Learn2Smile: Learning Non-Verbal Interaction Through Observation

Will Feng[1], Anitha Kannan[1], Georgia Gkioxari[1], C. Lawrence Zitnick[1]

***Abstract*— Interactive agents are becoming increasingly common in many application domains, such as education, healthcare and personal assistance. The success of such embodied agents relies on their ability to have sustained engagement with their human users. Such engagement requires agents to be socially intelligent, equipped with the ability to understand and reciprocate both verbal and *non-verbal* cues. While there has been tremendous progress in verbal communication, mostly driven by the success of speech recognition and question-answering, teaching agents to appropriately react to facial expressions has received less attention.**

In this paper, we focus on non-verbal facial cues for face-to-face communication between a user and an embodied agent. We propose a method that automatically learns to update the agent's facial expressions based on the user's expressions. We adopt a learning scheme and train a deep neural network on hundreds of videos, containing pairs of people engaging in a conversation, and without external human supervision. Our experimental results show the efficacy of our model in sustained long-term prediction of the agent's facial landmarks. We present comparative results showing that our model significantly outperforms baseline approaches and provide insightful human studies to better understand our model's qualitative performance. We release our dataset to further encourage research in this field.

I. INTRODUCTION

A long-standing goal of robot intelligence is to enable real-time interactions with humans. This includes interaction through dialog [1], [2], [3], and collaborative task completion or training [4], [5]. Naturally, human interaction occurs through a variety of modalities [6]; speech, vision, touch, etc. From a very early stage humans show responses to their surroundings [7], [8], one of which is through their facial expressions [9]. Facial expressions convey a wealth of information about a human's internal emotional state. This non-verbal form of communication is known to be important for sustained human interaction [2], [3]. For instance, user studies concluded that agents with behavioral displays such as eye blinks are perceived as more intelligent and more capable [3]. Tasks such as turn-taking are also informed from non-verbal cues such as eye gaze [10], [11]. As progress in developing interactive agents advances, the need for effective non-verbal communication becomes imperative.

In this paper, we explore how to interact with humans using visual expression cues. In particular, we learn to predict appropriate responses to a user's facial expressions *through observation*. For this, we use hundreds of videos with pairs of people engaging in a conversation without any external human supervision. This is unlike previous approaches [12], [13] that explicitly label emotional states, such as happy, sad, surprised etc. In our work, we train a deep neural network to predict the agent's expressions conditioned on the user's expressions, while eliminating the expensive step of manual supervision. No doubt, our approach could be further improved by using contextual cues from the conversational content of the interaction, e.g. in the form of audio signal or transcribed text. However, in this work we choose to focus on the direct expression-to-expression approach, to serve as a fundamental building block of the final system.

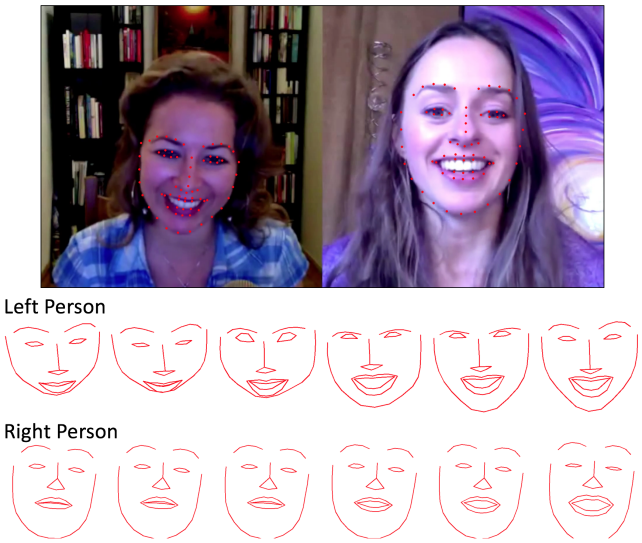

Fig. 1: When communicating, humans express themselves using both verbal and non-verbal cues. (top) Non-verbal facial cues are captured during a conversation using two front facing cameras. (bottom) Facial keypoints can be tracked through time to train a system to respond to non-verbal cues.

The success of a predictive model in forecasting an agent's appropriate expression lies in learning to recognize the subtle facial expressions of the user. Our training data contains a variety of facial expressions, such as talking, laughing and cringing. Even though the appearances of individuals in our dataset differ, their expressions share similarities which can be extracted from the configuration of their facial landmarks. For example, when people cringe the configuration of their eyebrows and mouth is most revealing about their emotional state. Indeed, small variations in expression can be very informative [14]. In order to capture those subtleties, we extract the 2D locations of facial landmarks from all the video frames in our dataset, and only use the landmark keypoints as input to the model. This allows our model to

[1]All authors are with Facebook AI Research. Emails: {willfeng, akannan, gkioxari, zitnick}@fb.com

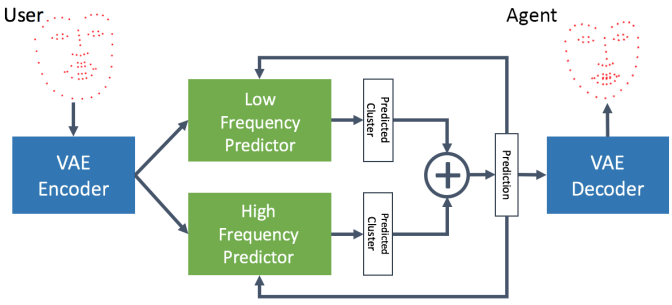

Fig. 2: Overview of our model: The 2D keypoints tracked on the user's face are encoded using the VAE encoder. Next, the clusters representing the agent's low and high frequency facial movements for the next 15 frames are predicted. The agent's facial expressions for the next 15 frames are computed by summing the means of the predicted clusters. The predicted facial expressions are then fed back into the model to predict the future sequence. Finally, the VAE decoder transforms the predicted encoded keypoints into the agent's 2D facial keypoints that are used for display.

focus on subtle responses, such as microexpressions [15].

