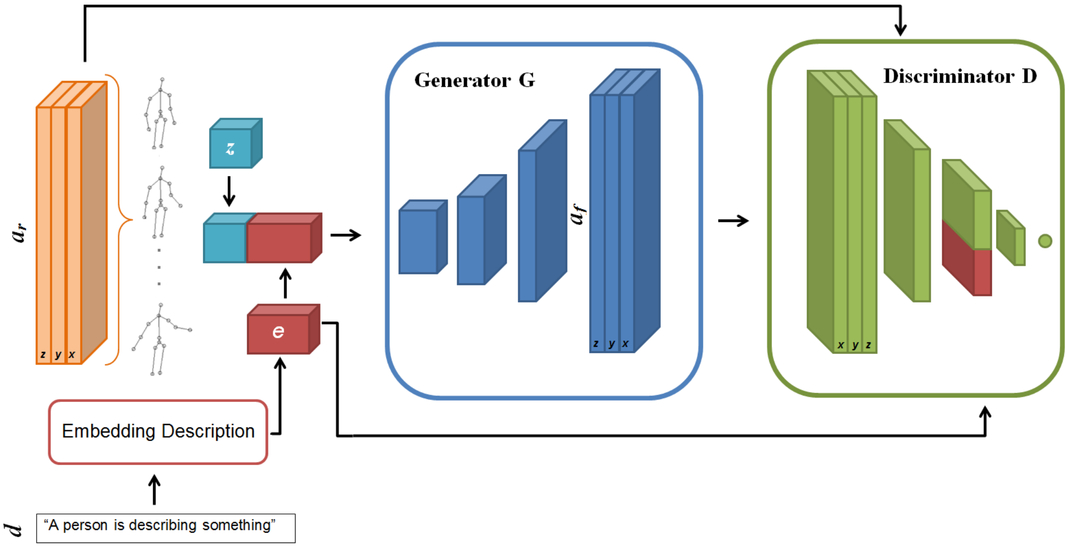


Fig. 1: The designed model based on CGAN for generating gestures a_f conditioned to the input descriptions d.

II. METHODOLOGY

Fig. 1 illustrates the proposed approach based on CGAN. The model consists of Generator G and Discriminator D. Firstly, $a_r = [S_1, S_2, S_3, ..., S_T]$ denotes a real action from the training data that contains a sequence of skeleton frames S over a period of time T. Here, the motion a_r contains 3 channels representing its joint positions using the Cartesian x, y, z coordinates in 3 dimensions. On each channel, the horizontal axis represents the time sequence of skeleton frames, while the vertical axis shows the spatial distribution of joints at a certain timestamp. This representation of action enables the convolutional neural network to capture the spatial and temporal information of motion at the same time [13]. On the other hand, $d = [w_1, w_2, w_3, ..., w_k]$ is a natural language sentence composed of k words describing the action a_r, which is fed into the Embedding Description network. The output vector $e \in \mathbb{R}^{n_e}$ from this model represents the meaning of the given text d. Then, the embedding vector e is concatenated with the noise vector $z \in \mathbb{R}^{n_z}$ before being fed into the G model. Finally, the fake action a_f is generated via $a_f \leftarrow G(z, e)$ having the same dimensions as the sample a_r, where $a_f = [S'_1, S'_2, S'_3, ..., S'_T]$ consists of T poses. The action a_f can be transformed into the target robot's motion, as shown in Fig. 2.

The embedding vector e is also fed into D. Here, with the same action description input, the Discriminator tries to differentiate between the real action a_r and the fake gesture a_f by considering e. The Generator G generates more realistic actions to beat the Discriminator. Specifically, D and G play the min-max game on the objective function given by Eq. 1.

$$\min_G \max_D V(D,G) = \mathbb{E}_{a_r, e \sim p_{data}(a_r, e)} [logD(a_r, e)] + \mathbb{E}_{e \sim p_{data}(e), z \sim p_z(z)} [log(1 - D(G(z,e),e))] \quad (1)$$

The remainder of this section will explain the proposed model in more detail.

