Emotional Talking Face Generation Using Deformable Convolutional Networks

Lalit Bhagat
Computer Science department
University of California, Los Angeles
lalitbhagat7@ucla.edu

Parth Shettiwar
Computer Science department
University of California, Los Angeles
parthshettiwar@ucla.edu

Pranjal Jain
Computer Science department
University of California, Los Angeles
pranjal0000@ucla.edu

Sahil Bansal
Computer Science department
University of California, Los Angeles
sahilbansal@ucla.edu

Satvik Mehul Mashkaria
Computer Science department
University of California, Los Angeles
satvikm@ucla.edu

***Abstract*—Talking face generation is the process of generating a video of a person saying the input audio with appropriate lip and facial features synced at each time step. In audiovisual speech communication, visual emotion expression is crucial. Using deep learning, researchers have been able to make considerable progress in tackling this computer vision task. CNNs are designed to model big, unknown transformations, however they have limitations. In this work, we propose a hybrid approach by adding deformable layers and attention modules. We further compare our approach to purely deep learning based approaches and contrast them using the structural similarity (SSIM) to peak signal-to-noise ratio (PSNR) metrics and achieve state-of-the-art results.**

***Index Terms*—Deformable models, talking face generation, audiovisual, emotion**

I. Introduction

Emotion has a direct influence on the communicated message during voice communication and can substantially alter the meaning. In human-to-human or human-to-machine contact, visual and aural modalities are two essential sensory channels. It is critical for autonomous talking face generation systems to generate visual emotion expressions in order to make the visual depiction more realistic and enhance voice communication. Upon incorporating natural language processing, accuracy has been improved because language models provide probability scores of which words are more likely to have been uttered.

In loud surroundings and for the hard-of-hearing population, the inclusion of visual signals increases speech comprehension [1], [2], [3], [4]. As a result, researchers devised systems that can synthesize talking faces from audio automatically in order to offer visual clues when they are unavailable [5], [6], [7], [8], [9], [10], [11], [12]. These systems can help the hearing challenged people get access to more readily available audio-only materials while also improving the quality of human-computer interactions [13], [14].

Cross-modality learning and modeling, which includes computer vision, computer graphics, and multimedia, has recently gotten a lot of interest in multidisciplinary research [15], [16], [17], [18], [19]. CNNs [20] are widely used for tasks like segmentation [22], object detection [23] and image classification [21]. Although, they perform quite well on these tasks, they still have drawbacks.

[24] presented a approach disregarding the emotions indicated in the spoken audio and basing the production of the talking face on a separate emotion variable. It also gives behavioral scientists a valuable tool for conducting emotion-related tests that were previously impossible. This approach, however, uses CNNs and which is not capable of modeling geometric transformations. Therefore, we introduce these new modules that greatly enhance CNNs' capability of modeling geometric transformations.

In this project, we leverage the work of [24] and propose a hybrid end to end deep learning network system by adding deformable layers and attention mask that generates emotional talking faces from speech conditioned on two types of emotions: happy and sad. A speech utterance, a reference face picture, and a categorical emotion condition are all inputs to the network, which subsequently generates a talking face that is synced with the input speech and comprises emotional emotions. Our main contributions are as follows:

- We propose an improved talking face generation model for two categories of emotions.
- We add Deformable layers in image encoder that learn specific facial features image and they have discrimination in the latent space.
- We also use attention masks so that the model automatically learns specific facial features.
- Further we evaluate our model on different baselines, show our performance on quantitative an qualitative aspects and achieve state-of-the-art results.

The rest of the project report is organized as follows: Related

work and background study is discussed in section II. We describe the proposed method and objective functions in Section III. Then, we present experimentation details, and results in Section IV. Finally, we conclude the report in Section V.

II. Related work

A. Talking Face Generation

Previous talking face generation methods are broadly, video driven [25], [26], [27], [28], [29], [30], [31] or audio driven [32], [33], [34], [35], [36], [37], [38], [39], [40]. In video driven method only visual information whereas, both visual and auditory modalities are used in audio-driven approaches. Video-driven face reenactment transfers expression and head pose to a target face image. GANs is the most popular learning based method used these days. Other optimization based approaches uses 3DMM parameters for transferring expressions. A arbitrary talking face or specific talking face can be generated from a audio and a specified face as input. Encoder-Decoder is one of the successful model used. More improved techniques have been proposed for this task.

