Received: 12 May 2022 | Accepted: 15 May 2022

DOI: 10.1002/cav.2076

SPECIAL ISSUE PAPER

WILEY

Disentangling audio content and emotion with adaptive instance normalization for expressive facial animation synthesis

Che-Jui Chang | **Long Zhao** | **Sen Zhang** | **Mubbasir Kapadia**

Department of Computer Science, Rutgers University, Piscataway, New Jersey, USA

Correspondence
Che-Jui Chang, Department of Computer Science, Rutgers University, Piscataway, NJ 08854, USA.
Email: chejui.chang@rutgers.edu

Funding information
National Science Foundation, Grant/Award Numbers: IIS-1703883, IIS-1955404, IIS-1955365, RETTL-21192; U.S. Department of Homeland Security, Grant/Award Number: 22STESE00001 01 01

Abstract

3D facial animation synthesis from audio has been a focus in recent years. However, most existing literature works are designed to map audio and visual content, providing limited knowledge regarding the relationship between emotion in audio and expressive facial animation. This work generates audio-matching facial animations with the specified emotion label. In such a task, we argue that separating the content from audio is indispensable—the proposed model must learn to generate facial content from audio content while expressions from the specified emotion. We achieve it by an adaptive instance normalization module that isolates the content in the audio and combines the emotion embedding from the specified label. The joint content-emotion embedding is then used to generate 3D facial vertices and texture maps. We compare our method with state-of-the-art baselines, including the facial segmentation-based and voice conversion-based disentanglement approaches. We also conduct a user study to evaluate the performance of emotion conditioning. The results indicate that our proposed method outperforms the baselines in animation quality and expression categorization accuracy.

KEYWORDS

adaptive instance normalization, audio-driven animation, content-emotion disentanglement, emotion-conditioning, expressive facial animation synthesis

1 | INTRODUCTION

The movement and expression of a human face are tightly interlinked with the phonetic content and underlying emotional intent. Recent studies[1-3] have attempted to generate facial animations from speech. Those speech-driven methods leverage the correlation between the speech and the talking face and build end-to-end models for the mapping. However, their assumption that the acoustic emotion can infer the emotional face movement is invalid because the correlation of *emotion* in audio and faces could differ from one person to another or from context to context. As a result, a model relying entirely on audio input to infer the facial movements could only learn the mapping between audio and visual contents.

In this work, we formulate the problem of expressive facial animation synthesis as a trainable model capable of generating the desired facial animations given an audio and an emotion label. We separate audio and emotion as two input

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

modalities, enabling the users to directly control the expression of the facial animation with emotion specification. We argue that the disentanglement of the content and emotion from audio is indispensable. A model designed for such a task should be able to learn the content information from the audio input and the expressive facial dynamics from the emotion input. Suppose the emotion from the audio input and the conditional input are both passed to the model. In that case, the model is likely to ignore the conditional input and primarily accept the emotion information from the audio modality. In this case, the model fails to perform "*emotion conditioning*."

We approach the expressive 3D facial animation synthesis with a learnable model that takes audio and emotion as inputs, disentangles the audio content and emotion, entangles the content with the specified emotion, and outputs the vertex positions and texture maps. To the best of our knowledge, we are the first to generate facial vertices with textures for expressive facial animation synthesis. We further achieve the emotion-conditioned synthesis by adopting adaptive instance normalization (AdaIN)[4] in our architecture. Our model effectively learns the joint embedding of audio content and the specified emotion, laying the foundation for controllable stylistic facial animation generation from an interpretable latent space.

We further conduct a user study to evaluate the performance of content-emotion disentanglement and emotion conditioning. We compare our proposed method with the experiment's facial segmentation-based and voice conversion-based disentanglement approaches. The result indicates our model outperforms the baselines.

2 | RELATED WORKS

2.1 | End-to-end speech-driven facial animation

Over the past few years, deep learning approaches have become dominant for end-to-end facial animation synthesis..[1,2,6-8] Karras et al.[2] build a convolution-based model that outputs fine-grained facial meshes with head poses and emotions. Pham et al.[6] apply facial action unit detection on a wild video dataset based on FACS and learns to output the facial AU parameters. Recently, Zhou et al.[7] were proposed to disentangle content and speaker channels for speech-driven animation, but the facial expression is still determined by the input speech and the reference image. Our work is motivated by the aforementioned papers for end-to-end audio-visual modeling but different in how the model has to learn the content-emotion disentanglement jointly.

