Pretrained Audio Extractor is All You Need for better Lip Syncing

Anonymous
HKUST
Anonymous@connect.ust.hk

Advisor: Anonymous
HKUST
Anonymous@cse.ust.hk

Advisor: Anonymous
HKUST
Anonymous@connect.ust.hk

Abstract

This paper presents an advancement in the Wav2Lip framework by integrating state-of-the-art audio feature extraction methods, namely HuBERT and ContentVec, to enhance lip synchronization in talking head technologies. Our study focuses on improving the lip-syncing ability of the model, which is validated through human evaluations. Notably, despite the training on English speech data, the modified model demonstrates an impressive capacity to generalize to music audio and speech in various languages.

Our findings highlight the potential of advanced audio feature extraction methods in enhancing the performance of talking head models, while also emphasizing the necessity for more accurate evaluation methods and improvements in the visual aspects of these models. This study serves as a foundation for future research in the field of human-computer interaction and digital media, pushing the boundaries of artificial intelligence applications in realistic audio-visual synchronization.

1. Introduction

The advent of talking faces in digital media represents a groundbreaking convergence of artificial intelligence, computer vision, and human-computer interaction. This technology, at the forefront of modern computer science, has profound implications for how we interact with digital systems and consume media. By synthesizing human-like facial movements and speech, talking faces are not only enhancing the realism and engagement in digital communications but also revolutionizing fields such as virtual reality, entertainment, and automated customer service. The capability to produce convincing and natural talking faces is a testament to the significant advancements in machine learning algorithms and computational power, offering a window into the future of human-machine interactions.

However, there are significant challenges and grounds for enhancement in the widespread application of talking head technologies. Potential issues include unstable lip synchronization, suboptimal clarity of generated outputs, and susceptibility to background noise interference. This paper aims to address these shortcomings by developing an algorithm that is resilient to noise and capable of generating videos with stable and clear lip movements. Our approach builds upon the Wav2Lip [4] framework, which, while not the most advanced, serves as a suitable foundation for our study. Due to computational constraints, we utilize a moderately sized dataset HDTF [7]. Therefore, the primary objective of this research is not to set new benchmarks but to refine existing methodologies and offer insights for future investigations.

During the development phase, we encountered several engineering challenges. These included having clear box boundaries around the generated faces, issues in processing long videos, and slow data loading speeds. Despite these obstacles, effective solutions were implemented to successfully overcome these difficulties.

2. previous works

SyncNet [2] proposes a novel approach for determining the synchronization between video and audio in multimedia content. Building upon this, Wav2Lip [4] leverages SyncNet's proficiency in lip reading, guiding a generative model to produce videos that are in sync with the audio. Both the SyncNet and Wav2lip employs frequency-based techniques, such as Mel-Frequency Cepstral Coefficients (MFCC), combined with a Convolutional Neural Network (CNN) for audio feature extraction, while this work aims to explore more efficient ways to extract audio features.

Autovc [5] employs a simple autoencoder with a carefully designed bottleneck for voice conversion, conducting self-reconstruction and speaker disentanglement. Wav2Vec 2.0 [1] marks a significant advancement in audio feature extraction through a self-supervised learning framework. It predicts quantized latent representations for masked waveform segments, thereby enriching the acoustic information capture. HuBERT [3] introduces a novel pre-training strategy based on iterative self-supervised learning. This method involves generating pseudo-labels for both masked and unmasked audio segments, based on clusters from a previous iteration. This enables HuBERT to extract more