Our model takes as input the 2D keypoints tracked on the user's face. The locations of the 2D keypoints are encoded using a Variational AutoEncoder (VAE) [16]. A novel deep neural network is used to predict the facial expression of an agent in response to the facial expression of a user. Given a history of encoded facial expressions from both the user and the agent, the neural network predicts both the high and low frequency changes in the agent's expression at each time period. This allows the network to learn movements of lower frequency, such as head nods, and high-frequency movements such as eye blinks and mouth movements when speaking. For training, we use the facial keypoints of pairs of people talking to each other in videos, as shown in Figure 1. No other information is used during training, such as explicit labels of emotion [18]. Figure 2 shows an overview of our approach.

We evaluate our approach on a held out test set of videos containing humans engaging in a conversation. We pick one person to play the role of the user, and the second to play the role of the agent. We show that our approach significantly outperforms several baseline approaches in predicting the agent's facial expressions. In addition, we present several ablation experiments, which highlight the importance of our system's components. Given the ambiguity in responses regarding facial expressions, we run human studies to evaluate whether the generated facial expressions appear to respond to the user in a realistic manner. We demonstrate that our approach is capable of making predictions that are quantitatively close to the facial movements made by humans, while also producing results that are qualitatively similar.

II. RELATED WORK

Facial expression and recognition. Facial Action Coding System [19] enumerates all the action units "AU" that cause facial movements. In fact, any facial expression can be represented as a combination of these AUs. A number of methods have been proposed to either classify facial expression into their AU units ([20]) or into a small number of protoypical facial expressions corresponding to dominant emotions such as happiness and anger ([21]). For detailed survey, we refer the reader to [12], [13].

In the context of human-agent interaction, existing methods predominantly recognize AU units from the human's facial expression to update the expression of the agent [22], [23], [24], [25]. Our goal in this paper is a continuous update of the agent's expression that is both responsive to the user and is consistent with the agent's expression in time. Therefore, we refrain from explicitly categorizing the user's expression and instead directly predict how the agent's facial expression needs to be adjusted based on its own and the user's past history.

Social robots. The importance of visual cues such as facial expression, gaze and gesture for sustained interactivity between humans and agents is well studied ([26], [27] for survey). Of particular relevance is the work in modulation of agent's facial expression during an interaction. Existing works view this through the lens of affect, with the goal of emoting back to the user's affect. Hence, a common approach is to infer user's affect using facial expression analysis and often use deterministic rules to change the agent's expression [22], [23], [28]. More recently, affective policies are being learned from data. In [24], affective policy for robot tutor Tego was learned by combining student's performance in a task with their inferred affect. Levine et al. [18] studied the related problem of body motion given speech, in which HMMs are used to drive an agent's pose.

Our work differs from this line of work in two important ways: first, instead of explicitly inferring the affect, our model *directly* predicts the expression of the agent in response to the user. This enables capturing valuable nuanced microexpressions that are usually lost when expression is categorized into discrete emotional states. Secondly, current approaches are reactionary and instantaneous: the current state of the user's facial expression drives the expression of the agent. In contrast, our approach models temporal dynamics, where the human and the agent are continuously engaged in a conversation, and hence the agent's facial expression is a function of both, the user's and the agent's past expressions.

III. APPROACH

The goal of our model is to predict the change in expression of an agent in response to a user. Designing such a predictive model has several challenges. First, the number of facial expressions and poses is exponential in the number of keypoints, and not all variants are seen in the training data. To improve generalization, we embed the space of keypoints using a VAE such that the inferred stochastic latent variables capture the core underlying keypoint control knobs. Second, different facial expressions change with different frequency. For instance, eye blinks occur very quickly, whereas, head