In recent years, scientists have become more interested in the autonomous generation of talking faces from voice. One method is to translate speech into facial landmarks first, then estimate video frames based on the expected landmarks. A technique has been proposed, which used LSTM to predict PCA coefficients of face from speech and them maps them. This method, however, is limited to a single speaker. Further many improvements have been done is this domain. [24] presented a approach disregarding the emotions indicated in the spoken audio and basing the production of the talking face on a separate emotion variable. This method is currently state of the art. This method used purely deep learning techniques. We apply deformable layers on top of the existing method and achieve state of the are results. Existing work in the area of emotive talking face creation is rather restricted.

B. Deformable Convolutional Networks

Deformable Convolutional Networks, first introduced by [41]. The regular grid R is reinforced with offsets in deformable convolution, and the sampling is done on the irregular and offset sites. The input feature map x, which defines the receptive field size and dilation, like $\mathcal{R} = \{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\}$, is sampled using a regular grid R. For each location p_0 on the output feature map y, we have $y(p_0)$, where p_n enumerates the locations in R.

$$\mathbf{y}\left(\mathbf{p}_0\right)=\sum_{\mathbf{p}_n\in\mathcal{R}}\mathbf{w}\left(\mathbf{p}_n\right)\cdot\mathbf{x}\left(\mathbf{p}_0+\mathbf{p}_n\right) \tag{1}$$

Deformable convolution adds offsets to the conventional grid sampling sites $\{\Delta\mathbf{p}_n \mid n=1,\ldots,N\}$ $(N=|\mathcal{R}|)$ in comparison to ordinary convolution, allowing for free form deformation of the sample grid.

$$\mathbf{y}\left(\mathbf{p}_0\right)=\sum_{\mathbf{p}_n\in\mathcal{R}}\mathbf{w}\left(\mathbf{p}_n\right)\cdot\mathbf{x}\left(\mathbf{p}_0+\mathbf{p}_n+\Delta\mathbf{p}_n\right) \tag{2}$$

Additional convolutional layers are used to learn the offsets Δp_n from the prior feature maps. As a result, the deformation is local, dense, and adaptively conditioned on the input attributes. Both the convolutional kernels for producing the output features and the offsets are learnt at the same time during training. The gradients are backpropagated by bilinear operations to learn the offsets.

RoI pooling is also introduced. We acquire $y(i,j)$ for the (i,j)-th bin $(0\le i,j<k)$, where n_{ij} is the number of pixels in the bin and offsets $\{\Delta\mathbf{p}_{ij} \mid 0\le i,j<k\}$ are added to the spatial binning positions by bilinear interpolation, as illustrated in formula 3 and 4.

$$\mathbf{y}(i,j)=\sum_{\mathbf{p}\in\mathrm{bin}(i,j)}\mathbf{x}\left(\mathbf{p}_0+\mathbf{p}\right)/n_{ij} \tag{3}$$

$$\mathbf{y}(i,j)=\sum_{\mathbf{p}\in\mathrm{bin}(i,j)}\mathbf{x}\left(\mathbf{p}_0+\mathbf{p}+\Delta\mathbf{p}_{ij}\right)/n_{ij} \tag{4}$$

It turns an arbitrary-sized rectangular area into fixed-size features. Adding offsets to enable adaptive component localisation for objects with varying forms is the underlying notion behind deformable convolution and deformable pooling. Furthermore, both modules are lightweight, requiring only a few parameters and computations for offset learning.

Based on deformable convolution v1, a new version called deformable convolution v2 was presented to enable the network take advantage of its greater modeling capacity. The main three changes were- 1). stacking more deformable Conv layers; 2). modulated deformable modules; 3). R-CNN feature mimicking. In this paper

C. Attention Mask

Attention mechanisms are a way of selectively focusing on certain parts of a sequence. Specifically, *self attention* relates different positions of a single sequence in order to compute a representation of the sequence and has been used in a variety of task, eg. abstractive summarization, textual entailment, and learning task-independent sentence representations. We incorporate self-attention in the Transformer network architecture. Learning long-range dependencies is a key challenge in many sequence transduction tasks. One key factor affecting the ability to learn such dependencies is the length of the paths that forward and backward signals have to traverse in the network. The shorter these paths between any combination of positions in the input and output sequences, the easier it is to learn long-range dependencies. In the transformer, this value is constant $\mathcal{O}(1)$. The attention function used in a Transformer can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the

corresponding key. Specifically, scaled-dot product attention is done by taking dot product of *Query* Q and *Key* K vectors. This is then divided by $\sqrt{d_k}$, where d_k is the dimension of K. This scalar value is multiplied by value vector V, and a softmax is taken to get a probability distribution.