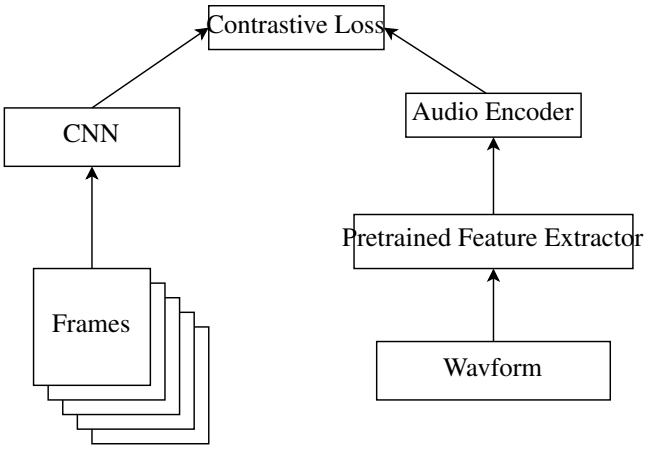

Figure 1. syncnet

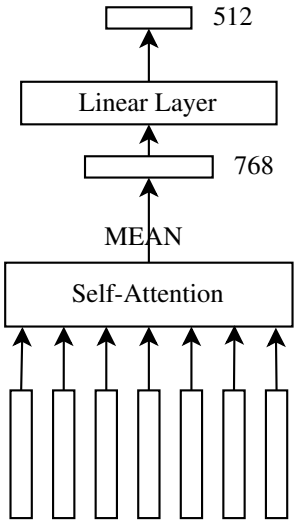

Figure 2. Audio Encoder

abstract and linguistically pertinent speech features. ContentVec [6] proposes a speech representation framework focusing on content retention and speaker invariance. Utilizing two losses, l_{contr} and l_{pred}, it ensures feature invariance to speaker-specific attributes while preserving speech content. The l_{contr} loss disentangles speaker characteristics, and l_{pred} aligns extracted features with HuBERT-generated ones, enhancing content integrity.

3. Method

3.1. SyncNet

The SyncNet structure is illustrated in Figure 1. For a 0.2-second video clip, the model employs both visual and audio feature extractors to derive the corresponding embeddings. The visual feature extractor, following the design in [4], stacks frames by channel and processes them through a Convolutional Neural Network (CNN) with skip connections to obtain the visual embedding. This work enhances the audio feature extractor originally described in [4]. While the original audio feature extractor in [4] utilizes frequency-based methods, such as Mel-Frequency Cepstral Coefficients (MFCC), followed by a CNN for feature extraction, our approach begins with a state-of-the-art pretrained feature extractor, such as HuBERT [3] or ContentVec [6], to acquire initial audio features. These features are subsequently fed into an audio encoder to derive the final audio embedding. The architecture of the audio encoder, depicted in Figure 2, averages the outputs of a self-attention layer and then applies a linear layer to yield a 512-dimensional embedding, aligning it with the dimensionality of the visual embedding.

During the training phase, the model determines the probability P_{sync} that the given audio-video pair is in sync. This probability is computed using the cosine similarity of the ReLU-activated embeddings for video $\mathbf{v}$ and speech $\mathbf{s}$, which is then passed through a binary cross-entropy loss function. The calculation of P_{sync} is as follows:

$$P_{\text{sync}} = \frac{\mathbf{v} \cdot \mathbf{s}}{\max(\|\mathbf{v}\|_2 \cdot \|\mathbf{s}\|_2, \epsilon)} \tag{1}$$

The resulting value ranges between [0, 1], indicating the likelihood of synchronization. The loss function is subsequently defined as:

$$\text{Loss} = -\left[y \log(P_{\text{sync}}) + (1-y) \log(1 - P_{\text{sync}})\right] \tag{2}$$

where y denotes the ground truth label, with 1 indicating that the audio-video pair is in sync and 0 indicating it is not. The term ϵ is a small constant added to prevent division by zero.

3.2. Wav2Lip

The Wav2Lip architecture is illustrated in Figure 3. Wav2Lip selects random out-of-sync frames from a video and masks the lower half of the frames to be predicted. These frames are input into the Visual Encoder, with the middle inputs and final output stored for later use. The audio is processed through a pretrained feature extractor, such as HuBERT or Contentvec, before being fed into the Audio Encoder, as shown in Figure 2. The embeddings from the visual and audio encoders are concatenated and passed into the Visual Decoder. A skip connection is employed to retain spatial information, such as edge locations, that may be lost during compression. The output of the generator is used to calculate the L1 Reconstruction Loss with the ground truth, detailed as:

