A High-Fidelity Open Embodied Avatar with Lip Syncing and Expression Capabilities

Deepali Aneja

deepalia@cs.washington.edu University of Washington Seattle, Washington

Daniel McDuff

damcduff@microsoft.com Microsoft Research Redmond, Washington

Shital Shah

shitals@microsoft.com Microsoft Research Redmond, Washington

ABSTRACT

Embodied avatars as virtual agents have many applications and provide benefits over disembodied agents, allowing nonverbal social and interactional cues to be leveraged, in a similar manner to how humans interact with each other. We present an open embodied avatar built upon the Unreal Engine that can be controlled via a simple python programming interface. The avatar has lip syncing (phoneme control), head gesture and facial expression (using either facial action units or cardinal emotion categories) capabilities. We release code and models to illustrate how the avatar can be controlled like a puppet or used to create a simple conversational agent using public application programming interfaces (APIs). GITHUB link: https://github.com/danmcduff/AvatarSim

CCS CONCEPTS

• Human-centered computing → Interactive systems and tools; • Computing methodologies → Animation; Machine learning.

KEYWORDS

avatars; multimodality; conversational systems; embodied agents; expression retargeting.

ACM Reference Format:

Deepali Aneja, Daniel McDuff, and Shital Shah. 2019. A High-Fidelity Open Embodied Avatar with Lip Syncing and Expression Capabilities. In 2019 International Conference on Multimodal Interaction (ICMI '19), October 14-18, 2019, Suzhou, China. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3340555.3353744

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. ICMI '19, October 14-18, 2019, Suzhou, China

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-6860-5/19/05. https://doi.org/10.1145/3340555.3353744

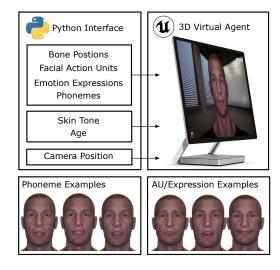


Figure 1: We present an open embodied avatar with lip syncing and expression capabilities that can be controlled via simple python interface. We provide examples of how to combine this with publicly available speech and dialogue APIs to construct a conversational embodied agent.

1 INTRODUCTION

Virtual agents have many applications, from assistance to companionship. With advancements in machine intelligence, computer graphics and human-computer interaction, agents are becoming more and more capable of real-time interaction with the user, synthesizing speech and expressions, style matching in conversations, tracking and completing tasks as assistants. How we represent an intelligent system via a user interface has implications for how the user perceives the system and associates or attributes intelligence with it. One line of research emphasizes that embodied agents offer several advantages over non-embodied dialogue systems. An agent that has a physical presence means that the user can look at it. Cassell [3] argues that embodiment can help locate intelligence for users and reconcile both interactional and propositional intelligence. Embodied agents have greater expressive capabilities and naturally embodied agents have been used to explore the role of social signals [4] and emotions [21] in human-computer interaction.

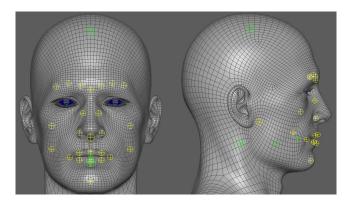


Figure 2: The locations of controllable bone positions on the avatar. Each position can be controlled with 6 degrees of freedom. See Table 1 for descriptions.

Improvements in speech recognition [28], dialogue generation [12, 24], emotional speech synthesis [17, 26] and computer graphics have made it possible to design more expressive and realistic conversational agents. However, there are still many uncertainties in how best to design embodied agents. Building a state-of-the-art system for performing research to address these questions is non-trivial.

Open code, data, and models democratize research, helping advance the state-of-the-art by enabling a researcher to build upon existing work. LUCIA [14] is an MPEG-4 facial animation system that can copy a real human being by reproducing the movements of passive markers positioned on his face. Several architectures have been proposed for designing and building embodied conversational agents (ECA). AVATAR [23] is an architecture to develop ECAs based on open source tools and libraries. FACSvatar [27] processes and animates action units in real-time. The framework is designed to be modular making it easy to integrate into other systems and has low latency allowing real-time animations. FACSHuman [9] is a similar tool that allows animation of a 3-dimensional avatar using the facial action coding system (FACS)[5]. The system, however, does not allow for easily constructing real-time multimodal ECAs.