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}((\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V})$$

Using multiple heads, rather than one, allows the language model to attend to information from different representation subspaces rather than from a single one. Because the layer after the concatenation is a feed forward network with an input dimension that does not depend on the number of heads, we take a linear combination of the concatenated vectors by multiplying with a matrix W^O so that the resulting vector has the same dimension as the input dimension of the feed forward network.

$$MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(head_1, \cdots, head_h).\mathbf{W}^O \quad (5)$$

where $head_i$ = $Attention(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V)$ The architecture described by [43] introduces masks in the self-attention layers.

III. Methodology

GAN framework is implemented and Figure 1 shows an overview of the system. [24] network architecture is build by leveraging [42]. The only modification is to add emotion as an input. [24] employs one discriminator to distinguish between emotions expressed in movies and another discriminator to discriminate between actual and generated video frames in the discriminator networks.

A. Generator

The generator network is made up of the sub-networks shown below:

1) Image Encoder: From the input condition face image, the image encoder creates an image embedding. It has six layers of 2-D convolutional layers with the (128, 3, 2), (256, 3, 2), (512, 3, 2), (512, 3, 2), (512, 3, 2), (512, 4, 1) filter counts, that are kernel sizes , and down-sampling factors. Each convolutional layer is followed by a ReLU activation with a 0.3 slope. Downsampling is accomplished using nearest-neighbor interpolation rather than strides. The video decoder receives the final image embeddings and intermediate representations over U-Net style skip connection.

2) Speech Encoder: The speech encoder takes the input voice waveform and converts it into a speech embedding. It has five convolutional layers in the time domain, each with one-dimensional kernels. (64, 63, 4), (128, 31,4), (256, 17, 2), (512, 9, 2), (16, 1, 1) are the number of filters, filter sizes, and strides. After each convolutional layer, we employ a ReLU activation with a 0.3 slope. After these five convolutional layers, we add a context layer to concatenate the past and future speech characteristics. By sending just every fifth frame to the next layer, the context layer lowers the 125 time-steps to 25 time-steps. As a result, our created videos have a frame rate of 25 frames per second (FPS). The context layer's output is supplied into a fully linked layer, which is followed by two LSTM layers that produce the speech embedding sequence.

3) Emotion Encoder: The emotion label is encoded as a one-hot vector first, then supplied into the emotion encoder. The emotion encoder projects the one-hot vector to an emotion embedding using a two-layer fully connected (FC) neural network. For each time step, the embedding is repeated. After each FC layer, we employ a ReLU activation with a 0.3 slope.

4) Noise Encoder: We create a noise vector from the conventional Gaussian distribution for each frame of the movie. This sequence of noise vectors is processed by a single-layer LSTM, which outputs the noise embedding. The goal of this module is to represent head motions that are unrelated to speech, picture, or emotion.

5) Video Decoder: The video decoder also takes emotion as a an input. The voice, picture, noise, and emotion embeddings are concatenated and sent into the decoder. The decoder employs convolutional layers and reshape operations to project the embeddings into 4 x 4 pictures for each time step. Except for the last layer, these 4 x 4 pictures are concatenated channel-wise with the skip connections originating from the image encoder in the U-Net way. Each convolutional layer is followed by a ReLU activation with a 0.3 slope, except for the final layer, which uses hyperbolic tangent activation instead since the pictures are normalized to have values between -1 and 1.

B. Discriminators

1) Emotion Discriminator: It is a video-based emotion classifier with the addition of a false video category. Its goal is to improve the emotional expression that our network generates.

2) Image Quality Discriminator(GAN style): Its goal is to increase the visual quality of the output video while maintaining the target identity throughout. First, we concatenate the target picture by repeating it for the number of frames in the input video. Each frame is then processed via five levels of two-dimensional convolutional layers.

C. Loss Function

In this paper, we use different loss functions for different parts of the pipeline: MRM loss, perceptual loss, frame discriminator loss and an emotion discriminator loss. The MRM loss improves mouth to audio synchronization, the perceptual