2.2 | Controllable talking face generation

Controllable talking face synthesis aims to generate faces that match the audio and a reference. Zhou et al.[9] are proposed to effectively control the pose of the talking face from a referenced pose video. The reference could also be emotion. For example, Wang et al.[5] accept an audio and an emotion label as inputs. It then separates the entire face into audio-related and emotion-related regions. The two regional faces are then combined and passed to the refinement network for photo-realistic image generation. Vougioukas et al.[10] take as reference the speaker's identity. It attempts to encode the identity into their model and applies recurrent neural networks and GAN,[11] respectively, for temporal modeling and facial animation synthesis.

Usually, controllable emotional face generation requires the disentanglement of audio content and emotion. For instance, Ji et al.[12] formulate the disentanglement of audio content and emotion as a separate problem from animation synthesis. It first aligns audio clips by dynamic time warping and then learns a cross-reconstruction model. Compared with our method, it requires additional training and a parallel dataset to accomplish.

2.3 | Face generation with 3D morphable models

In the field of computer graphics, 3D morphable models (3DMMs)[13-15] are widely used in facial representations because 3DMM parameters have the advantage of being easily manipulated by humans or generated by models to create a new plausible face. Several researches[13,16-18] have been conducted using 3DMM as a prior for facial reconstruction from images or video. Recent studies[19-21] also aimed at reconstructing fine-scale facial details from monocular videos. These methods learn to synthesize several textures along with the 3DMM parameters and map the textures to the face meshes for

optimization. The reconstructed morphable model parameters can also become the learning target of neural network models for facial synthesis. For example, the authors in References 22-24 approach audio-driven video portraits by reconstructing 3DMM parameters from a talking face video and training an audio-to-face network to generate the parameters. The coarse-scale 3D faces are passed to a neural rendering network to generate photo-realistic talking faces. Our method is motivated by the related papers[11,18,24] that leverage the reconstructed 3DMM parameters as the learning objective. We further extend coarse-scale reconstruction to fine-scale to preserve the expression details for expressive facial synthesis.

3 | METHODOLOGY

To generate expressive facial animations with the specified emotions, we build an encoder-decoder-based model that takes as inputs the speech and a one-hot emotion vector and outputs a sequence of facial vertex positions and texture maps. The model architecture is illustrated in Figure 1. Our model first encodes the sequential audio input with a bidirectional LSTM Audio Encoder. The output of the encoder is projected and then passed to the AdaIN module. AdaIN[4,25] is applied in our architecture to remove the emotion information in the audio features and entangle the remaining content features with the specified emotion embedding. As we obtain the joint representation from the two separate modalities (audio encoder and emotion embedder), we pass the feature to the bidirectional LSTM face decoder. The bidirectional setting of both the encoder and decoder helps the model capture the visual coarticulation effect and temporal dependency. The decoder output is then projected to the latent space of our *joint content-emotion embedding*. Finally, the embedding is