These systems have been very valuable. However, there are some limitations. First, driving the non-verbal behavior of the agent is often not simple. For example, LUCIA [14] requires the user to wear reflective markers. In [18], people rated the realistic characters or very cartoon-like characters as more appealing or pleasant, but not characters which lie in the middle of 'abstract to realistic' scale. GRETA is an autonomous and expressive agent that is somewhat realistic in appearance and has numerous capabilities for social and emotional expression in addition to dialogue [22]. We present a freely available high-fidelity embodied avatar that has lip syncing, head gesture and facial expression capabilities. It can be controlled via a simple python interface, using either

bone positions, action units or basic expressions. The goal is to make it easier for people to develop ECAs.

The contributions of this paper are: (1) to present a high-fidelity open embodied avatar that is capable of lip syncing and facial expression control via a simple python interface, using either bone positions, FACS or basic expressions, (2) to provide an example of an end-to-end conversational agent interface via publicly available APIs, and (3) to show how expression transfer can be used to control the avatar.

2 THE AVATAR

We created the avatar within the AirSim [25] environment using Unreal Engine 4 (UE4) [8]. The avatar, a head, and torso is placed in a room with multiple light sources (windows, overhead lights, and spotlights). Pictures of the avatar are shown in Figure 1. The avatar's appearance can be manipulated in several ways. The position and orientation of 38 "bone" positions, which are facial locations (or landmarks) can be controlled, each with six degrees of freedom. We also created 24 FACS action unit presets [5] and 19 phonetic mouth shape presets (visemes) for the avatar. Each of these variables can be controlled in real-time using a simple Python interface. Once the avatar executable is running and a client connection formed between the python instance the positions of bones, the action unit and phoneme presets or the location of the camera can be updated as frequently as needed. The avatar controls are described as follows:

Bone Position Controls. The position (x_b, y_b, z_b) and orientation $(\sigma_b 1, \sigma_b 2, \sigma_b 3)$ of each of the 38 bone positions can be set. Figure 2 shows the location of the bone positions and Table 1 lists their names. The green points indicate joints that control rigid head and jaw motions. The yellow points indicate landmarks that control specific facial regions (e.g., eyebrows). The bones positions can be referred to via these names (e.g., LUpperEyelid for the left upper eyelid) or by their integer as specified in our documentation.

Facial Action Controls. FACS is the mostly widely used and objective taxonomy for describing facial behavior. Therefore, in addition to control of bone positions the avatar has controls for 24 facial action unit presets (see Table 1).

Expression Controls. A set of basic emotional expressions can be created via combinations of action units (e.g., using EMFACS [7]) or using expression retargeting (Section 3). The avatar is also capable of blinking the eyes by controlling the upper and lower eyelid bone position controls.

Lip Syncing Controls. For conversational applications lip syncing is enabled via phoneme control. In our example scripts, we show how a phoneme classifier [13] can be first

Table 1: The avatar has bone position, facial action unit and phoneme controls. M=middle, L=left, R=right, U=upper, L=lower, C=corner, I=inner, O=outer

Bone Positions	Action Units	Phonemes
U. Cheek (L & R)	1: I. Brow Raiser	ae
Mid U. Lip (M, L & R)	2: O. Brow Raiser	b
Mid L. Lip (M, L & R)	4: Brow Lowerer	d
Lip Corner (L & R)	5: U. Lid Raiser	ee
U. Nose (M, L & R)	6: Cheek Raiser	ngnk
O. Brow (L & R)	7: Lid Tightener	eh
Mid Brow (L & R)	9: Nose Wrinkler	ar
I. Brow (L & R)	10: U. Lip Raiser	f
Nostril (L & R)	11: Nasolabial Deep.	g
Cheek Dimple (L & R)	12: Lip C. Puller	h
Eye (L & R)	13: Sharp Lip Puller	i
U. Eyelid (L & R)	14: Dimpler	k
L. Eyelid (L & R)	15: Lip C. Depressor	1
Jaw	16: L. Lip Depressor	m
Chin	17: Chin Raiser	n
Tongue Base	18: Lip Pucker	oh
Tongue Tip	20: Lip Stretcher	000
Chest	22: Lip Funneler	oh
Neck	23: Lip Tightener	m
Head	24: Lip Pressor	
Root	25: Lips Part	
Hair	26: Jaw Drop	
	27: Mouth Stretch	
	28: Lip Suck	

used to identify phonemes from an audio segment and then used to control the lips in synchrony with the audio playback. Nineteen phoneme presets are included, while there are 44 phonemes in English the 19 presets can be used quite effectively for most dialogue.