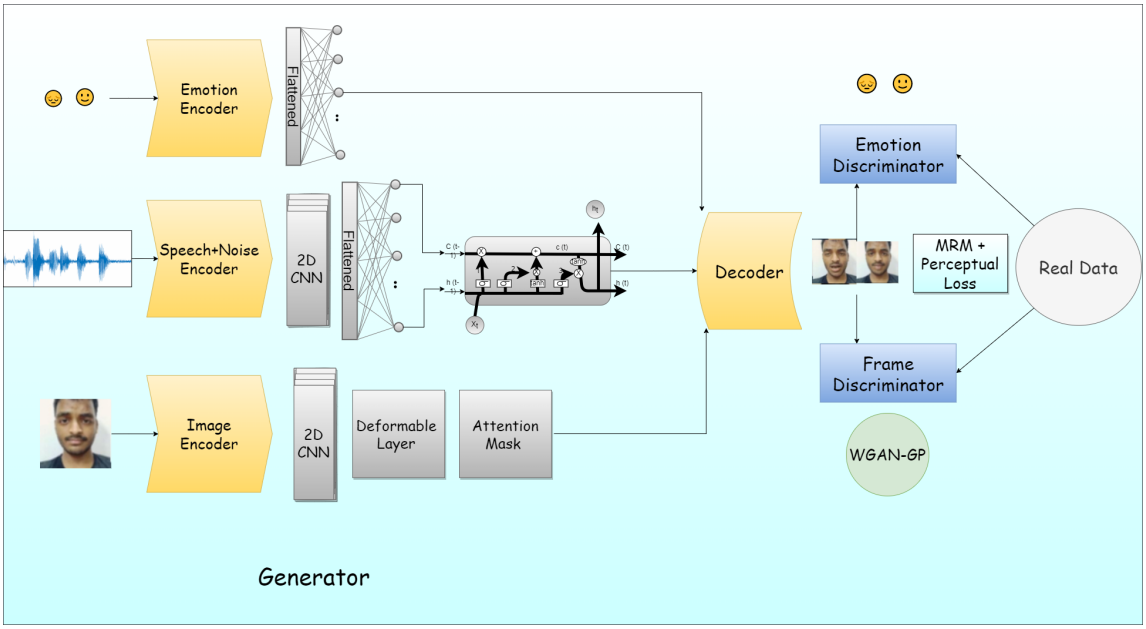

Fig. 1. Network Architecture

loss improves image quality, a frame GAN loss for image quality, and an image GAN loss for emotion expression.

1) Mouth Region Mask loss: It's found using the generated and ground-truth videos of the mouth. It's a weighted L1 reconstruction loss. By using a 2D Gaussian centered at the mean position of the mouth coordinates as the weights, the MRM intuitively focuses attention on the mouth region improving mouth to audio synchronization.
2) Perceptual Loss: A pre-trained model is used to calculate intermediate features from certain layers for both the generated and ground-truth videos. The mean-squared loss is between the intermediate features is used as the perceptual loss.
3) Frame Discriminator Loss: The sharpness is further improved using a frame GAN loss calculated by the discriminator. More stable training is achieved using a Wasserstein GAN.
4) Emotion discriminator loss: An emotion GAN loss is calculated using the emotion discriminator: a cross-entropy loss using two classes and a fake class. Because our problem specifically requires multiple classes, and a vanilla GAN would simply classify samples as real or fake, we use a multi-class version which incorporates multi-class classification losses and mitiages the issue of mode collapse.

The final loss function is a weighted combination of the four loss functions, where the weights of each loss is essentialy a hyperparameter.

D. Attention Module

In our network, such hints are realized by the attention mask, which are values between 0 and 1 indicating the importance of every element. The frame goes through one convolutional layer, before being mapped to the range of (0, 1) by an element-wise sigmoid function.

$$M = \sigma(conv_S(F))$$

$$F = M \cdot F$$

where M is the mask predicted by the network and F is the feature vector from the output of image encoder.

E. Deformable Convolution Module

Our deformable convolution module consists of 3 2D convolution modules of appropriate dimensions. The offset_conv calculates the offset, the modulator_conv calculates the masks and these are combined together using the torchvision ops module deform_conv2d to implement the deformable convolution module.

