ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-6874-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSP49660.2025.10890415

Efficient Fusion of Computationally Diverse Modalities Using Chunking and Cross-Attention

Christian Flores[1], Lucas Goncalves[1], Carlos Busso[1,2]
[1]*The University of Texas at Dallas, Richardson, TX, 75080, USA*
[2]*Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, 15213, USA*
Christian.Flores@utdallas.edu, goncalves@utdallas.edu, busso@cmu.edu

***Abstract*—Emotion recognition is inherently a multimodal problem. Humans use both audible and visual cues to determine a person's emotions. There has been extensive improvement in the methods we use to fuse audio and visual representations between two unimodal deep-learning models. However, there is a lack of accommodation for modalities that have a disparity in the amount of computational resources needed to provide the same amount of temporal information. As the sequence length increases, current methods often make simplifications such as discarding frames or cropping the sequence. This paper introduces a chunking methodology designed for cross-attention-based multimodal transformer architectures. The approach involves segmenting the visual input—the more computationally demanding modality—into chunks. Cross-attention is then performed between the encoded audio and visual features instead of the original sequence lengths of the unimodal backbones. Our method achieves significant improvements over conventional cross-attention techniques in the audio-visual domain for a six-class emotional recognition problem, demonstrating better F1 score, precision, and recall on the CREMA-D database while reducing computational overhead.**

***Index Terms*—multimodal learning, audio-visual emotion recognition, deep learning**

I. Introduction

Emotion plays a crucial role in human communication. It shapes how we interact, make decisions, and form social connections. Human communication is intrinsically multimodal, relying on a combination of verbal and visual cues. Consequently, developing multimodal models is vital to providing a more complete and refined understanding of human communication.

Multimodal learning enables deep learning models to achieve a more robust and nuanced understanding of data by integrating information from multiple sources. This allows the model to capture relationships that unimodal models miss. Recently, studies have commonly used a cross-attention mechanism to fuse the features of two modalities over different sequence lengths [1]–[3]. However, this can be an issue when the modalities' model architectures may dictate different sequence sizes even after temporally aligning the samples. Furthermore, the vast difference between the sampling rates of two modalities, such as acoustic and visual data, inherently leads to a mismatch between the models.

The difference in sequence lengths in multimodal systems can limit the temporal information one modality can provide. For instance, video encoders have to be able to provide spatial information for every sample, not just temporal information like with audio, thus making computation costly [4], [5]. We can reduce the number of frames sampled throughout time to alleviate the computational cost (e.g., sampling frames or cropping the sequence length). However, this approach limits the temporal information available to the model. As a result, when fusing the audio and visual modalities using techniques such as cross-attention, the cross-attention mechanism can be hindered due to the visual encoder's reduced temporal information.

This paper proposes chunking the inputs of the more computationally intensive modality, the visual modality in this case, to both reduce computational costs and prevent the need to reduce temporal information provided by the modality. This approach enables cross-attention to be performed over a predefined number of chunks, reducing the model's complexity. Thus, the strategy allows for the fusion of the modalities without only relying on the sequence lengths dictated by the unimodal backbones.

Our proposed method achieved statistically significant improvements over conventional cross-attention techniques at a lower computational cost in a six-class emotion recognition task using the CREMA-D database [6]. Comparing our best-performing chunking method with our baselines, we observe an average of over 4% improvement in micro and macro F1 score, macro precision, and macro recall. We also observed over 58% reduction in required GPU memory and around 38% reduction in training time. We investigated different chunking strategies to identify critical factors influencing performance, such as chunk gap and overlap. We find that avoiding excessive overlap between the chunks is essential. While previous multimodal frameworks have used cross-attention, our key contribution is the efficient segmentation to combine the modalities, considering both local information conveyed in the chunks and global information achieved by combining the chunk-level representations.