A. Embedding Description

In order to encode the input description into the fixed-length embedding vector e, which efficiently captures the meaning of the whole sentence, $d = [w_1, w_2, w_3, ..., w_k]$ is fed into the Embedding Description. Here, we use the encoder phase of the skip-thoughts model [14]. The output vectors from this model effectively represent the semantics and syntax of the sentence to be encoded [14].

The hidden layer h_k represents the sequence of words $\{w_1, ..., w_k\}$. h_k is calculated by Eq. 2, where c_k is the word embedding of w_k, W, U are the weight matrices, $\odot$ denotes a component-wise product, z_k and r_k represent the update gate and reset gate of Gated Recurrent Unit [15], respectively. The hidden state h_k captures the meaning of the whole sentence d, this value is then compressed into a smaller dimensional vector e before being fed into the Generator and Discriminator model.

$$h_k = (1 - z_k) \odot h_{k-1} + z_k \odot tanh(Wc_k + U(r_k \odot h_{k-1})) \quad (2)$$

B. Generator Network

The proposed model is based on the transposed convolutional network which has been shown to be useful in many different research contexts such as image generation [16], [17], video generation [18], and audio generation [19]. Motivated by the above-mentioned applications, this paper investigates the convolution operation toward an autonomous generation of communicative robot actions. Firstly, the noise vector z is sampled from the normal distribution $N(0,1)$. The vector z is then concatenated with the embedding vector e from the previous step before being fed into the Generator model. The model G consists of a fully connected layer to reshape the input and four fractionally-strided convolutions to up-sample the data to the output target size. On the first three convolutional layers, batch normalization plays an important role in stabilizing the learning process. This

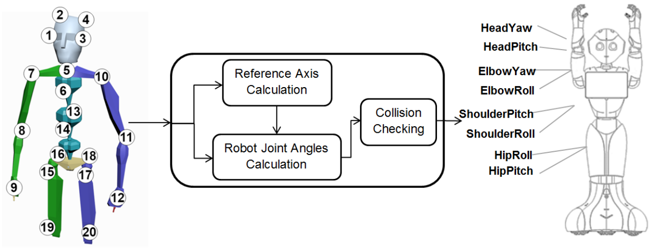


Fig. 2: The transformation model [21] converts generated actions in 3D Cartesian space to the robot joint motions.

operation normalizes the input to each unit to have zero mean and unit variance. The output values are then followed by the Rectified Linear Unit (ReLU) activation [20]. The last convolutional layer transposes the neuron units to the output target size. Here, $tanh$ activation function is applied before producing the action a_f.

C. Discriminator Network

The Discriminator D consists of five convolutional layers. The first one takes an action (either from the real training data a_r or action a_f from the Generator G) as the input. Similar to the architecture of G, batch normalization and ReLU activation functions are applied to all layers except the output. At the fourth layer, the embedding vector e is concatenated with the output of the convolutional layer. At the last layer, the results pass through a sigmoid function to produce the output value, which represents the probability that the input action comes from the training data.

The objective function in Eq. 1 is optimized using the gradient-based approach. It is solved in two steps; update the Discriminator and then followed by the Generator. Firstly, D is trained to maximize its ability to differentiate between the real sample a_r and the generated one a_f, referencing the input condition e. This is done by training the Discriminator to output 1 when the input is the real action a_r. Otherwise, D releases 0 if the model receives the action a_f as the input. Here, the binary cross-entropy is applied to compute the miss-classification error of the Discriminator:

$$L_D = log(y_r) + log(1 - y_f), \tag{3}$$

where $y_r \leftarrow D(a_r, e)$ is the output probability assigned by D for a pair of action a_r and embedding vector e of text description d. Similarly, $y_f \leftarrow D(a_f, e)$ is the output sigmoid from D for the a_f and e input. Thus, the Discriminator aims to maximize y_r while minimizing y_f. Specifically, the parameters of D are updated while keeping the parameters of G constant.