$$L_{\text{recon}} = \frac{1}{N} \sum_{i=1}^{N} \|L_g - L_G\|_1 \tag{3}$$

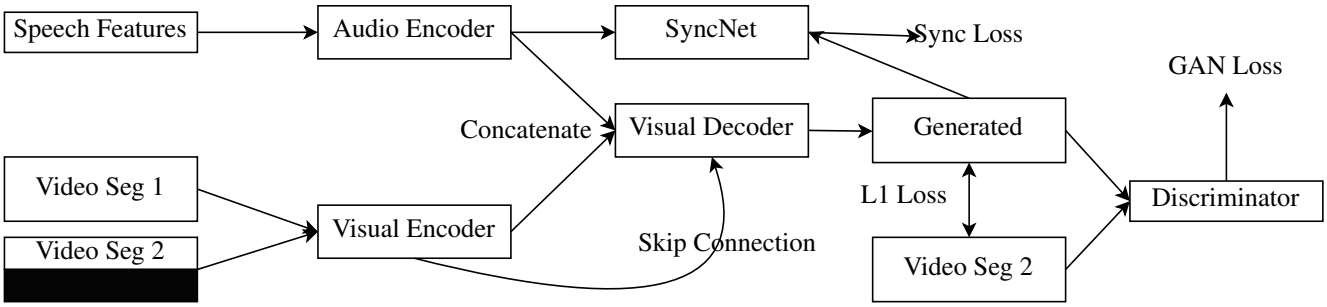

Figure 3. wav2lip

The generator also produces frames that are evaluated by the SyncNet for Sync Loss, computed by:

$$L_{\text{sync}} = \frac{1}{N}\sum_{i=1}^{N} -\log(p^{i}_{\text{sync}}) \tag{4}$$

Lastly, both the generated frames and the ground truth are input into the discriminator to compute the GAN Loss, which consists of the Generator Loss L_{gan} and the Discriminator Loss L_{disc}:

$$L_{\text{gen}} = \mathbb{E}_{x\sim L_g}[\log(1 - D(x))] \tag{5}$$

$$L_{\text{disc}} = \mathbb{E}_{x\sim L_G}[\log(D(x))] + L_{\text{gan}} \tag{6}$$

The generator aims to minimize the total loss L_{total}, which is the weighted sum of the reconstruction loss L_{recon}, the synchronization loss L_{sync}, and the adversarial loss L_{gan}. This is represented by Equation 6:

$$L_{\text{total}} = (1 - s_w - s_g) \cdot L_{\text{recon}} + s_w \cdot L_{\text{sync}} + s_g \cdot L_{\text{gan}} \tag{7}$$

In this equation, s_w is the weight assigned to the synchronization penalty, and s_g is the weight assigned to the adversarial loss.

In the inference phase, the Video Segement 1 and Video Segment 2 would be the same, being the videos clips to predict.

3.3. Contributions of our work

Our work is largely inspired by the Wav2Lip model [4], with significant enhancements to the audio feature extraction process. Traditional frequency-based methods, such as Mel-Frequency Cepstral Coefficients (MFCC), typically followed by Convolutional Neural Network (CNN) layers, have been supplanted in our approach by more advanced deep learning techniques. We utilize state-of-the-art models like HuBERT and ContentVec, which are then processed by an Audio Encoder architecture that leverages self-attention mechanisms. This shift from conventional spectral features to sophisticated neural embeddings allows for a more nuanced representation of the audio, potentially leading to improved lip synchronization in the generated video.

4. Experiments

4.1. Dataset

The dataset central to our study is the High-resolution Audio-visual Dataset (HDTF), accessible at https://github.com/MRzzm/HDTF. This carefully curated dataset is composed of video content, specifically speeches by American politicians, sourced from YouTube. Each video in the HDTF dataset is characterized by a resolution of 720p, ensuring high-definition visual quality. The frame rate of the videos is standardized at 25Hz, providing a consistent temporal resolution suitable for detailed audio-visual analysis. This dataset's focus on high-resolution video and audio content makes it an excellent resource for training self-supervised talking head algorithms.

4.2. Preprocessing

The preprocessing stage of our study involves several critical steps to prepare the dataset for analysis. Initially, videos lacking accompanying audio tracks, referred to as silent videos, are excluded from the dataset. Subsequently, each remaining video is segmented into 5-second clips, with the caveat that the final clip may be shorter than this duration.

For each video clip, the facial region is extracted using a sophisticated face detection method, detailed at https://github.com/1adrianb/face-alignment. This process also involves the extraction and storage of individual frames from these clips for further analysis.

A notable enhancement in our methodology addresses a limitation in the original Wav2Lip framework, which did not store extracted audio features, leading to inefficient data loading times. We circumvent this issue by storing speech features extracted using pretrained models such as HuBERT or ContentVec, thereby streamlining the data processing workflow.