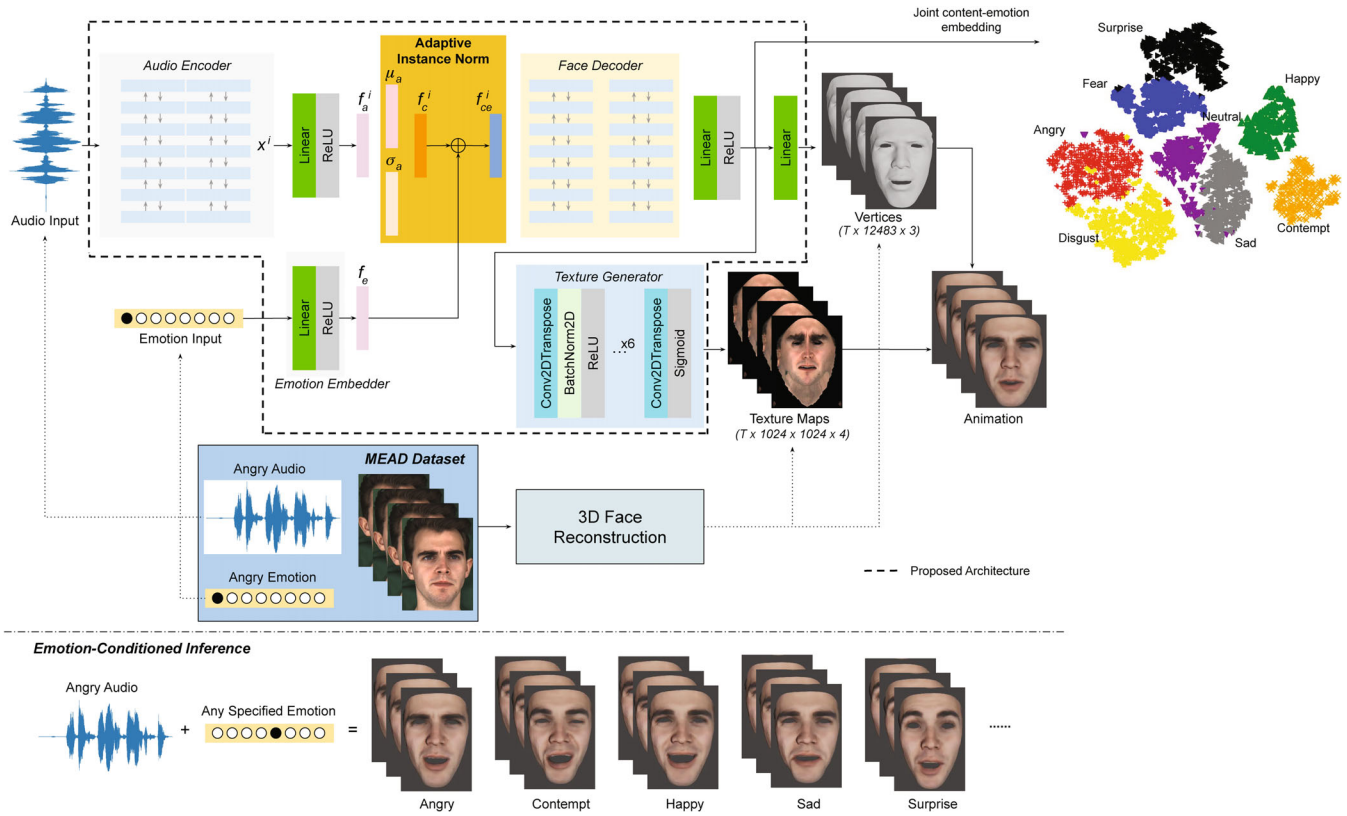

FIGURE 1 Our proposed pipeline and model architecture. We first reconstruct facial vertices and texture maps from MEAD dataset[5] as the training targets. Audio and its corresponding emotion label are used as inputs during training. The adaptive instance norm (AdaIN) module removes the emotion feature from audio and entangles the content feature with the specified emotion for vertex and texture generation. For emotion-conditioned inference (down), the model accepts audio and any specified emotion as inputs and outputs the expressive facial animation that matches the audio content and the specified emotion. The tSNE visualization of the learned content-emotion embedding is shown in the upper-right corner.

used to generate the positions of all facial vertices and used as the input to the deep convolutional texture generator[26] for the albedo and displacement maps.

3.1 | Emotion-conditioned synthesis

Disentangling audio content and emotion in the model is necessary for achieving emotion conditioning. If the audio emotion is not removed, the audio emotion and the specified emotion, which are different during the inference phase, will both cause changes to the output facial expressions. To prevent the emotion feature in the audio from leaking to the decoder, we apply AdaIN[4] to eliminate the time-invariant information from the audio feature and then combine the time-varying content feature with the specified emotion embedding. The same technique is also used in voice conversion that aims to remove speaker information from audio features and then combine the content with the speaker feature from another modality for the decoder. To formulate the normalization process, let's consider $X = \{x^i\}_{i=1,\dots,T}$ as the input speech features and h_e as the one-hot emotion condition. The vertex and texture at the ith time step are denoted as y^i_{vert} and y^i_{tex}. The audio encoder and the emotion embedder process X and h_e as follows:

$$F_a = \text{ReLU}(W_a \cdot E_a(X) + b_a), \tag{1}$$

$$f_e = \text{ReLU}(W_e \cdot h_e + b_e). \tag{2}$$

E_a is the bidirectional LSTM audio encoder. F_a is the audio embedding with time-varying content and time-invariant emotion information. We denote $F_a = \{f^i_a\}_{i=1,\dots,T}$ because it is a sequential vector. f_e is the emotion embedding. AdaIN removes emotion information from f^i_a and entangles the resulting content feature, f^i_c, with f_e to generate the joint representation of the audio content and specified emotion, f^i_{ce}, as formulated in the following equations:

$$f^i_c = \frac{f^i_a - \frac{1}{T}\sum_j f^j_a}{\sigma(F_a)}, \tag{3}$$

$$f^i_{ce} = f^i_c + f_e. \tag{4}$$

Note that the mean and standard deviation are calculated along the time axis in Equation (3) and f_e is added to the content feature at every timestep in Equation (4). Unlike the vanilla instance normalization that has the learnable scale and shift parameters, Equation (3) only normalizes the audio feature to zero mean and unit variance. In practice, we find this more effective in our experiments.