In the second step, the Generator is trained to maximize its ability to fool the Discriminator D with the loss function as shown in Eq. 4. Hence, the goal is to maximize the output probability y_f. The parameters of G are adjusted while keeping the parameters of D to remain unchanged.

$$L_G = log(y_f) \tag{4}$$

III. Experimental Results and Discussion

A. Dataset

To demonstrate the validity of the proposed model, we used the Karlsruhe Institute of Technology (KIT) whole-body motion dataset [22] and the corresponding natural language annotations [23]. The KIT motion dataset provides a rich corpus of human whole-body motion in a wide range of motion types. The selected data contains $2,127$ motions captured by 53 optical markers in 3D at the frequency of 100 Hz. Since this paper focuses on generating the motions for the humanoid robot Pepper, only 20 markers capturing the motion of human upper body and knees were selected out as the training data as shown in Fig. 2. The knees were included in computing the robot hip joint angles. Each selected action $a_r = [S_1, S_2, S_3, ..., S_T]$ consists of a sequence of skeleton frames over a period of time T. At the frame i $(i \leq T)$, $S_i = [x_1, x_2, .., x_{20}, y_1, y_2, .., y_{20}, z_1, z_2, .., z_{20}]$ $(S_i \in \mathbb{R}^{60})$ is the 60 dimensional vector that defines the positions of 20 joints in Cartesian space.

B. Preprocessing

Before feeding the dataset into the model, the preprocessing steps were conducted on the training data. The spelling errors in natural language annotations describing the demonstrative actions were corrected. With the $5,136$ usable annotation samples from the dataset (one action could be associated with more than one annotation), each description d was associated with the corresponding motion a_r. In order to encode the embedding vector representing the meaning of natural language description as described in II-A, the skip-thoughts model trained with the BookCorpus dataset [24] was utilized. As the BookCorpus dataset contains a collection of $11,038$ books in 16 different genres, it can be regarded that the training data does not suffer any bias towards any particular domain. Then, the encoder phase of the skip-thoughts model was used for generating the embedding description, as mentioned above. In terms of the demonstrative actions, we constructed joint values with respect to the top-chest coordinates. On the other hand, the size of the demonstrators is different from the training samples. Therefore, the actions were normalized to have the variance 1. Afterward, the motions were downrated to 10 Hz and padded to have an equal length of 240 frames. In total, $51,360$ pairs of motions and descriptions were obtained. We split it into 90% for training and 10% for testing.

C. Evaluation Metric

Let us consider that $a_r = [S_1, S_2, S_3, ..., S_T]$ is the real motion associated with the input description d. $a_f = [S'_1, S'_2, S'_3, ..., S'_T]$ is the generated action from the model G given d. In order to quantitatively validate a_f, we evaluate how close the motion sequence a_f is to the real action a_r over the whole time sequence T. Specifically, covariance with temporal hierarchical construction [25] was utilized to encode the action a_r and a_f into the corresponding feature vector C_r and C_f, respectively, using Eq. 5. Here, $\overline{S}$ is the

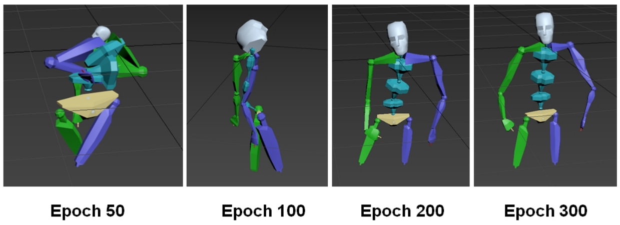


Fig. 3: Generated actions visualized using Autodesk 3DS Max: the Generator model imitated the human joint configuration to yield a natural human posture.