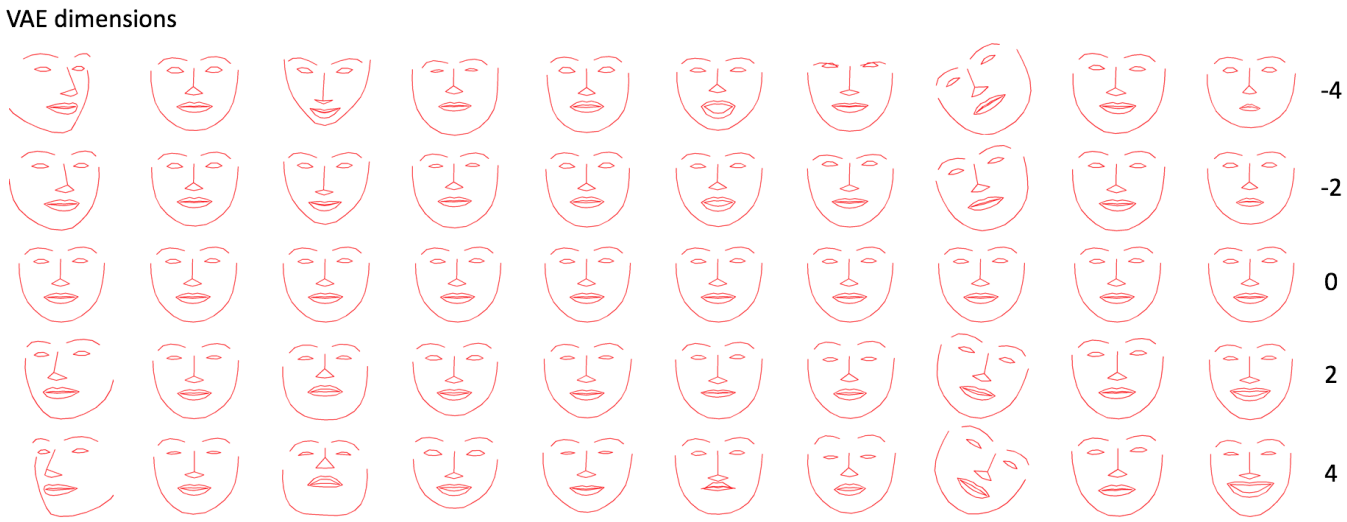

Fig. 3: Illustration of the effect of changing values in 10 of the VAE dimensions from -4 to 4. Notice that each dimension affects a different aspect of the face, e.g., face orientation, opening the mouth, opening the eyes, smiling, etc.

nods occur more slowly. To cope with the problem of capturing expressions at different frequency scales, our model predicts low and high frequency components separately, as shown in Figure 2. Finally, predictive models with continuous output spaces are known to suffer from the regression-to-mean problem, i.e., the prediction favors the mean of the output data it has seen. To avoid this, we propose a predictor model with a discrete output space quantized by clustering.

Next, we describe how we encode and decode the 2D facial keypoints using a VAE, the process for discretizing the output space, and our final predictive model.

A. *Representing the 2D facial keypoints*

We encode the 2D facial landmarks of each face frame using VAE [16]. VAE pairs a top down generative model or decoder with a bottom up recognition network or encoder for amortized probabilistic inference. Both the encoder and decoder are jointly trained to maximize a variational lower bound on the data likelihood.

In our setting, we represent the agent's and the user's face with 136-dimensional vectors corresponding to the 68 2D keypoints. The keypoints are tracked from videos which contain frontal faces of two humans talking to each other as shown in Figure 1. Each frame is assumed to be independent when training the VAE. In our experiments, we use 20 latent stochastic dimensions, with hidden layer size of 400. We describe the training details in § IV.

After training, the VAE encoder is used to encode facial landmarks in each frame. The frame's posterior distribution is provided by a forward pass on the encoder using the flattened 2D keypoints as input. The mean of this distribution is used as the encoded representation for the corresponding frame. In Figure 3 we show the learned dimensions and their effect on the facial 2D keypoints. Notice how each dimension affects a specific aspect of the face, e.g., face orientation, mouth movement, eye movement, etc. The low dimensional manifold (modeled by stochastic latent variables) has effectively learned to capture the variation across the data points, leading to the disentangled dimensions shown in Figure 3.

Similarly, the trained VAE decoder is used for generation. The prediction model described in § III-C predicts the values for the 20 VAE dimensions for each frame in the next 15-frame segment. We use this predicted representation as a sample from the posterior (as done during training), and perform a forward pass on the decoder to obtain the corresponding 136 dimensional vector that corresponds to the 68 2D keypoints for the predicted frame.

B. *Prediction space*

Instead of regressing to the values of the VAE projection, we cast predictions on a discrete output space. A discrete space allows us to sample the set of possible outcomes during prediction to increase the variance of the agent's facial expressions. In contrast, a regression model would produce relatively static expressions due to the regression-to-mean problem.

In order to discretize the output space, for each VAE latent dimension we cluster temporal segments of length $T = 15$, which is half a second in videos with 30 fps frame rate. Since different expressions occur at different frequencies, we first filter the temporal signals into high and low frequencies using a Butterworth filter with order $N = 3$ and cutoff frequency $Wn = 0.05$. We separately cluster the high and low frequency segments using k-means clustering. We repeat this process for each dimension of the VAE projection, which produces 40 clusters for each of the 20 VAE dimensions for both low and high frequencies, resulting in a total of $40 \times 20 \times 2 = 1600$ clusters. All overlapping segments of length T in the training sequence are used for clustering. We use SciPy library [29] for both, filtering and clustering steps.