Finally, the dataset is randomly partitioned into three distinct subsets: training (comprising 8,812 samples), validation (1,109 samples), and testing (1,093 samples). This partitioning facilitates a comprehensive and robust evaluation of our model's performance across different data segments.

4.3. SyncNet Training

In this study, three distinct versions of SyncNet are trained. While the visual feature extractor remains consistent with the original Wav2Lip model, modifications are made to the audio feature extractor. We experiment with three different audio feature extractors: the traditional Spectrogram (as used in Wav2Lip), HuBERT, and ContentVec.

To ascertain the optimal model, we rely on the best validation loss as the key criterion. The following table summarizes the best validation losses obtained for each architecture:

Table 1. Best Validation Losses for Different Audio Feature Extractors

Audio Feature Extractor	Best Validation Loss
Spectrogram	0.32
HuBERT	0.28
ContentVec	0.28

It is noteworthy that the training process exhibits a tendency towards overfitting. This issue is hypothesized to stem from two primary factors. First, the training does not exclusively focus on the lip region but instead uses the lower half of the face. Given the relatively small size of the dataset, this approach may introduce excessive and unnecessary details into the images, thereby contributing to overfitting. Second, the methodology employed for processing visual data may be suboptimal. The procedure involves stacking consecutive frames (five frames from a 0.2-second video clip) along the channel, resulting in a 15-channel input for a CNN. This approach, while straightforward, is potentially less effective for extracting sequential features compared to more sophisticated structures such as self-attention mechanisms or LSTM networks.

4.4. Wav2Lip Training

Following the successful training of SyncNet, the focus shifts to training the Wav2Lip model. During this process, SyncNet is frozen. Initially, the weight of the loss function L_{sync} is set to 0. This weight is then adjusted to 0.03 once the average L_{sync} falls below 0.75. If a Generative Adversarial Network (GAN) is employed, the weight of L_{gan} is set to 0.07.

The learning rate for the generator is initially set at 1×10^{-4}. However, using the same initial learning rate for the discriminator resulted in it overpowering the generator. To mitigate this, the initial learning rate for the discriminator is reduced to 1×10^{-5}.

Model selection is based on the L_{total} of the validation set. Four distinct models were trained, each retaining the visual encoder and decoder configurations of the original Wav2Lip, while varying in their audio encoder components. The configurations for the audio encoders are as follows: Spectrogram without GAN, ContentVec without GAN, ContentVec with GAN, and HuBERT with GAN.

4.5. Inference

In the inference phase, unlike the training phase where reference frames are randomly selected from the video and ground truth frames are presented with the lower half masked, both the reference frames and the corresponding lower half-masked frames used as inputs to the model are identical. This phase encountered two significant technical challenges:

1. The pretrained audio feature extractor was not adept at handling long audio sequences. To address this limitation, audio inputs are segmented into 5-second clips. This segmentation aligns with the audio length used during the training phase, ensuring consistency in processing.

2. A noticeable boxed region was observed in the generated results. To alleviate this issue, the mediapipe package is utilized for precise facial region recognition. The output, a rectangular region encompassing the face, is then processed using mediapipe to extract and accurately position the face within the generated image.

These adjustments and the use of additional tools like mediapipe significantly enhance the quality and accuracy of the generated results in the inference process.

4.6. Result Analysis

As previously mentioned, our study involves four distinct models, differentiated primarily by their audio feature extractors: Spectrogram, ContentVec, high-quality HuBERT (with GAN), and high-quality ContentVec (with GAN).

To assess the performance of these models, extensive experimental evaluations were conducted. The results of these experiments are available for download and further examination from the following Google Drive link: https://drive.google.com/drive/folders/1q42ab7gl8Yp-udnsHrbEEL8bqVECJLfk?usp=sharing.