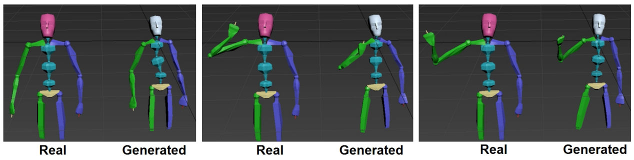


Fig. 4: Key poses of action for *"A person waves with its right hand"*. The left skeleton represents the real training data a_r, while the right shows the generated one a_f. The similarity between two gestures is 0.8841.

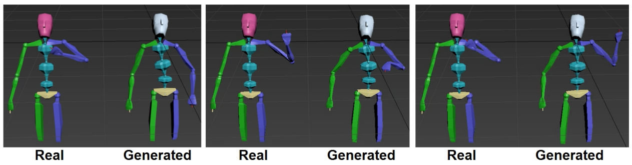


Fig. 5: Comparison between real data a_r and generated one a_f given the input *"A person waves with the left hand"*. The similarity between a_r and a_f is 0.7675.

sample mean of S_i computed over the time T and $^\intercal$ represents the transpose operator. Covariance $C \in \mathbb{R}^{n_c}$ efficiently represents the 3D joint movements over the time sequence by a fixed-length vector. This feature descriptor has been widely used for action recognition in both supervised [25] and unsupervised learning tasks [26]. With the calculated vectors $C_r \in \mathbb{R}^{n_c}$ and $C_f \in \mathbb{R}^{n_c}$, the similarity between a_r and a_f was measured by the cosine distance between them given in Eq. 6. The similarity score 1 means that they are exactly the same vectors.

$$C = \frac{1}{t-1}\sum_{i=1}^{T}(S_i - \overline{S})(S_i - \overline{S})^\intercal \tag{5}$$

$$Similarity(C_r, C_f) = \frac{C_r \cdot C_f}{||C_r||\,||C_f||} \tag{6}$$

D. Identification of Human Joint Spatial Configuration

The motions and the corresponding natural language annotations from the training set were fed into the designed model with the batch size 100. The Adam optimizer [27] was used at the learning rate 2×10^{-5} for both the Generator and Discriminator network. The model was trained until Epoch $1,200$. During the first 30 epochs, only the Discriminator was trained. After that, both D and G were sequentially trained. In order to monitor intermediate motions of a_f, the same description d and noise z were given to G during the training process. Fig. 3 shows the first frame S'_1 of each generated action.

At the beginning of the training phase, G could not capture the spatial configuration of the training samples. Because of that, the generated gestures at Epoch 50 are totally different from the shape of the human body. Starting from Epoch 100, G ameliorated the human joint configuration coordination problem and produced more natural human-like poses. At Epoch 300, the generated pose was well-proportioned, as seen in Fig. 3. Hence, throughout the training process, the Generator G was able to learn the coordination of human joint configurations. By the end of the training phase, G could generate the human body properly and symmetrically.

E. Generated Gesture Conditioned to Input Description

To demonstrate that the Generator model is able to produce the communicative gestures given the verbal content of speech, different text descriptions were tested. By feeding the sentence *"A person waves with its right hand"* and *"A person waves with the left hand"* which are included in the training dataset, the results of the real gestures and the generated ones are shown in Figs. 4 and 5, respectively. It can be seen that the motions produced by our proposed network are similar to the training data. However, the corresponding pair of poses on each frame is different, indicating that our G model does not simply memorize and reproduce the training data.

Our experimental results indicated that the embedding vectors well captured the meaning of description, and the conditional input efficiently controlled the generated data. The following texts *"Someone over there is waving with their both two hands"* and *"They are taking a deep bow to show their respect"* were modified from the original descriptions while keeping their meaning of *"waving both hands"* and *"make a bow"* intact. The produced gesture in Fig. 8a looks like a person waves with his/her two hands. Similarly, Fig. 8b represents a sequence of frames as a person is collapsing their body downward while the arms are kept lower than the hip. The action is ended with an upright body posture. In the training data, the joint positions were constructed with respect to the top-chest coordinates. As a result, the Generator G always tries to keep the position of top-chest the same throughout the frame sequence, as shown in Fig. 8b.