3.2 | Optimization

As our model generates two outputs, vertices $\hat{y}^i_{\text{vert}}$ and texture maps $\hat{y}^i_{\text{tex}}$, we define two loss terms, $\mathcal{L}_{\text{vert}}$ and $\mathcal{L}_{\text{tex}}$, respectively, for the optimization. The two losses are simply the mean square errors between the ground truth and the prediction, as shown in Equations (5) and (6). However, we find that if only the vertex error is used in the loss function, the model tends to predict over-smoothed vertex motions. Therefore, we follow[2] to add the vertex velocity constraint in the loss function. Adding vertex velocity error encourages the model to learn the vertex motions and to be more responsive to audio contents. The velocity constraint, $\mathcal{L}_{\text{vel}}$, is defined in Equation (7), where the vertex velocity is the difference between neighboring vertex positions.

$$\mathcal{L}_{\text{vert}} = \frac{1}{T}\sum_{i=1}\left\|y^i_{\text{vert}} - \hat{y}^i_{\text{vert}}\right\|^2_2, \tag{5}$$

$$\mathcal{L}_{\text{tex}} = \frac{1}{T}\sum_{i=1}\left\|y^i_{\text{tex}} - \hat{y}^i_{\text{tex}}\right\|^2_2, \tag{6}$$

$$\mathcal{L}_{\text{vel}} = \frac{1}{T-1}\cdot\sum_{i=2}\left\|(y^i_{\text{vert}} - y^{i-1}_{\text{vert}}) - (\hat{y}^i_{\text{vert}} - \hat{y}^{i-1}_{\text{vert}})\right\|^2_2. \tag{7}$$

In practice, we observe that the velocity constraint also speeds up the training process. Finally, the loss of the entire model is defined as a combination of the three loss components:

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{\text{vert}} + \lambda_2 \cdot \mathcal{L}_{\text{vel}} + \lambda_3 \cdot \mathcal{L}_{\text{tex}}. \tag{8}$$

In our experiment, we set λ_1 as 1, while λ_2 and λ_3 as 10 and 10,000, respectively.

4 | EXPERIMENTS

4.1 | Dataset and preprocessing

We leverage Facescape[27] for fine-scale facial reconstruction to obtain expressive and emotional talking face data. We then reconstruct the faces from MEAD dataset.[5] It is a parallel audio-visual talking face dataset, captured from multiple viewpoints with eight emotions (angry, contempt, disgust, fear, happy, neutral, sad, surprise) and three intensity levels (weak, medium, and strong). We select the subject, M003, and reconstruct the 3D faces with all emotions and intensities from the frontal view. For audio processing, we carefully tune the window and step sizes so that the resulting MFCC features are the same fps as the video. The input audio features are 80-dimensional. The dimension of the output vertex is $12{,}483 \times 3$, and the size of the texture is 1024×1024. We use 256 for all hidden units for the model architecture and 2 layers for both the encoder and decoder. We use the same parameters as DCGAN[26] for the texture generator. We use 80% of the sentences for training and the other 20% as validation data. The train-validation split is the same as Reference 5 to make fair comparisons.

4.2 | Baselines

We compare our model with three baselines.

AudioE2E[2] takes as input a window of audio features and a learnable emotion embedding and outputs the vertex dynamics. Once the model is trained, we take the average of the embeddings in an emotion category and use it as the conditional input for emotion-conditioned synthesis. Note that no texture or albedo is generated in this model.

The MEAD baseline disentangles the audio content and emotion by segmenting faces into content-related and emotion-related regions. Since the baseline is originally designed for emotion-conditioned talking face generation, we further apply the same reconstruction method on top of the output videos. In this article, we call it **MEAD***.

The voice conversion-based disentanglement (**VCBD**) is our third baseline. We use AutoVC[28,29] to extract the content features from the input audio. Note that the VC-trained audio content extractor was originally proposed in Reference 7 for talking face synthesis. Here we apply the same concept and train the 3D facial animation synthesis with the same model architecture without AdaIN.

4.3 | Metrics

We will use vertex error, ϵ_v, and texture error, ϵ_t, to evaluate the model performance on our validation set. The two errors, defined in Equations (5) and (6), only reflect how well the model can generate the facial animation when the audio emotion and the conditional emotion are of the same category. However, we are more interested in the result of disentanglement. In other words, when the conditional emotion is different from the audio emotion, the model should be able to ignore the audio emotion and output the animation that matches the specified emotion. We further conduct a user study for the evaluation of emotion conditioning.

User study. We recruit 317 participants. Most of them are undergraduates with no animation or character design background. The survey was conducted anonymously and adhered to university IRB requirements. The survey was created via Qualtrics,[30] and the link was distributed to the participants by university email.