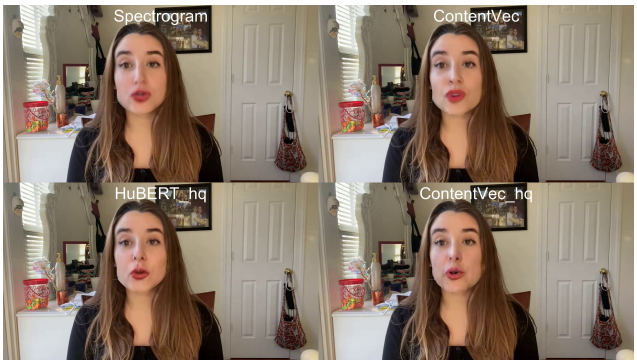

Figure 4. comparison of videos in the wild

Figure 5. comparison of videos in the training set

4.6.1 Videos in the wild

Our primary test involved analyzing the models' effectiveness on 'videos in the wild.' Specifically, six one-minute videos, all in English, were downloaded from YouTube. These videos then underwent an audio swap process. As shown in Figure 4, the results indicated that the Spectrogram model produced somewhat obscure outputs, whereas the ContentVec model yielded clearer yet still somewhat opaque results. Notably, both the HuBERT and ContentVec models with GAN demonstrated significant improvements in video clarity. However, it was observed that the visual quality of the generated videos by the GAN-enabled models on real-world videos still did not meet optimal standards.

As for the synchronization of lip movements, it was observed that the outputs generated using the Spectrogram audio feature extractor exhibited the most instability. In contrast, the models employing ContentVec and HuBERT as audio feature extractors produced lip movements that were generally well-aligned with the speech audio.

4.6.2 Videos in the dataset

Secondly, the models were evaluated using videos from the HDTF dataset, but with audio tracks sourced from YouTube, as depicted in Figure 5. This test revealed that the model employing ContentVec with GAN consistently outperformed the others, excelling both in visual quality and lip synchronization. Based on these findings, we have chosen to utilize the ContentVec with GAN model for subsequent experiments.

4.6.3 Noise, Music, and Multilingual Test

We conducted tests on the high-quality ContentVec model to evaluate its performance in handling speech with added background noise, music, and voices in other languages, specifically Chinese, Japanese, and Korean. The results were notably impressive. Despite the training set comprising solely clean English speech data, the model demonstrated a remarkable ability to generalize to noisy environments, musical backgrounds, and non-English audio. This adaptability is intriguing, considering that the HuBERT or ContentVec audio extractors were pretrained only on the LibriSpeech dataset, which contains English speech exclusively.

These findings suggest that speeches across different languages and forms of expression (such as normal speech or singing) may share common traits that are effectively captured by the audio extractor. However, it is important to acknowledge that the model's performance is not uniformly optimal. Certain scenarios, particularly those involving high-pitched and prolonged sounds characteristic of music (e.g., the elongated pronunciation in "loooove") but rare in ordinary speech, posed challenges for the model.

5. Conclusion

This study aims to enhance the Wav2Lip framework by substituting its audio extractor module with state-of-the-art audio feature extraction methods like HuBERT and ContentVec. This modification has notably improved the lip-syncing capabilities of our model, as evidenced by human evaluations of the generated results. Intriguingly, our experiments reveal that despite being trained exclusively on English speech data, the model exhibits a remarkable ability to generalize to music audio and speech in other languages.

However, our work is not without limitations. Firstly, we encountered challenges in identifying suitable evaluation metrics that exclusively measure lip synchronization. The LSE-D and LSE-C metrics proposed in Wav2Lip [4] often yield higher scores for outputs that are perceptibly inferior according to human assessments. Secondly, we did not refine the visual encoder and decoder components of the model. As a result, our system struggles to maintain high visual quality for videos in the wild.

These findings underscore the need for continued research in this field, particularly in developing more accurate evaluation methods and enhancing the model's visual

qualities.

References

[1] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*, 2020. 1

[2] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2016. 1

[3] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *arXiv preprint arXiv:2106.07447*, 2021. 1, 2

[4] K R Prajwal, Rudrabha Mukhopadhyay, Vinay Namboodiri, and C V Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. arXiv preprint arXiv:2008.10010, 2020. 1, 2, 3, 5

[5] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. Autovc: Zero-shot voice style transfer with only autoencoder loss. *arXiv preprint arXiv:1905.05879*, 2019. 1

[6] Kaizhi Qian, Yang Zhang, Heting Gao, Junrui Ni, Cheng-I Jeff Lai, David Cox, Mark Hasegawa-Johnson, and Shiyu Chang. Contentvec: An improved self-supervised speech representation by disentangling speakers. *arXiv preprint arXiv:2204.09224*, 2022. 2

[7] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3661–3670, 2021. 1