For the quantitative evaluation of the generated actions, starting from Epoch 800, the descriptions d from the testing data were fed into the Generator model. The generated motion a_f and the ground truth sample a_r were plugged into Eq. 5 and Eq. 6 for measuring their similarities. Additionally, instead of representing the actions by the 3-channel matrix as mentioned in II, we tested the proposed model with a single channel approach. Specifically, the action a_r and a_f were described by a 2D matrix. Here, the horizontal axis captures the time sequence of action, while the vertical axis covers the human joints in the x, y, z coordinates. With this representation of action, the output layer of G and the input layer of D in Fig. 1 were modified to a single-channel matrix, while keeping the other parameters of the network remain

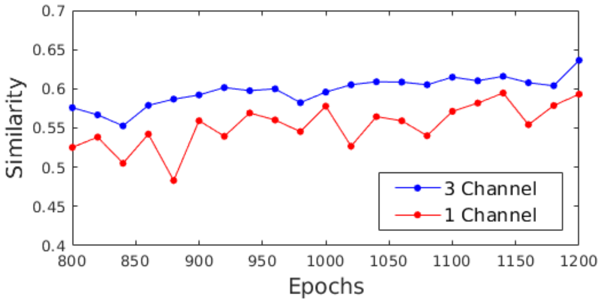


Fig. 6: The average similarity between the generated and real actions with the 1 channel and 3 channel models.

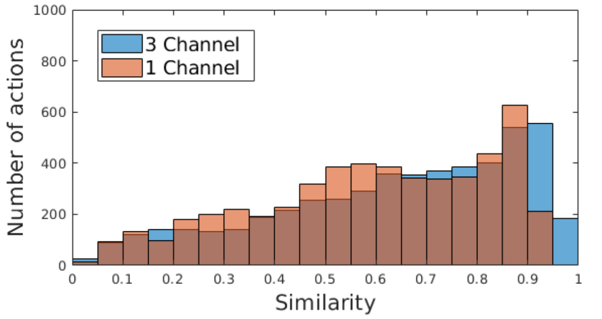


Fig. 7: The distribution of similarity values between generated and real gestures at Epoch $1,200$.

unchanged. It is then followed by the same training and testing procedures as conducted with the 3-channel approach. Fig. 6 shows the average similarity score between the real actions of testing data and generated actions produced by the single and 3 channel approaches.

Overall, the performance of both models was improved over the training time. However, the actions created from the 3-channel network always yield higher accuracy than the single-channel approach. Indeed, by representing the motion by 3 channels corresponding to its 3D Cartesian coordinates, the similarity score gradually increases over the training period. Fig. 7 shows a closer look at Epoch $1,200$ for the distribution of similarity scores. The result indicated that the 3-channel network produces actions more similar to the real samples than the single-channel system. Thus, it is convinced that the 3-channel network captures the spatial and temporal information of actions better than the single-channel method. The reason is that by separating the 3D joints into the individual coordinates, the spatial relations between them could be detected faster than the single-channel representation. Consequently, the Generator could receive more informative feedback for optimizing its generated data.

F. Transforming Generated Gesture into the Target Robot

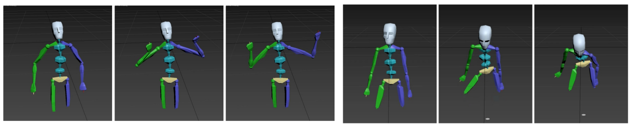
The Generator network produces actions defined in the 3D Cartesian space. Through the transformation model, it can be converted to a set of corresponding joint angles subject to the target robot physical constraints. Figs. 8a and 8b show the generated actions defined by the human joint positions, while Figs. 9a and 9b represent the corresponding gestures of the Pepper robot. In order to evaluate the quality of the robot's gestures generated by our approach compared to the Pepper robot's NAOqi API (as a baseline for comparison) [1], the same input descriptions were fed to both systems. The generated action are shown in Figs. 10a and 10b, respectively.