Head Gesture and Appearance Controls. The head rotations can be varied continuously by controlling the yaw, pitch and roll between -1.0f to 1.0f. The skin tone and age of the avatar can be controlled via independent parameters. The skin tone varied from light (0) to dark (1), and the skin age texture can be varied from youthful (0) to old (1).

Camera Controls. The camera position within the virtual environment that governs the perspective from which the agent will be captured can be controlled with six degrees of freedom (position (x_c, y_c, z_c) and orientation ($\sigma_c 1, \sigma_c 2, \sigma_c 3$)).

3 APPLICATIONS

Conversational Agent

To illustrate how the avatar can be used as a conversational agent, we provide an end-to-end pipeline for conversational

dialogue, similar to [11]. The pipeline uses public APIs for natural language generation and speech synthesis from Microsoft. We created a demonstration script that calls a natural language API to generate utterances and then uses a text-to-speech service to return a synthesized voice. We convert this voice into a sequence of visual groups of phonemes (visemes) using PocketSphinx [13]. Finally, we synchronously play the audio and drive the avatar using the phoneme presets to achieve a simple lip syncing example of a conversational agent. Python scripts for running this conversational agent are provided with the agent software.

Facial Expression Retargeting

In virtual embodied agent interactions, whether for conversation or not, expressions and head motions are important for creating a more natural and lifelike agent. Our next example shows how expressive faces can be generated. We provide two interfaces for driving the expressions of the avatar (see Figure 3), one using bone controls and the other using the FACS presets. In both cases, we utilize existing deep learning based frameworks to perform the expression transfer i.e., synthesizing the avatar's face with the expressions of a human subject in the input video. This creates the effect of facial expression mimicry. We provide scripts and a FACS-based trained model for systematically generating faces. The facial expressions on the avatar can be synthesized as follows:

Expression Synthesis via Facial Action Units. We describe an end-to-end pipeline to retarget a human's facial expression to the avatar's face using facial action unit controls as shown in Figure 3 (A). The pipeline utilizes a webcam attached to the computer and processes the video frames. First, we detect the face in the input video. We use an open source HoG-based face detector [15]. Next, we analyze the facial region of interest (ROI), defined by the face detection bounding box, using the Facial Action Unit Recognizer. The output from the Action Unit Recognizer is used to synthesize the expressions on the avatar's face by the Expression Synthesizer.

Facial Action Unit Recognizer. We trained a Convolutional Neural Network (CNN) [10] on a large facial expression dataset [6] for recognizing 12 facial action units - 1, 2, 4, 5, 6, 9, 12, 17, 20, 25, 26 (refer to Table 1 for descriptions) and 43 (Eyes Closed). The Action Unit Recognizer returns a 12-dimensional feature vector representing the probability of each action unit. The model was trained using PyTorch [20] to minimize the binary cross-entropy loss with an average F1-score of 0.78. More details about the training and evaluation of the model can be found in [16]. Alternatively, the avatar could be used with an off-the-shelf FACS AU detection SDK, such as [19].

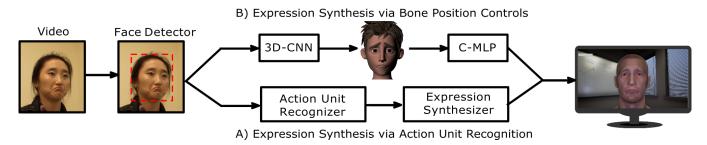


Figure 3: We present two pipelines for retargeting an expression from a human to the avatar. A) Expression Synthesis via Action Units Controls. Our pipeline takes a human video as an input, recognizes 12 FACS Action Units from the user's detected face and synthesizes the same expression on the avatar's face. B) Expression Synthesis via Bone Position Controls. Our pipeline takes a human video as an input, detects the user's face, generates 3D parameters on the primary character 'Ray' through a 3D-CNN and synthesizes the same expression on the avatar's face. See Section 3.2 for details for the Action Unit Recognizer, Expression Synthesizer, 3D-CNN and Character Multi-layer Perceptron (C-MLP).