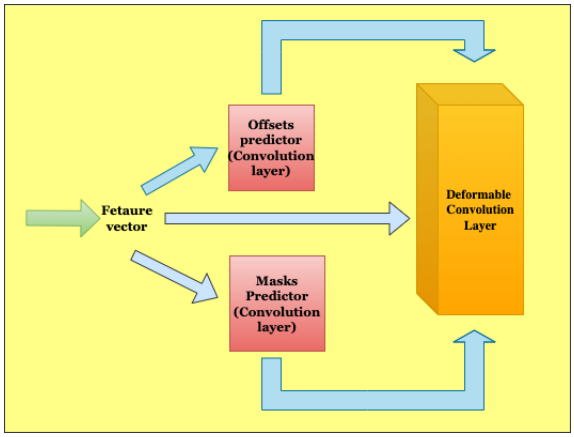

Fig. 2. Deformable Convolution Module

IV. Results

A. Dataset

In this project, we use Crowd-Sourced Emotional Multimodal Actors - Datset (CREMA-D) dataset [44]. It contains videos of 48 male and 43 female (91 individuals) actors. Each actor records video for a subset from a set of 12 sentences, and in six different emotions - happiness, sadness, neutral, anger, fear, disgust. Each emotion is expressed in four levels - Low, Medium, High and Unspecified. Dataset contains 7442 such video clips. Each video clip is 30 Frames Per Second (FPS) and each frame is of resolution 480×360. The audio is sampled at 44.1 kHz. For our experiments, due to limited time and compute resources, we only consider two emotions - Happy and Sad for randomly chosen individuals. Thus, we have 520 videos in our sub-sampled dataset. We consider a 80-10-10 split for train-test-validation sets.

B. Training and Implementation

We perform our training in three steps: first we only train generator. Then we train only discriminator keeping generator weights fixed. In the third step, we jointly train generator and discriminator. Such step-wise training has proven to be useful in complex vision tasks. Since joint alternative GAN-style training is difficult and unstable, first driving both the networks towards their respective minimas stabilizes the training. We have trained our models on Google Colab and P5 GPU on Hoffman Cluster. To be fair, we train our model and all baselines for same amount of time.

C. Results

In this section, we explore the qualitative and quantitative performance of our network and compare it with the baselines and groundtruth. For all experiments, we first trained the discriminator and used this pre trained discriminator, for the final joint training of the model. For [24], we loaded the pre-trained model which was provided, as part of joint training stage and noted the results. For our model, we pre-trained the generator before performing the joint training of the network.

D. Quantitative Comparisons

In order to make judgements about the quality of generated videos, we quantitatively compute the SSIM and PSNR. Structural similarity index (SSIM) is a metric used to measure the similarity between two given images. The SSIM extracts 3 key features from an image: luminance, contrast and structure and performs a comparison based on these.
The peak-signal-to-noise ratio (PSNR) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise. This ratio is used as a quality measurement between the ground-truth and generated image. We observe that our model achieves best performance on both of the metrics, beating the previous two baselines by a significant margin.

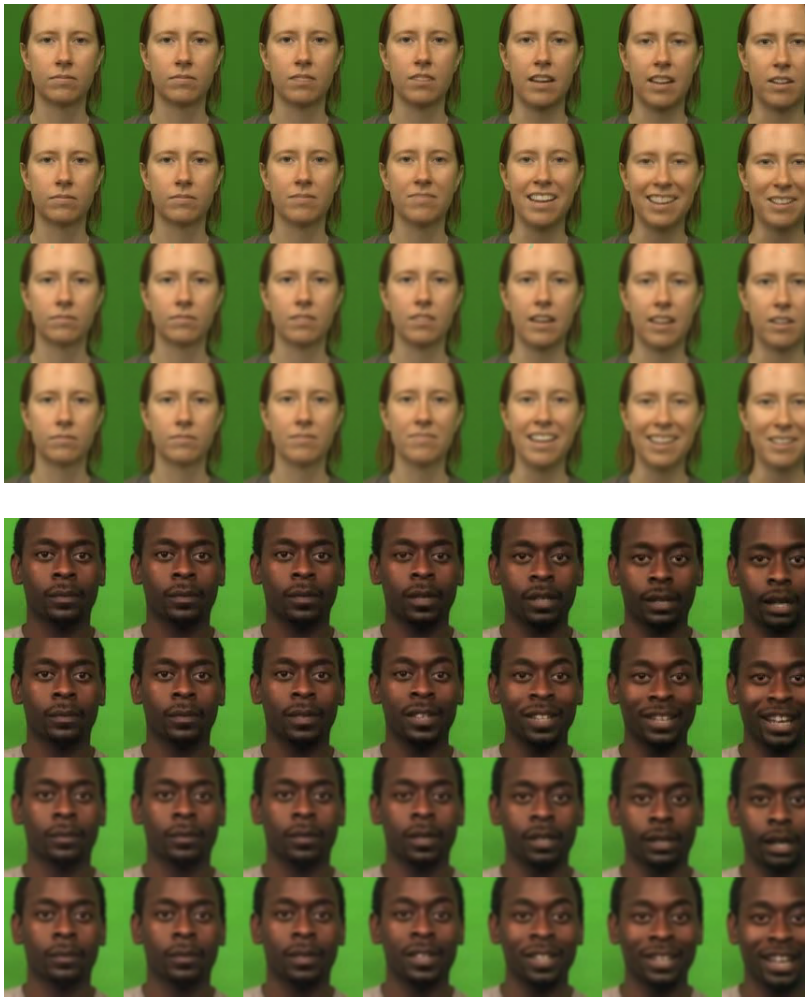

Fig. 3. Qualitative Results: Top 2 rows in each image correspond to the groundtruth video frames and bottom 2 rows correspond to the predicted video frames from our network