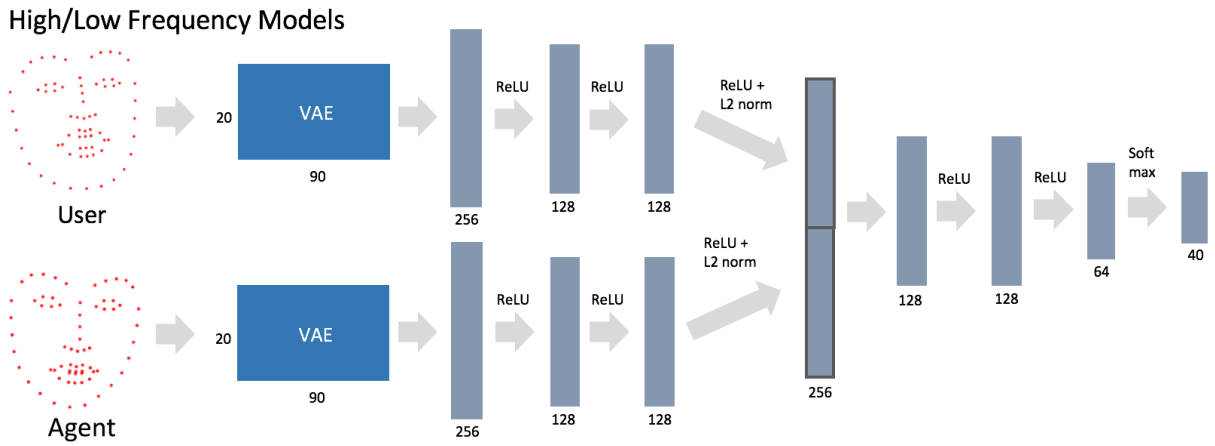

Fig. 4: The model for low-frequency and high-frequency cluster prediction of the agent's facial expression given the user's and the agent's history. Each box corresponds to a layer with its dimensionality written below. The output is a 40-way softmax indicating the probability for each cluster. Models are learned separately for each VAE dimension.

C. Prediction Model

The prediction model generates cluster predictions for the agent every $T = 15$ frames for each VAE dimension. We denote the VAE representation at time t as $\mathbf{a}_t$ and $\mathbf{u}_t$ for the agent and user respectively. The prediction model takes as input the last $l = 90$ frames of the agent, $\mathcal{A} = \{\mathbf{a}_{t-l-1}, \ldots, \mathbf{a}_{t-1}\}$, and the user $\mathcal{U} = \{\mathbf{u}_{t-l-1}, \ldots, \mathbf{u}_{t-1}\}$.

The model consists of two predictors: The low-frequency predictor predicts the low-frequency component of the next 15-frame segment, while the high-frequency predictor predicts the corresponding high-frequency component. Both predictors are conditioned on the user's and agent's previous segments. Introducing separate models for low- and high-frequencies leads to a combinatorial set of possible expressions, thus increasing the expressive power of our model's predictions. We visually verify this claim in § IV by comparing to a model without low- and high-frequency decomposition.

a) Low-frequency cluster predictor: This predictor is a deep neural network that predicts the cluster index of the agent's low-frequency component for each VAE dimension for the next 15 frames. The network takes as input the user's and the agent's previous encoded facial expressions $\mathcal{U}$ and $\mathcal{A}$ respectively. It consists of alternating fully connected layers with a ReLU non-linearity, as shown in Figure 4. The final layer is a 40-way softmax layer, and the low-frequency cluster ID with the highest probability is selected. We use the centroid of the cluster as the predicted value for the next 15-frame segment for this dimension, and repeat this step for all VAE dimensions and combine them to produce the final 15-frame segment for the low-frequency component. To remove discontinuities between consecutive low-frequency segments we perform smoothing near the segment boundaries.

b) High-frequency cluster predictor: The high-frequency predictor (Figure 4) is also modeled as a deep neural network. The input to the high-frequency predictor is exactly the same as for the low-frequency predictor. The procedure for predicting high-frequency outputs is similar to the low-frequency case, with the only difference that the predicted cluster index corresponds to the high-frequency clusters.

Finally, the output signals from the low- and high-frequency predictor are added together and fed through the VAE decoder to produce the facial expression of the agent for the next 15 frames. We repeat this process to generate all the future frames for the agent.

Note that the task of forecasting the agent's expression based on non-verbal cues is a dynamic task. In order to capture time dependencies and correlations we use the agent's and user's past experessions as an input. Time correlations are extremely important if we wish to capture the latent emotional state of the user and the agent as well as the dynamics of the expressional change in the facial landmarks. Computationally, our system is able to extract these cues starting from the lowest layers of the network. Alternatively, a recurrent neural network could also capture time dependencies at some predefined depth of the neural network. However, RNNs are harder to train than feed forward networks and require a lot more data in order to converge. For this reason, we use feed forward neural networks with time-structured inputs.

An important requirement for social robots is to be able to produce responses in real-time. On a single Tesla M40 GPU, given the user's facial keypoints, our pipeline can predict the agent's facial keypoints for the next 15 frames in 0.02 seconds, which satisfies the real-time requirement of 30 frames per second.

IV. EXPERIMENTS

In this section we describe our experimental results. We evaluate our approach using automatically computed quantitative metrics and human studies. We begin by describing our experimental setup, including the dataset and the method used to extract facial keypoints.

A. Dataset

We collected 250 Skype chat videos from Youtube. Each video is a recording of a two-person chat over Skype where both faces are shown side-by-side. (See Figure 1 for an example.) The conversations are on common topics such as personal fitness and well-being, study-abroad experiences, and spirituality. The total number of video frames in the dataset is about 8M. In each video, we treat one person as the user and the other as the agent. We first keep 10% of the dataset as holdout data, and then partitioned the remaining dataset into train, test and validation in the ratio of 90:5:5. The partitions are maintained throughout the experiments.