The survey has two sections, expression categorization and preference for animation quality. For the first section, each participant was first presented with eight reconstructed face images representing all the emotions. Then they were given an animation generated by one of the proposed methods and asked to select one of the eight emotions that

matched the expression of the given animation. Sixty animations were presented in the first section. For the second section, the participants were first given an instruction asking them to select their preference of animation quality. A clear message was displayed to clarify that the overall animation quality should account for the quality of the facial texture and whether or not the animation matches the audio content, regardless of the emotion or facial expression. Then the participants were presented with two animations, generated from the same inputs but by two different methods, one from ours and the other from a baseline. The selection of the animations, methods, and the presentation sequence was randomized. Followed by the presentation of two videos, the question (Overall, which animation has better quality?) and five options were displayed. The five choices are: definitely the first, probably the first, no preference, probably the second, and definitely the second. A total of 40 questions were presented to each participant in this section.

5 | RESULTS

First, we present the static expressions generated by our model in Figure 2. Four audio clips with different emotions are used as input to our model. We then specify all eight expressions to guide the output animation. The resulting 32 faces are shown in Figure 2. We can see the emotion encoded in the input audio is eliminated so that the faces in the same column share the same specified expression. It is also important that the specified emotion changes the entire facial vertices and textures. For example, a happy face would usually have a wide-open mouth, and a contemptuous one would have a skewed mouth. This is expected because we allow the conditional emotion to be encoded in the latent representation in our model and cause changes to the final vertices. The setting is different from **MEAD*** that only allows the specified emotion to change the upper face. For animation results, please refer to our supplementary video.

We also show the result of expression categorization in the user study. Our confusion matrix is shown in Figure 3. We can tell that users are more confused when the specified emotion is anger, contempt, or disgust. Fear is falsely classified because it is more difficult to model. When the specification is happy, users predict with the highest accuracy. The neutral animations, in some cases, are confused with anger, contempt, or disgust. We also observe that the sad expression is easily confused with disgust, and surprise is easily confused with a happy face.

From the t-SNE[31] visualization of the joint content-emotion embedding in Figure 1 (upper right), we can see the embeddings are clustered according to the specified emotions. We see in the figure that some clusters are closer to others. For example, the angry cluster is close to disgust, while fear is close to surprise. The observation is in partial accordance with the categorization result, where disgust is confused with anger, but fear is treated as neutral. Interestingly, the proximity between each emotion is also preserved in the joint representation. We believe this is because the geometrical differences in facial expressions do not necessarily agree with the perceptual disparities.

FIGURE 2 Rendered faces with different expressions with our proposed method. The "emotion conditioning" is achieved because the generated facial expression does not correspond to the audio emotion but the specified emotion.

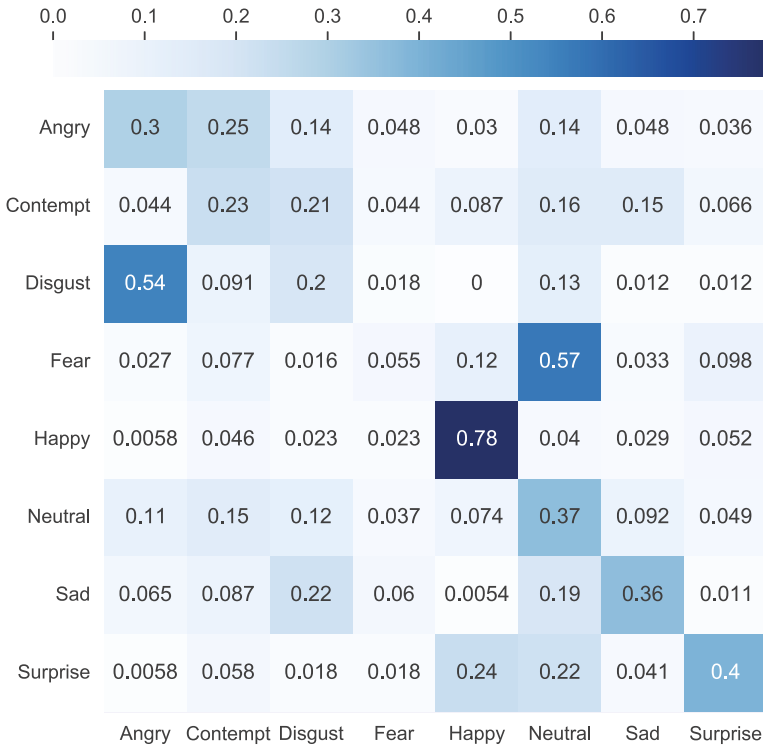

FIGURE 3 Confusion matrix of expression categorization. The *Y*-axis is the true label, and the *X*-axis is the prediction. Statistically, we collected 163–184 answers for each of the eight emotion categories.