	PSNR	SSIM
Baseline1 [10]	22.01	0.61
Baseline2 [24]	22.94	0.63
Ours	**23.57**	**0.65**

TABLE I
Quantitative Comparisons with Baselines

E. Qualitative Comparisons

We performed an extensive qualitative analysis of our model and compared the outputs with the groundtruth. As can be seen in Figure 3, our network predicts almost accurate frame at each time step comparing to groundtruth, with little blurring of image. We suspect this due to the gaussian kernel being used in the MRM loss function. Hence we observe a tradeoff in using a gaussian kernel, where at first place it puts attention on the mouth features and ensures an accurate lip syncing, on other hand, this results in overall blurring of image. In the end, we alos tried to evaluate our mode on out of distribution images, specifically we tested our model on faces of our project team members. It was observed that model distorted the faces, but

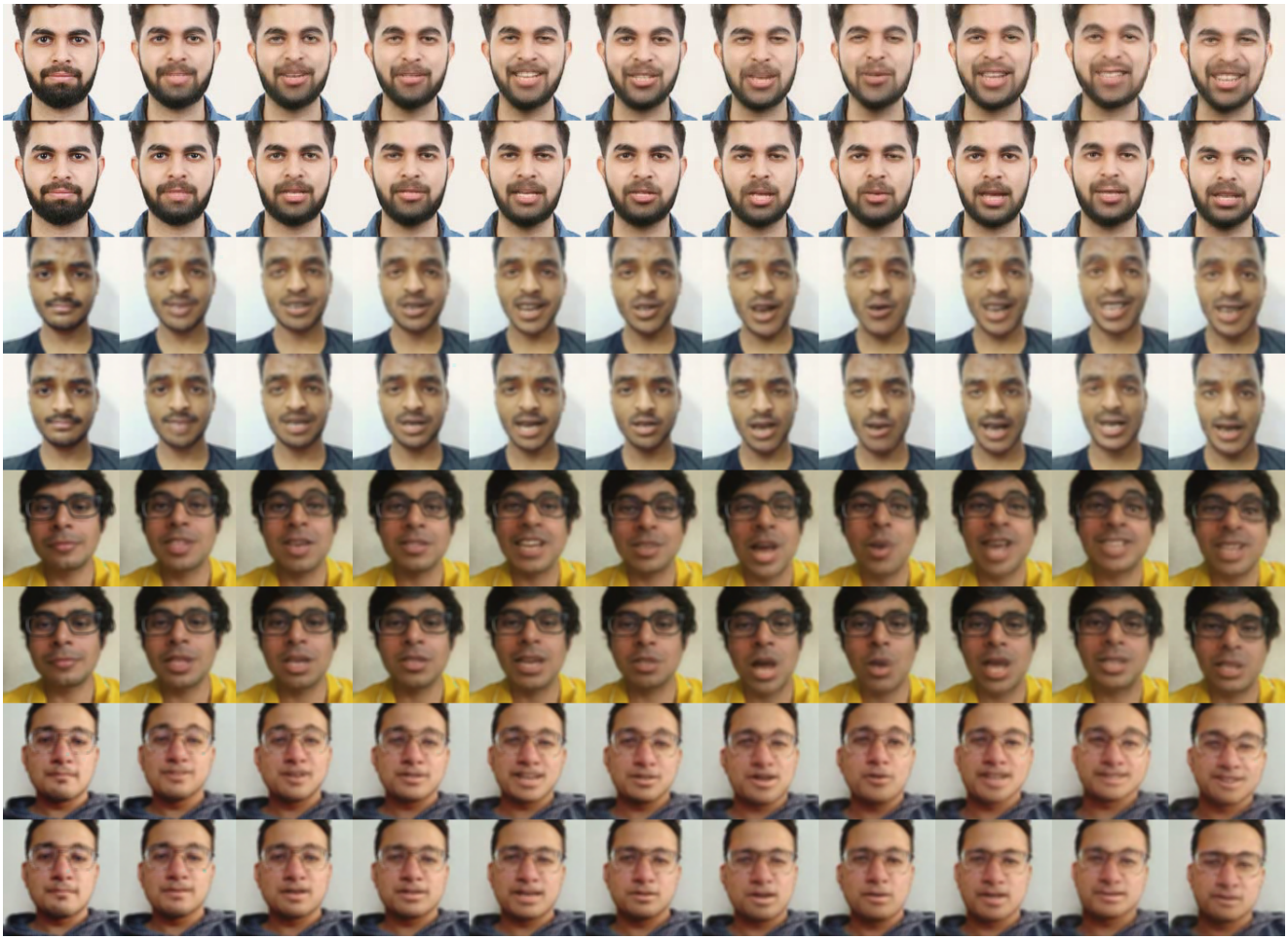

Fig. 4. Qualitative Results: Output of our model when the team members faces were given as input. We can clearly see, that even though these are out of distribution images, the network still generates emotional faces and lip features with very little distortion

at same time ensured that appropriate facial and lip features are synced with the audio

V. Conclusion and Future Work

In this project, we add deformable layers to the image encoder. This encourages better discrimination in the latent space. Furthermore, we add attention masks to focus on specific facial features. We use a hybrid approach to talking face generation, in contrast to previous approaches, which use a purely deep learning based approach. On comparison with other baselines, we observe that we achieve state-of-the-art results.

In future works, we aim to improve our pipeline by exploring better speech processing techniques such as FFT or MFCC to ensure improved learning and discriminated features in the latent space. We also aim to use multihead attention, used nowadays in vision transformers, to learn appropriate attention masks for each important facial feature. We further aim to modify our loss function to avoid gaussian blurring (we are currently exploring other bell-shaped functions like Cauchy distribution). Furthermore, currently our model was just trained on 2 emotions, we further aim to extend the emotion space by training on a much larger dataset.

References

[1] C. A. Binnie, "Bi-sensory articulation functions for normal hearing and sensorineural hearing loss patients," Journal of the Academy of Rehabilitative Audiology, vol. 6, no. 2, pp. 43–53, 1973.