Expression Synthesizer. We use the facial action unit probability scores from the Facial Action Unit Recognizer to control the avatar's facial action controls and mimic the human facial expressions. The raw probability scores can be used directly to drive the actions, or these values can be smoothed or rounded first. Python scripts for running the expression synthesis via facial action units are provided with the agent software along with the trained Action Unit Recognizer model.

Expression Synthesis via Bone Position Controls. We describe an end-to-end pipeline to retarget a human's facial expression to the avatar's face using the bone position controls as shown in Figure 3 (B). As with the previous approach face detection is first performed on the input video and the bounding box used to define the ROI. We then use a multi-stage deep learning system ExprGen [1] to retarget facial expressions from the human to the avatar. The pipeline takes 2D images of human facial expressions as input and generates 3D parameters of the avatar. We have two separate components in our pipeline: Human to Primary Character expression transfer and Primary Character to Avatar expression transfer.

Human to Primary Character transfer. The 3D-CNN produces parameters for a human to a primary character expression transfer in 3D including both perceptual and geometric similarity as shown on 'Ray' character in Figure 3 (B) (middle).

Primary Character to Avatar transfer. We trained a Character Multi-layer Perceptron (C-MLP) to transfer the expression of the primary character to the avatar. The C-MLP was trained to minimize the mean square error loss between the primary character parameters and the avatar parameters. More details about training the model can be found in [1, 2].

Some of the limitations of our work are: The intensity of FACS action units is not validated, and the synthesis of composite facial expressions using multiple facial action units

along with phoneme based lip syncing is currently rule-based and not validated for expression clarity. When using the avatar as a conversational agent along with expression retargeting, it can result in implausible facial movement due to the conflict between lip syncing controls and expression controls. To avoid such a scenario, we control the expressiveness of the avatar's upper face only (above the mouth) while lip syncing. However, the avatar allows for exploration of more advanced solutions to this problem. In future research, we can use the Facial Action Coding System (FACS) and/or perceptual studies to evaluate the intensity of the retargeted avatar facial expression.

4 ACCESS AND CITATION

The *avatar* is available for distribution to researchers online. The avatar, code, models and software information can be found at https://github.com/danmcduff/AvatarSim. The software is released under the MIT license and a responsible AI license¹. This allows the use of the software without restriction, including without limiting rights to use, copy or modify it, with the exception of a set of explicit applications. If you find this software helpful or use it in research, please cite this paper. Citation information can be found on the GITHUB page.

5 CONCLUSION

We present an open high-fidelity embodied avatar with bone, action unit, and phoneme controls. We release code and models to illustrate how the avatar can be controlled like a puppet, via expression retargeting or used to create a simple conversational agent using public APIs. We hope that this resource enables researchers to advance the state of the art in high-fidelity embodied virtual agents. Contributions to the code repository are welcomed.