As can be seen from Fig. 9a, the Pepper robot's action well expressed the original meaning represented by the skeleton model in Fig. 8a. In order to synchronize with the text description *"Someone over there is waving with their both two hands"*, the Pepper robot is gradually moving its two hands over the shoulder and then waving. In contrast, as shown in Fig. 10a, the Pepper robot's NAOqi API *ALAnimatedSpeech* produced only a slight hand movement to support the given sentence, which was a difficult sign to understand. Fig. 9b shows the generated robot's gesture from the proposed approach given the text *"They are taking a deep bow to show their respect"*. The result shows that the action looks like Pepper is collapsing its upper body while its two hands remained unchanged. Compared to the gestures produced by our proposed approach, most of the actions produced by *ALAnimatedSpeech* are not related to the text description. The motions shown in Fig. 10b can be taken in such a way that a person is describing something, which may lead to misperception. The experimental results showed that the connection between the Pepper's handcrafted gestures and the given text does not fitting the situation due to the fact that the NAOqi API uses a set of gestures manually designed by animation experts. It is often the case that the Pepper robot's gestures are randomly launched if some specific keywords are not detected from the given text. As a result, the generated actions shown in Fig. 10a and Fig. 10b are not significantly correlated with the given sentence. Instead of focusing on specific keywords, in our approach, the encoded vectors capture the semantics and syntax of the whole sentence. This information is used as the determining condition to control the generated gestures. Therefore, the robot's gestures generated by our approach more appropriately fits the input descriptions.

IV. Conclusion and Future Work

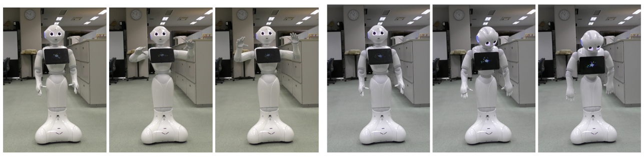
In this paper, we proposed a new model of generating communicative gestures for social robots. The model was based on a CGAN constructed by the convolutional network. Our approach receives speech text as the determining condition to generate co-speech actions. To demonstrate the validity of the proposed approach, the model was trained on the publicly available dataset. The experimental results indicated that our model could imitate the human joints distribution from the training data and generate human-like gestures. The generated actions efficiently emphasize the meaning of input description and ensure that the actions and the ground truth data are as similar as possible. The generated motions were then transformed into the target robot motions. In a series of real robot experiments, it was shown that the communicative robot gestures created by our

[1] http://doc.aldebaran.com/2-5/naoqi/audio/alanimatedspeech-api.html



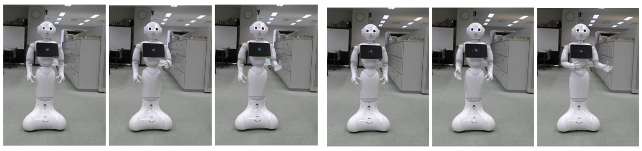
(a) *"Someone over there is waving with their both two hands"* (b) *"They are taking a deep bow to show their respect"*

Fig. 8: Key poses of actions by the G network.



(a) *"Someone over there is waving with their both two hands"* (b) *"They are taking a deep bow to show their respect"*

Fig. 9: Pepper's key poses by the proposed model.



(a) *"Someone over there is waving with their both two hands"* (b) *"They are taking a deep bow to show their respect"*

Fig. 10: Pepper's key poses by the NAOqi API.

model more appropriately fit the given input sentence than those produced by the robot's existing module available onboard. By considering the relation between human gestures and natural language for generating social robot's actions, it is believed that the generated gestures could be more understandable and acceptable to the general public. In our future work, the effect of emotions on the communicative gestures will be investigated.