B. Facial keypoint extraction

The proposed predictor model uses facial keypoints as the input. There has been a number of research in the space of facial keypoint extraction, such as [31], [32], [33], [34], [35]. We use the publicly available OpenFace implementation [36] to obtain 68 2D facial keypoints from each of the two faces from each video frame in our dataset. To encourage further research, we release our keypoint dataset at: `https://yf225.github.io/Learn2Smile/dataset`.

We construct a dataset of $\langle \text{user}, \text{agent} \rangle$ sequence pairs that correspond to two people interacting in the video. When no or only a single person is identified in a specific frame, we remove that frame and split the video into two sub-sequences, so that continuity of interaction is always guaranteed in each sequence. We also only keep sequences that are at least 500 frames long, to remove sequences that cannot provide enough context. In the end, we have about 3,000 video sequences, with a median of 1,088 frames in length for each sequence.

C. Learning Details

For the VAE model, we closely follow the formulation described in [16]. We refer the reader to [17] for the model architecture and the training code. We set the hidden layer size to be 400. At train time, we use ADAM optimization with an initial learning rate of 0.01 and batch size 128. The network is trained until the validation error has converged, which happens after about 800 epochs. Throughout this work, we use Torch [30] to optimize our networks.

Figure 4 shows the network architecture for the high- & low-frequency predictor. Both networks are optimized using backpropagation to predict the ground truth segment cluster ID computed from the other person's facial expressions. The networks are trained in tandem using minibatch ADAM optimization, with an initial learning rate of 0.01 and batch size 128. The network converges after about 100 epochs of training.

D. Baselines

We consider the following baseline models in comparison to our model:

Mirror: This is the simplest baseline where we mirror the facial expressions of the user, delayed by 90 frames.

Avg. Mirror: This baseline returns a moving average of the user's facial expressions for the past 30 frames. In other words, with this baseline the agent responds to the user by imitating the user's average expression. Due to the averaging this baseline tends to have muted expressions.

Random: This baseline randomly chooses a 900-frame sequence from the training set, and then selects one of the faces as the agent. Since the sequence is taken directly from the dataset, its facial expressions would seem realistic if considered in isolation. This baseline helps determine whether it is necessary to consider the user's facial expressions.

NN: Given a user's input, this baseline finds the closest user segment in the training set and returns its corresponding agent segment. For this baseline, we used the approximate nearest neighbor with hierarchical K-means tree implemented in FLANN [37]. We expect this to be a strong baseline because the agent segment will look natural if a good match of the user segment is found in the training set. But, if the best matches for any two nearby segments are found in different video sequences in the training set, there will be discontinuities between the predicted segments.

E. Model ablations

We do a number of ablation studies to understand the (a) importance of predicting segments of different frequencies and (b) role of past agent expressions for predicting the future. To understand these, we trained the following variants of our model:

Ours without frequency split: This model is a variant of our model that does not learn to predict segments of different frequencies. It takes as input the user's past and the agent's past VAE dimensions and directly predicts the cluster ID for the next 15-frame segment of each VAE dimension for the agent. Since our original model has 40 clusters for both low- and high-frequency components, we doubled the number of clusters for the 15-frame segments to 80 for this model, to match the total number of clusters with our original model. The network architecture for this model mimics our original model, except that the last layer is a softmax layer with 80 outputs. However, note that this model will lack the expressiveness obtained by factorizing into low- and high-frequency components ($40 + 40$ vs. 40×40).

Ours without frequency split and agent input: This model also does not learn to predict segments of different frequencies. In addition, it takes as input *only* the user's past VAE dimensions and predicts the cluster ID for the next 15-frame segment of each VAE dimension for the agent. Again, we doubled the number of clusters from 40 to 80. The main differences in the network architecture compared to our original model are (a) it is a single stream model without the concatenation layer in between, and hence the size of the input for the next layer is adjusted accordingly and (b) the last layer is a softmax layer with 80 outputs.

Ours without agent input: This model is for studying the importance of the agent's past facial expressions as input to our model. It uses as input *only* the user's past VAE dimensions and predicts the cluster ID separately for the low- & high-frequency components, for the next 15-frame segment of each VAE dimension for the agent. Therefore,