¹https://www.licenses.ai/

REFERENCES

- [1] Deepali Aneja, Bindita Chaudhuri, Alex Colburn, Gary Faigin, Linda Shapiro, and Barbara Mones. 2018. Learning to generate 3D stylized character expressions from humans. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 160–169.
- [2] Deepali Aneja, Alex Colburn, Gary Faigin, Linda Shapiro, and Bar-bara Mones. 2016. Modeling stylized character expressions via deep learning. In Asian conference on computer vision. Springer, 136–153.
- [3] Justine Cassell. 2001. Embodied conversational agents: representation and intelligence in user interfaces. *AI magazine* 22, 4 (2001), 67–67.
- [4] Justine Cassell, Joseph Sullivan, Elizabeth Churchill, and Scott Prevost. 2000. *Embodied conversational agents*. MIT press.
- [5] Paul Ekman. 1972. Universals and cultural differences in facial expressions of emotion. In *Nebraska Symposium on Motivation*, J Cole (Ed.). University of Nebraska Press, Lincoln, 207–282.
- [6] C Fabian Benitez-Quiroz, Ramprakash Srinivasan, and Aleix M Martinez. 2016. Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 5562–5570.
- [7] Wallace V Friesen, Paul Ekman, et al. 1983. EMFACS-7: Emotional facial action coding system. *Unpublished manuscript, University of California at San Francisco* 2, 36 (1983), 1.
- [8] Epic Games. 2007. Unreal engine. Online: https://www. unrealengine. com (2007).
- [9] Michaël Gilbert, Samuel Demarchi, and Isabel Urdapilleta. 2018. FAC-SHuman a Software to Create Experimental Material by Modeling 3D Facial Expression. In Proceedings of the 18th International Conference on Intelligent Virtual Agents. ACM, 333–334.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE* conference on computer vision and pattern recognition. 770–778.
- [11] Rens Hoegen, Deepali Aneja, Daniel McDuff, and Mary Czerwinski. 2019. An End-to-End Conversational Style Matching Agent. arXiv preprint arXiv:1904.02760 (2019).
- [12] Bernd Huber, Daniel McDuff, Chris Brockett, Michel Galley, and Bill Dolan. 2018. Emotional dialogue generation using image-grounded language models. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. ACM, 277.
- [13] David Huggins-Daines, Mohit Kumar, Arthur Chan, Alan W Black, Mosur Ravishankar, and Alexander I Rudnicky. 2006. Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, Vol. 1. IEEE, I–I.
- [14] CINECA-Consorzio Interuniversitario. [n. d.]. LUCIA: An open source 3D expressive avatar for multimodal hmi. ([n. d.]).
- [15] Davis E. King. 2009. Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research 10 (2009), 1755–1758.

- [16] Beibin Li, Sachin Mehta, Deepali Aneja, Claire Foster, Pamela Ventola, Frederick Shic, and Linda Shapiro. 2019. A Facial Affect Analysis System for Autism Spectrum Disorder. arXiv:arXiv:1904.03616
- [17] Shuang Ma, Daniel Mcduff, and Yale Song. 2018. A generative adversarial network for style modeling in a text-to-speech system. (2018).
- [18] Rachel McDonnell, Martin Breidt, and Heinrich H Bülthoff. 2012. Render me real?: investigating the effect of render style on the perception of animated virtual humans. ACM Transactions on Graphics (TOG) 31, 4 (2012), 91.
- [19] Daniel McDuff, Abdelrahman Mahmoud, Mohammad Mavadati, May Amr, Jay Turcot, and Rana el Kaliouby. 2016. AFFDEX SDK: a crossplatform real-time multi-face expression recognition toolkit. In Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems. ACM, 3723–3726.
- [20] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).
- [21] Catherine Pelachaud and Isabella Poggi. 2002. Subtleties of facial expressions in embodied agents. The Journal of Visualization and Computer Animation 13, 5 (2002), 301–312.
- [22] Isabella Poggi, Catherine Pelachaud, Fiorella de Rosis, Valeria Carofiglio, and Berardina De Carolis. 2005. Greta. a believable embodied conversational agent. In Multimodal intelligent information presentation. Springer, 3–25.
- [23] Marcos Santos-Pérez, Eva González-Parada, and José Manuel Cano-García. 2011. AVATAR: An Open Source Architecture for Embodied Conversational Agents in Smart Environments. In International Workshop on Ambient Assisted Living. Springer, 109–115.
- [24] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*.
- [25] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. 2018. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In Field and service robotics. Springer, 621–635.
- [26] RJ Skerry-Ryan, Eric Battenberg, Ying Xiao, Yuxuan Wang, Daisy Stanton, Joel Shor, Ron J Weiss, Rob Clark, and Rif A Saurous. 2018. Towards end-to-end prosody transfer for expressive speech synthesis with tacotron. arXiv preprint arXiv:1803.09047 (2018).
- [27] Stef van der Struijk, Hung-Hsuan Huang, Maryam Sadat Mirzaei, and Toyoaki Nishida. 2018. FACSvatar: An Open Source Modular Framework for Real-Time FACS based Facial Animation. In Proceedings of the 18th International Conference on Intelligent Virtual Agents. ACM, 159–164.
- [28] Wayne Xiong, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. 2018. The Microsoft 2017 conversational speech recognition system. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 5934–5938.