ACKNOWLEDGMENT

This work was supported by the EU-Japan coordinated R&D project on "Culture Aware Robots and Environmental Sensor Systems for Elderly Support" commissioned by the Ministry of Internal Affairs and Communications of Japan and EC Horizon 2020 Research and Innovation Programme under grant agreement No. 737858. The authors are also grateful for financial supports from the Air Force Office of Scientific Research (AFOSRAOARD/FA2386-19-1-4015).

REFERENCES

[1] M. L. Knapp, J. A. Hall, and T. G. Horgan, *Nonverbal Communication in Human Interaction*. Cengage Learning, 2013.

[2] C. L. Breazeal, *Designing Sociable Robots*. MIT press, 2002.

[3] M. Salem, F. Eyssel, K. Rohlfing, S. Kopp, and F. Joublin, "To err is human(-like): Effects of robot gesture on perceived anthropomorphism and likability," *International Journal of Social Robotics*, vol. 5, no. 3, pp. 313–323, 2013.

[4] B. Bruno, N. Y. Chong, H. Kamide, S. Kanoria, J. Lee, Y. Lim, A. K. Pandey, C. Papadopoulos, I. Papadopoulos, F. Pecora, A. Saffiotti, and A. Sgorbissa, "Paving the way for culturally competent robots: a position paper," in *IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2017, pp. 553–560.

[5] J. Cassell, H. H. Vilhjálmsson, and T. Bickmore, "Beat: The behavior expression animation toolkit," in *Life-Like Characters*. Springer, 2004, pp. 163–185.

[6] S. Marsella, Y. Xu, M. Lhommet, A. Feng, S. Scherer, and A. Shapiro, "Virtual character performance from speech," in *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2013, pp. 25–35.

[7] A. Shimazu, C. Hieida, T. Nagai, T. Nakamura, Y. Takeda, T. Hara, O. Nakagawa, and T. Maeda, "Generation of gestures during presentation for humanoid robots," in *IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2018, pp. 961–968.

[8] M. Plappert, C. Mandery, and T. Asfour, "Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks," *Robotics and Autonomous Systems*, vol. 109, pp. 13–26, 2018.

[9] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[11] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[12] Y. Yan, J. Xu, B. Ni, W. Zhang, and X. Yang, "Skeleton-aided articulated motion generation," in *ACM International Conference on Multimedia*. ACM, 2017, pp. 199–207.

[13] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3d action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3288–3297.

[14] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *Advances in Neural Information Processing Systems*, 2015, pp. 3294–3302.

[15] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[16] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[17] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter Semester*, vol. 2014, no. 5, p. 2, 2014.

[18] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances In Neural Information Processing Systems*, 2016, pp. 613–621.

[19] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," *arXiv preprint arXiv:1802.04208*, 2018.

[20] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning*, 2010, pp. 807–814.

[21] N. T. V. Tuyen, S. Jeong, and N. Y. Chong, "Emotional bodily expressions for culturally competent robots through long term human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2018, pp. 2008–2013.

[22] C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "The kit whole-body human motion database," in *International Conference on Advanced Robotics*. IEEE, 2015, pp. 329–336.

[23] M. Plappert, C. Mandery, and T. Asfour, "The kit motion-language dataset," *Big Data*, vol. 4, no. 4, pp. 236–252, 2016.

[24] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," in *IEEE International Conference on Computer Vision*, 2015, pp. 19–27.

[25] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations," in *International Joint Conference on Artificial Intelligence*, vol. 13, 2013, pp. 2466–2472.

[26] N. T. V. Tuyen, S. Jeong, and N. Y. Chong, "Learning human behavior for emotional body expression in socially assistive robotics," in *International Conference on Ubiquitous Robots and Ambient Intelligence*. IEEE, 2017, pp. 45–50.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.