[2] J. G. Bernstein and K. W. Grant, "Auditory and auditory-visual intelligibility of speech in fluctuating maskers for normal-hearing and hearingimpaired listeners," The Journal of the Acoustical Society of America, vol. 125, no. 5, pp. 3358–3372, 2009.

[3] K. S. Helfer and R. L. Freyman, "The role of visual speech cues in reducing energetic and informational masking," The Journal of the Acoustical Society of America, vol. 117, no. 2, pp. 842–849, 2005

[4] R. K. Maddox, H. Atilgan, J. K. Bizley, and A. K. Lee, "Auditory selective attention is enhanced by a task-irrelevant temporally coherent visual stimulus in human listeners," eLife, vol. 4, 2015.

[5] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical crossmodal talking face generation with dynamic pixel-wise loss," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[6] J. S. Chung, A. Jamaludin, and A. Zisserman, "You said that?" in British Machine Vision Conference (BMVC), 2017.

[7] S. E. Eskimez, R. K. Maddox, C. Xu, and Z. Duan, "Generating talking face landmarks from speech," in International Conference on Latent Variable Analysis and Signal Separation. Springer, 2018, pp. 372–381.

[8] A. Jamaludin, J. S. Chung, and A. Zisserman, "You said that?: Synthesising talking faces from audio," International Journal of Computer Vision, vol. 127, no. 11-12, pp. 1767–1779, 2019.

[9] Y. Song, J. Zhu, D. Li, A. Wang, and H. Qi, "Talking face generation by conditional recurrent adversarial network," in Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, 7 2019, pp. 919–925. [Online]. Available: https://doi.org/10.24963/ijcai.2019/129

[10] K. Vougioukas, S. Petridis, and M. Pantic, "Realistic speech-driven facial animation with gans," International Journal of Computer Vision, pp. 1–16, 2019.

[11] L. Yu, J. Yu, and Q. Ling, "Mining audio, text and visual information for talking face generation," in International Conference on Data Mining (ICDM). IEEE, 2019, Conference Proceedings, pp. 787–795.

[12] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang, "Talking face generation by adversarially disentangled audio-visual representation," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 9299–9306

[13] H. Tang, Y. Fu, J. Tu, M. Hasegawa-Johnson, and T. S. Huang, "Humanoid audio–visual avatar with emotive text-to-speech synthesis," IEEE Transactions on Multimedia, vol. 10, no. 6, pp. 969–981, 2008.

[14] O. Schreer, R. Englert, P. Eisert, and R. Tanger, "Real-time vision and speech driven avatars for multimedia applications," IEEE Transactions on Multimedia, vol. 10, no. 3, pp. 352–360, 2008.