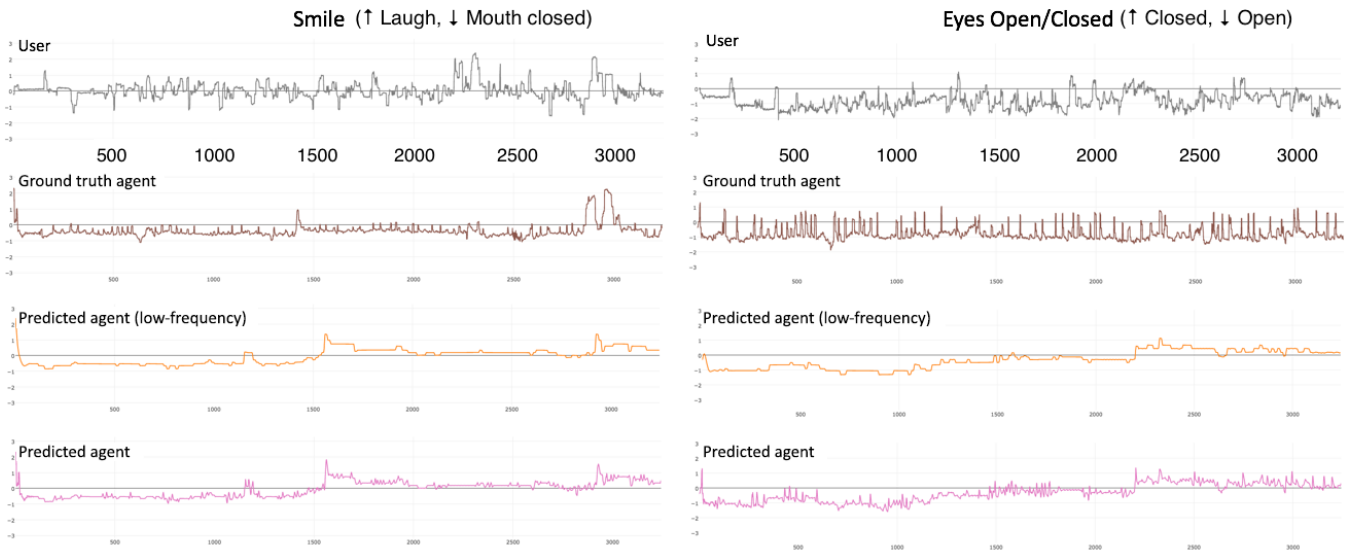

Fig. 5: Plots of user, ground truth agent, predicted agent (low-frequency) and predicted agent values through time for two VAE dimensions corresponding to smiling and eye-blinking. Notice that the predicted agent smiles at approximately the same time as the ground truth, and is able to produce sequence similar to the ground truth for eye blinks.

the architecture for the predictors operate as a single stream network without the concatenation layer in between, and hence the size of the input for the next layer is accordingly adjusted.

F. Quantitative Evaluation

Metrics : We use several measures to quantitatively evaluate our approach.

L2 Error: The simplest measure is the L2 distance between the agent's predicted 2D facial keypoints and the ground truth keypoints. We initialize the 2D keypoints of the agent's first 90 frames from ground truth, and predict 900 frames into the future. We report the average per-frame L2 distance across all frames.

L2 Error with shift: Since the prediction of facial expressions can be inaccurate with respect to the exact timing of their occurrences, we add another measure. Given the agent's predicted keypoints and the ground truth keypoints, for each of the 15-frame segments in the predicted sequence we expand the ground truth segment start point window to be ± 5 frames, and try to find a match within the window that minimizes the L2 distance between the agent's predicted segment and the ground truth segment. We report the average minimum L2 distance across all 15-frame segments in the predicted sequence, divided by 15 to get the per-frame average.

Variation: Our final metric measures whether the agent's predicted and ground truth sequences have similar amount of variation. For each 15-frame segment, we first flatten the 68 2D keypoints of each frame into a 136-dimension vector, then calculate the standard deviation of each dimension across the 15 frames. Next, we sum up the standard deviation value over all 136 dimensions, and then compute the absolute difference of this value between the predicted segment and the ground truth segment. We then average this difference across all segments to obtain the final value.

Model	L2 Error	L2 Error w/ Shift	Variation
Mirror	3.31	0.813	3.08
Avg. mirror	3.25	0.805	2.59
Random	2.47	0.616	2.89
NN	3.40	0.815	3.14
Ours w/o high/low, agent	1.99	0.478	3.63
Ours w/o high/low	**1.59**	0.410	2.33
Ours w/o agent	1.94	0.478	3.45
Ours	**1.60**	**0.406**	**2.09**

TABLE I: Quantitative results for baselines, model ablations and our model. Measures include the L2 error of the predicted 2D keypoints, the L2 error with temporal shifting, and the difference in temporal variation between the predicted keypoints and the ground truth keypoints. See text for details.

Results: The results are shown in Table I. Based on the quantitative measures, the baseline approaches perform poorly. Also, the models that do not split the agent's facial movements into high and low frequencies are only slightly worse than those that do. While the quantitative results show that our original model is only slightly better, the qualitative differences as perceived by human eyes are dramatic as we show in our human studies.

To gain further insight into our model, we show some qualitative results in Figure 5. We show the values of two VAE dimensions corresponding to smiling and eye-blinking. The predicted agent smiles at approximately the same time as the ground truth, and is able to produce sequence similar to the ground truth for eye blinks.

Model Pair	success rate (Ours vs.)	success rate (GT vs.)
Ground truth	52.59%	N/A
Ours	N/A	47.41%
Mirror	47.92%	55.85%
Avg. mirror	79.24%	76.1%
Random	51.15%	54.01%
NN	68.72%	62.27%
Ours w/o high/low	90.21%	87.23%
Ours w/o high/low, agent	69%	74.73%
Ours w/o agent	47.6%	47.94%

TABLE II: Results of human studies performed on the baselines, ablation models, and our model. Each number reports the percentage of A vs. B tests in which A is chosen over B. The first column compares our model to all other models. The second column compares the ground truth to all other models. Note that the human subjects judged the naturalness of the rendered sequences, and didn't appear to have taken into account, or couldn't judge whether the agent was properly interacting with the user.