5.1 | Comparisons

We show the quantitative comparative results with the three baselines in Table 1 and Figure 5. Our method outperforms **AudioE2E** in vertex error because their convolution-based model does not take into consideration the neighboring frames. On the contrary, our architecture successfully models the temporal relationship between frames and captures the coarticulation effect. In terms of expression classification, our model has even better accuracy. The facial animation generated by **AudioE2E** is textureless, so it is hard for participants to distinguish faces with different expressions. In terms of animation quality, 57% think ours is better while only 24% vote for **AudioE2E**.

When compared with **MEAD***, our proposed method is better in terms of vertex error, but worse in texture error. We find that the vertex motion generated by **MEAD*** is quite unstable—the instability results from the limited generalization ability of the MEAD approach. However, since MEAD can generate photorealistic face videos, the reconstructed faces of **MEAD*** have much better quality (5.62 in ϵ_t). We find that the learning objective of texture maps would encourage our model to output smooth texture maps, so the texture quality reduces. In terms of expression classification and animation quality, our model outperforms **MEAD***, as shown in Table 1 and Figure 5. Also, the dynamic results can be found in

TABLE 1 Comparison with other baselines

	ϵ_v	$\epsilon_t \cdot 10^4$	Accuracy
AudioE2E	2.64 (0.95)	N/A	0.261
MEAD*	3.94 (1.83)	**5.62** (1.30)	0.270
VCBD	2.97 (1.20)	16.71 (7.71)	0.222
Ours W/O AdaIN	2.53 (1.06)	16.97 (7.49)	0.307
Ours	**2.44** (1.02)	16.02 (7.42)	**0.335**

Note: The mean and standard deviation of the two errors are presented. Bold-face values mean the best performance.

Figure 4. The instability of texture maps synthesized by **MEAD***, including the rapid change in albedo and wrinkles in neighboring frames, is the main cause of quality and accuracy decrease.

When compared with the voice conversion-based disentanglement method, our AdaIN approach prevails in all metrics. The **VCBD** encoder trained on voice conversion tasks is designed initially to separate voice content and speaker information in the audio, not the content and emotion. When we transfer the content extractor to speech-driven facial synthesis, the extracted content feature could probably contain no speaker information but certain emotion information. The content feature could also be insufficient for the facial synthesis model to learn and infer the animation. We can see the artifacts of **VCBD** in Figure 4. When the happy audio is passed to the model, and a different emotion is specified, **VCBD** still generates happy expressions and fails to achieve emotion conditioning. The accuracy of expression classification in Table 1 also reflects that our AdaIN is better in emotion conditioning.

Ablation study. It is interesting to know to what extent the AdaIN module improves the experimental results. We see in Table 1 and Figure 5 that both the vertex and texture errors increase if AdaIN is removed from our model. In terms of expression classification, the one with AdaIN obtains higher accuracy. As for the animation quality, 39% vote for AdaIN while only 28% vote for no AdaIN. It is no surprise because, in the AdaIN module, we suppress the time-invariant emotion feature and entangle the content features with the specified emotion embedding. Instead, without AdaIN, we are allowing two emotion modalities to cause changes to the output, which confuses the model.

FIGURE 4 Dynamic results generated by two baselines (**MEAD*** and **VCBD**) and our method. Happy audio and the two specified emotions, angry and sad, are passed to each model as inputs. We show the artifacts in red boxes.

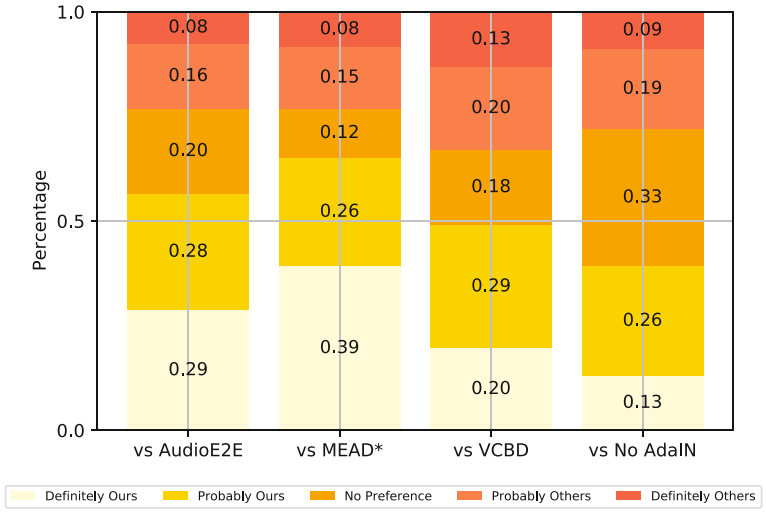

FIGURE 5 The preference for animation quality between all comparison pairs. We perform the comparisons of ours versus other methods. Statistically, 508, 502, 551, and 530 answers were gathered, respectively, for the four comparisons.