[15] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical crossmodal talking face generation with dynamic pixel-wise loss," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7832–7841.

[16] J. S. Chung and A. Zisserman, "Lip reading in the wild," in 13th Asian Conference on Computer Vision (ACCV 2016), 2016, pp. 87–103.

[17] J. S. Chung, A. Jamaludin, and A. Zisserman, "You said that?" in British Machine Vision Conference (BMVC 2017), 2017.

[18] A. Davis, M. Rubinstein, N. Wadhwa, G. J. Mysore, F. Durand, and W. T. Freeman, "The visual microphone: passive recovery of sound from video," ACM Trans. Graph., vol. 33, no. 4, pp. 79:1–79:10, 2014.

[19] T. Oh, T. Dekel, C. Kim, I. Mosseri, W. T. Freeman, M. Rubinstein, and W. Matusik, "Speech2face: Learning the face behind a voice," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7539–7548.

[20] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 1995.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.

[23] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.

[24] Sefik Emre Eskimez, You Zhang, and Zhiyao Duan. Speech driven talking face generation from a single image and an emotion condition. IEEE Transactions on Multimedia, 2021.

[25] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Nießner, P. Perez, ´ C. Richardt, M. Zollhofer, and C. Theobalt, "Deep video portraits," ¨ ACM Trans. Graph., vol. 37, no. 4, pp. 163:1–163:14, 2018.

[26] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and ¨ M. Nießner, "Face2face: Real-time face capture and reenactment of RGB videos," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2387–2395.

[27] H. Averbuch-Elor, D. Cohen-Or, J. Kopf, and M. F. Cohen, "Bringing portraits to life," ACM Trans. Graph., vol. 36, no. 6, pp. 196:1– 196:13, 2017.

[28] A. Pumarola, A. Agudo, A. M. Mart´ınez, A. Sanfeliu, and F. Moreno-Noguer, "GANimation: Anatomically-aware facial animation from a single image," in 15th European Conference (ECCV), 2018, pp. 835–851.

[29] O. Wiles, A. S. Koepke, and A. Zisserman, "X2face: A network for controlling face generation using images, audio, and pose codes," in 15th European Conference (ECCV), 2018, pp. 690–706.

[30] E. Zakharov, A. Shysheya, E. Burkov, and V. S. Lempitsky, "Fewshot adversarial learning of realistic neural talking head models," CoRR, vol. abs/1905.08233, 2019.

[31] Y. Zhang, S. Zhang, Y. He, C. Li, C. C. Loy, and Z. Liu, "One-shot face reenactment," CoRR, vol. abs/1908.03251, 2019.

[32] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical crossmodal talking face generation with dynamic pixel-wise loss," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7832–7841.

[33] J. S. Chung, A. Jamaludin, and A. Zisserman, "You said that?" in British Machine Vision Conference (BMVC 2017), 2017.

[34] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama: learning lip sync from audio," ACM Trans. Graph., vol. 36, no. 4, pp. 95:1–95:13, 2017.

[35] Y. Song, J. Zhu, D. Li, A. Wang, and H. Qi, "Talking face generation by conditional recurrent adversarial network," in Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI 2019), 2019, pp. 919–925.

[36] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang, "Talking face generation by adversarially disentangled audio-visual representation," in The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019), 2019, pp. 9299–9306.

[37] O. Wiles, A. S. Koepke, and A. Zisserman, "X2face: A network for controlling face generation using images, audio, and pose codes," in 15th European Conference (ECCV), 2018, pp. 690–706.

[38] B. Fan, L. Wang, F. K. Soong, and L. Xie, "Photo-real talking head with deep bidirectional LSTM," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015), 2015, pp. 4884–4888.

[39] K. Vougioukas, S. Petridis, and M. Pantic, "Realistic speech-driven facial animation with GANs," International Journal of Computer Vision, DOI:10.1007/s11263-019-01251-8, 2019. [Online]. Available: https://doi.org/10.1007/s11263-019-01251-8

[40] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," CoRR, vol. abs/1912.05566, 2019.

[41] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In ICCV, 2017.

[42] S. E. Eskimez, R. K. Maddox, C. Xu, and Z. Duan, "End-to-end generation of talking faces from noisy speech," in International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 1948–1952.

[43] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[44] Cao, H., Cooper, D. G., Keutmann, M. K., Gur, R. C., Nenkova, A., and Verma, R. (2014). Crema-d: Crowd-sourced emotional multimodal actors dataset. IEEE transactions on affective computing, 5(4), 377-390.