G. Evaluation using Human Studies

We also ran user studies to gauge the plausibility of the agent's predicted facial expressions as a response to the user. In particular, we conduct pairwise evaluations using Amazon Mechanical Turk. In our study, a HIT consists of two videos. One video shows the user's face and the agent's predicted face through time, while the other shows the same user's face and the agent's face produced by a competing model. The ordering of the videos shown to the judges is chosen randomly to eliminate any bias.

The judges are instructed to watch both videos and then asked to choose the video in which the pair of faces seemed to be more engaged in a conversation. As a hint to assess engagement, they were asked to look at how the agent's facial expressions are reacting to the user's, particularly whether it seemed natural, appropriate and socially typical.

Metrics : Each HIT is evaluated by 9 judges. For model *A* to have a successful prediction, we require that the majority (≥ 5) of judges find the interaction produced by *A* to be more socially engaged than the alternative model *B*. We measure the success rate of model *A* by computing the fraction of test sequences that were considered socially engaged by the above described measure.

Results: Table II shows comparison of success rates of our model against the ground truth, baselines, and all the ablations of our model.

First, two observations from column 2 in the table: Our model has success rate of 52% with respect to the ground truth, showing the efficacy of our model in predicting expressions that are on par with the ground truth. Second, when compared to the ablation 'Ours w/o high/low', our model have success rate of 90%, showing the importance of high-low frequency decomposition. In fact, qualitative comparisons show the acute visual differences in the sequences generated by our model and the ablation models, which can be seen in the accompanying video or at `https://yf225.github.io/Learn2Smile/video`.

A closer look at the table also shows that our model is on par with two baselines: 'Mirror', that perfectly mirrors the user, and "Random" that picks a random sequence from the training set. This observation led us to hypothesize that judges are answering an un-asked easier question of *whether the agent's expressions look realistic*, and not the harder asked question of *whether the agent's expressions look responsive to the user's*. This hypothesis can also explain why our model would have a high success rate of 90% against ablation 'Ours w/o high/low' that is slow-moving as seen from the accompanying video, while having lower success rate against baselines such as 'NN' that produces reasonable though choppy facial movements.

To further validate our hypothesis, we did another set of user studies. In this study, we used the exact same HIT setup, but replaced our model with the ground truth agent. If the judges are paying attention to the agent's engagement in the conversation, we would expect ground truth to have close to 100% success rate against the baseline 'Mirror', since the latter is not engaging in the conversation in a meaningful way. From the results in the last column of Table II, this is clearly not the case. In fact, the success rates of ground truth against all the competing models closely follow the success rate of our model against them. Hence, we can safely conclude that the judges didn't pay attention to whether the agent looks engaged in the conversation, and instead they focused on whether the agent's facial expressions look natural and consistent.

Thus, the user studies favor the models that produce natural facial expressions. When combined with the quantitative results in § IV-F, our model is the only one that is capable of making predictions that are quantitatively close to the ground truth movements in response to the user, while also being judged qualitatively realistic by humans.

V. DISCUSSION

In this paper, we focus on the non-verbal interaction between an agent and the user. We present a novel deep neural network model that automatically learns to generate the agent's facial expressions in response to the user. The only input to the model is the facial 2D keypoints tracked over time from the two people in the conversation, where one person is treated as the user, and the other as the agent.

Our method outperforms all other baselines, which we show through quantitative analysis and user studies. More importantly, our results highlight the importance of predicting at different frequencies, so that expressions are captured at their naturally appropriate temporal scales.

There are a number of natural next steps: First, deploy the model in an interactive environment where the user and the agent can be actively engaged: this would enable the user to also actively adjust their expressions in response to the agent, and thus a true interaction emerges.

The expression of non-verbal cues in an interaction is often dependent on more than the other person's expression. It may

also be dependent on the words that may be spoken, or the changing mental state of the person expressing the cues. For future work, we plan to explore these other factors that may lead to non-verbal expressions.

The expression of non-verbal cues also varies from person to person, and even varies culturally. We assume non-verbal cues are generated using a single model. However, it may be necessary to have latent variables that encode the type of response the agent prefers to produce, from a more subdued response to a very expressive response.

REFERENCES

[1] R. Gockley, A. Bruce, J. Forlizzi, M. Michalowski, A. Mundell, S. Rosenthal, B. Sellner, R. Simmons, K. Snipes, A. C. Schultz, et al. Designing robots for long-term social interaction. In IEEE International Conference on Intelligent Robots and Systems. IEEE, 2005.

[2] H. Kozima, M. P. Michalowski, and C. Nakagawa. Keepon: A playful robot for research, therapy, and entertainment. International Journal of Social Robotics, 2009.

[3] K. WJ and O. J. The representation of agents: anthropomorphism, agency, and intelligence. Conference on human factors in computing systems: common ground, 1996.

[4] R. Dillmann, O. Rogalla, M. Ehrenmann, R. Zollner, and M. Bordegoni. Learning robot behaviour and skills based on human demonstration and advice: the machine learning paradigm. In ROBOTICS RESEARCH-INTERNATIONAL SYMPOSIUM, volume 9, pages 229238, 2000.