5.2 | Blend between expressions and intensity level control

We present the demonstration in the supplementary video that our model takes as conditional input an interpolated or extrapolated vector between two different emotions and blends faces to generate unseen expressions in the dataset. Additionally, we train our model with eight different emotions and three intensity levels as the conditions. During inference, an emotion and a corresponding intensity level are specified. We present the result to show that the idea of AdaIN can be applied to not only emotion conditioning but also intensity level control. It is because the time-invariant features in audio are eliminated and then appended with the specified emotion and intensity embeddings.

6 | CONCLUSION

In this work, we focus on synthesizing the expressive facial animation from speech and the desired emotion. We generate facial vertices and textures by a bidirectional encoder-decoder-based model and a convolution-based texture generator. Most importantly, our AdaIN successfully disentangles content and emotion features from audio and learns the joint content-emotion embedding. We conduct a user study to evaluate the emotion conditioning and animation quality. The result indicates that our model is more effective than the three competitive baselines in generating emotion-matching expressions and high-quality animations.

ACKNOWLEDGMENTS

The research was supported in part by NSF awards: IIS-1703883, IIS-1955404, IIS-1955365, RETTL-2119265, and EAGER-2122119. This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 22STESE00001 01 01.

ORCID

Che-Jui Chang https://orcid.org/0000-0001-7935-8723
Long Zhao https://orcid.org/0000-0001-8921-8564

REFERENCES

1. Taylor S, Kim T, Yue Y, Mahler M, Krahe J, Rodriguez AG, et al. A deep learning approach for generalized speech animation. ACM Trans Graph. 2017;36(4):1–11.
2. Karras T, Aila T, Laine S, Herva A, Lehtinen J. Audio-driven facial animation by joint end-to-end learning of pose and emotion. ACM Trans Graph. 2017;36(4):1–12.
3. Cudeiro D, Bolkart T, Laidlaw C, Ranjan A, Black MJ. Capture, learning, and synthesis of 3D speaking styles. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2019. p. 10101–11.
4. Huang X, Belongie S. Arbitrary style transfer in real-time with adaptive instance normalization. Proceedings of the IEEE International Conference on Computer Vision; 2017. p. 1501–10.
5. Wang K, Qianyi W, Song L, Yang Z, Wayne W, Qian C, et alMead: a large-scale audio-visual dataset for emotional talking-face generation. ECCV; 2020.
6. Pham HX, Cheung S, Pavlovic V. Speech-driven 3D facial animation with implicit emotional awareness: a deep learning approach. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops; July 2017.
7. Zhou Y, Han X, Shechtman E, Echevarria J, Kalogerakis E, Li D. Makelttalk: speaker-aware talking-head animation. ACM Trans Graph (TOG). 2020;39(6):1–15.
8. Guo Y, Chen K, Liang S, Liu YJ, Bao H, Zhang J. Ad-nerf: audio driven neural radiance fields for talking head synthesis. Proceedings of the IEEE/CVF International Conference on Computer Vision; 2021. p. 5784–94.
9. Zhou H, Sun Y, Wu W, Loy CC, Wang X, Liu Z. Pose-controllable talking face generation by implicitly modularized audio-visual representation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021. p. 4176–86.
10. Vougioukas K, Petridis S, Pantic M. Realistic speech-driven facial animation with gans; 2019.
11. Pumarola A, Agudo A, Martinez AM, Sanfeliu A, Moreno-Noguer F. Ganimation: anatomically-aware facial animation from a single image. Proceedings of the European conference on computer vision (ECCV), pages 818–833, 2018.
12. Ji X, Zhou H, Wang K, Wayne W, Loy CC, Cao X, Xu F. Audio-driven emotional video portraits. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021. p. 14080–9.
13. Zollhöfer M, Thies J, Garrido P, Bradley D, Beeler T, Pérez P, et al. State of the art on monocular 3D face reconstruction, tracking, and applications. Comput Graph Forum. 2018;37:523–50.
14. Blanz V, Vetter T. A morphable model for the synthesis of 3D faces. Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques; 1999. p. 187–94.