[5] S. B. Kang and K. Ikeuchi. Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps. IEEE transactions on robotics and automation, 13(1):8195, 1997.

[6] M. Turk. Multimodal interaction: A review. Pattern Recognition Letters, 36:189195, 2014.

[7] C. D. Frith and U. Frith. Social cognition in humans. Current Biology, 17(16):R724R732, 2007.

[8] J. N. Saby, A. N. Meltzoff, and P. J. Marshall. Infants somatotopic neural responses to seeing human actions: I've got you under my skin. PLoS ONE, 8(10), 10 2013.

[9] A. N. Meltzoff and M. K. Moore. Explaining facial imitation: A theoretical model. Early development & parenting, 6(3-4):179, 1997.

[10] D. Bohus and E. Horvitz. Facilitating multiparty dialog with gaze, gesture, and speech. In International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction, page 5. ACM, 2010.

[11] C. Chao, J. Lee, M. Begum, and A. L. Thomaz. Simon plays simon says: The timing of turn-taking in an imitation game. In 2011 RO-MAN, pages 235240. IEEE, 2011

[12] V. Bettadapura. Face expression recognition and analysis: The state of the art. CoRR, abs/1203.6722, 2012.

[13] C. Chibelushi and F. Bourel. Facial expression recognition: A brief tutorial overview. CVonline: On-Line Compendium of Computer Vision, 9, 2003.

[14] G. Stratou, S. Scherer, J. Gratch, and L.-P. Morency. Automatic nonverbal behavior indicators of depression and ptsd: the effect of gender. Journal on Multimodal User Interfaces, 9(1):1729, 2015.

[15] T. Pfister, X. Li, G. Zhao, and M. Pietikainen. Recognising spontaneous facial micro-expressions. In 2011 International Conference on Computer Vision, pages 14491456. IEEE, 2011.

[16] D. P. Kingma and M. Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

[17] J. Van Amersfoort. VAE-Torch. https://github.com/y0ast/VAE-Torch, 2014.

[18] S. Levine, C. Theobalt, and V. Koltun. Real-time prosody-driven synthesis of body language. In ACM Transactions on Graphics (TOG), volume 28, page 172. ACM, 2009.

[19] P. Ekman and W. V. Friesen. Facial Action Coding System. Consulting Psychologists Press Inc, 1978.

[20] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski. Classifying facial actions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21, 1999.

[21] I. Essa and A. Pentland. Coding, analysis, interpretation, and recognition of facial expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19, 1997.

[22] C. Breazeal. Function meets style: insights from emotion theory applied to HRI. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 34(2):187194, 2004.

[23] T. Fukuda, M.-J. Jung, M. Nakashima, F. Arai, and Y. Hasegawa. Facial expressive robotic head system for human-robot communication and its application in home environment. Proceedings of the IEEE, 92(11), 2004.

[24] G. Gordon, S. Spaulding, J. K. Westlund, J. J. Lee, L. Plummer, M. Martinez, M. Das, and C. Breazeal. Affective personalization of a social robot tutor for childrens second language skills. 2016.

[25] R. Kirby, J. Forlizzi, and R. Simmons. Affective social robots. Robotics and Autonomous Systems, 58(3), 2010.

[26] T. W. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots: Concepts, design, and applications. Technical Report CMU-RI-TR-02-29, Robotics Institute, Pittsburgh, PA, 2002.

[27] I. Leite, C. Martinho, and A. Paiva. Social robots for long-term interaction: A survey. International Journal of Social Robotics, 5, 2013.

[28] B. Woolf, W. Burleson, I. Arroyo, T. Dragon, D. Cooper, and R. Picard. Affect aware tutors; recognising and responding to student affect. International Journal of Learning Technology, 4, 2009.

[29] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open Source Scientific Tools for Python, 2001-, http://www.scipy.org/.

[30] R. Collobert and K. Kavukcuoglu and C. Farabet. Torch7: A Matlab-like Environment for Machine Learning. BigLearn, NIPS Workshop, 2011.

[31] B. Czuprynski and A. Strupczewski. High accuracy head pose tracking survey. In International Conference on Active Media Technology, pages 407420. Springer, 2014.

[32] F. De la Torre and J. F. Cohn. Facial expression analysis. In Visual analysis of humans, pages 377409. Springer, 2011.

[33] E. Murphy-Chutorian and M. M. Trivedi. Head pose estimation in computer vision: A survey. IEEE transactions on pattern analysis and machine intelligence, 31(4):607626, 2009.

[34] E. Sariyanidi, H. Gunes, and A. Cavallaro. Automatic analysis of facial affect: A survey of registration, representation, and recognition. IEEE transactions on pattern analysis and machine intelligence, 37(6):11131133, 2015.

[35] N. Wang, X. Gao, D. Tao, and X. Li. Facial feature point detection: A comprehensive survey. arXiv preprint arXiv:1410.1037, 2014.

[36] T. Baltrusaitis, P. Robinson, and L.-P. Morency. Openface: an open source facial behavior analysis toolkit. In IEEE Winter Conference on Applications of Computer Vision, 2016.

[37] M. Muja and D. G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 36, 2014.