15. Paysan P, Knothe R, Amberg B, Romdhani S, Vetter T. A 3D face model for pose and illumination invariant face recognition. Proceedings of the 2009 6th IEEE International Conference on Advanced Video and Signal Based Surveillance. IEEE; 2009. p. 296–301.
16. Egger B, Smith WAP, Tewari A, Wuhrer S, Zollhoefer M, Beeler T, et al. 3D morphable face models—Past, present, and future. ACM Trans Graph (TOG). 2020;39(5):1–38.
17. Guo J, Zhu X, Yang Y, Yang F, Lei Z, Li SZ. Towards fast, accurate and stable 3D dense face alignment. Proceedings of the European Conference on Computer Vision (ECCV); 2020.
18. Zhao L, Peng X, Tian Y, Kapadia M, Metaxas DN. Towards image-to-video translation: a structure-aware approach via multi-stage generative adversarial networks. Int J Comput Vis. 2020;128(10):2514–33.
19. Chen A, Chen Z, Zhang G, Mitchell K, Jingyi Y Photo-realistic facial details synthesis from single image. Proceedings of the IEEE/CVF International Conference on Computer Vision; 2019. p. 9429–39.
20. Athar SR, Pumarola A, Moreno-Noguer F, Samaras D. Facedet3d: facial expressions with 3D geometric detail prediction. arXiv preprint arXiv:2012.07999; 2020.
21. Feng Y, Feng H, Black MJ, Bolkart T. Learning an animatable detailed 3D face model from in-the-wild images. arXiv preprint arXiv:2012.04012; 2020.
22. Wen X, Wang M, Richardt C, Chen Z-Y, Shi-Min H. Photorealistic audio-driven video portraits. IEEE Trans Vis Comput Graph. 2020;26(12):3457–66.
23. Song L, Wu W, Qian C, He R, Loy CC. Everybody's talkin': let me talk as you want. arXiv preprint arXiv:2001.05201; 2020.
24. Thies J, Elgharib M, Tewari A, Theobalt C, Nießner M. Neural voice puppetry: audio-driven facial reenactment. Proceedings of the European Conference on Computer Vision. New York, NY: Springer; 2020. p. 716–31.
25. Aberman K, Weng Y, Lischinski D, Cohen-Or D, Chen B. Unpaired motion style transfer from video to animation. ACM Trans Graph(TOG). 2020;39(4):64–1.
26. Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434; 2015.
27. Yang H, Zhu H, Wang Y, Huang M, Shen Q, Yang R, Cao X. Facescape: a large-scale high quality 3D face dataset and detailed riggable 3D face prediction. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE; 2020. p. 601–610.
28. Qian K, Zhang Y, Chang S, Yang X, Hasegawa-Johnson M. Autovc: zero-shot voice style transfer with only autoencoder loss. Proceedings of the International Conference on Machine Learning. PMLR; 2019. p. 5210–19.
29. Qian K, Jin Z, Hasegawa-Johnson M, Mysore GJ. F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder. Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE; 2020. p. 6284–8.
30. Qualtrics, Provo, Utah; 2021.
31. Van der Maaten L, Hinton G. Visualizing data using T-SNE. J Mach Learn Res. 2008;9(11):2579–605.

AUTHOR BIOGRAPHIES

Che-Jui Chang is a Ph.D. student in the Computer Science Department of Rutgers University. He has a B.Sc. degree in Physics (2016) and an M.Sc. degree (2018) in Communication Engineering from National Taiwan University. His research interests include intelligent virtual humans, embodied conversational agents, and multimodal behavior synthesis.

Long Zhao is a research scientist at Google Research. He obtained his Ph.D. in Computer Science from Rutgers University in 2022, advised by Professor Dimitris N. Metaxas. Before that, he got his M.S. and B.Eng. in Software Engineering from Tongji University in 2015 and 2012, respectively. His current research interests lie primarily in self-supervised representation learning, vision-language models, and image or video generation.

Sen Zhang is a Ph.D. student in the Computer Science Department of Rutgers University. He has a master's degree in Electrical and Computer Engineering from Rutgers University and bachelor's degree in Telecommunications Engineering of the Xidian University, China. His research interests include autonomous animation and virtual reality.

Mubbasir Kapadia is the Director of the Intelligent Visual Interfaces Lab and an Associate Professor in the Computer Science Department at Rutgers University. Previously, he was an Associate Research Scientist at Disney Research Zurich and the Assistant Director of the Human Modeling Simulation Lab at University of Pennsylvania. Kapadia's research lies at the intersection of artificial intelligence, visual computing, and human-computer interaction, with a mission to develop intelligent visual interfaces to empower content creation for human-aware architectural design, digital storytelling, and games. He has published more than 100 journal and conference papers at premier venues in Computer Graphics, Computer Vision, and Artificial Intelligence. Kapadia's research is funded by DARPA and NSF, and through generous support from industrial partners including Disney Research, Autodesk Research, Adobe Research, and Unity Labs. He received his Ph.D. in Computer Science at University of California, Los Angeles.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Chang C-J, Zhao L, Zhang S, Kapadia M. Disentangling audio content and emotion with adaptive instance normalization for expressive facial animation synthesis. Comput Anim Virtual Worlds. 2022;33(3-4):e2076. https://doi.org/10.1002/cav.2076