II. Related Work

In general, one of the main challenges in emotion recognition and sequence-to-one tasks is effectively handling varying duration inputs. Previous studies have proposed to segment the audio waveforms into chunks for *speech emotion recognition* (SER) [7]–[10]. Studies have also expanded this method

This study was funded by the NSF under grant CNS-2016719.

to audio-text applications [11], showing the importance of synchronizing the two modalities even when ignoring exact lexical boundaries. Properly temporally synchronizing both modalities is critical to the performance of the multimodal model.

In the audio-visual space, studies have proposed methods to align speech to lips [12], [13]. An approach is to dynamically adjust the timing between the two modalities instead of utilizing a constant shift [13]. AlignNet [14] improved on this strategy by creating an end-to-end trainable model that also utilized time warping to align video and audio temporally.

Past research has focused mainly on time warping to align audio-visual modalities. However, this approach requires computational overhead and inherently distorts the modalities. In contrast, when deploying chunk segmentation, training and inference times can be reduced [8], and we can avoid the need to distort the modalities.

When it comes to using chunks, studies have explored the most effective way to aggregate the segments after passing them through deep learning models. Some proposed strategies use mean pooling, a gated network, a *recurrent neural network* (RNN) attention mechanism, and self-attention [8]. In the past, these studies focused on chunking inputs for RNNs and *convolutional neural networks* (CNNs).

Transformers have gained significant popularity [15]. Scaled dot-product attention can provide a rich understanding of the input by focusing on different parts of the input sequence [16]. More recently, studies have used cross-attention to enable one modality to influence another by cross-attending over the sequence lengths of each modality's backbone, as demonstrated in [17]. However, transformers have a downside. Their computational cost grows quadratically with the sequence length [16]. Several studies have attempted to reduce their complexity [18]–[21], but they all resulted in a reduction in performance. Research has been conducted on utilizing chunking in order to reduce computational overhead of transformers; however, these efforts have largely focused on unimodal tasks and transformers [22]. In contrast, we explore these strategies when applied to a cross-attention-based multimodal transformer architecture. Our approach investigates the benefits of cross-attending over chunks of video instead of the original sequence lengths to facilitate the fusion of the audio and visual modalities while reducing the computational cost.

III. Proposed Approach

This paper proposes an audio-visual model that chunks the visual input, the more computationally intensive modality, into predetermined segment lengths and fuses the two modalities by cross-attending over these chunks, as shown in Fig. 1. This approach allows us to retain temporal information of both modalities even when there is a vast difference in sequence lengths.

The chunking method we deploy is dynamic in nature [8]. We predefine the chunk length and number of chunks and then evenly space out these chunks across the modality's input. Since the computational cost of scaled dot-product

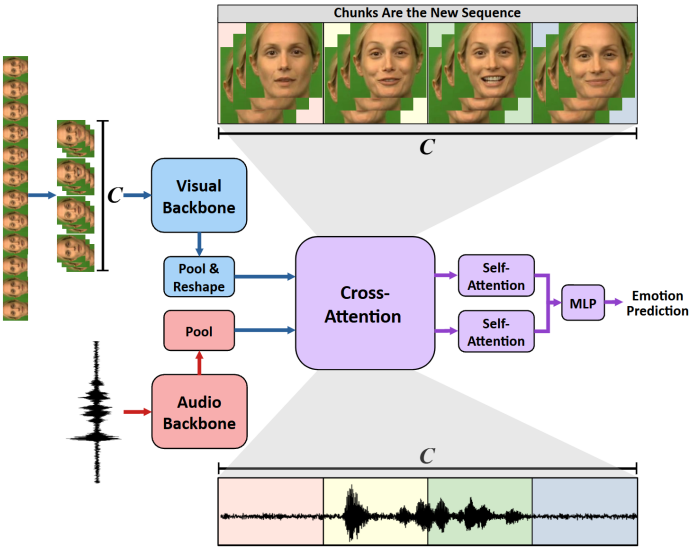

Fig. 1. Overview of our proposed audio-visual architecture using chunk-based segmentation. The blue branches represent the visual information, the red branches represent the acoustic information, and the purple branches represent the fused audio-visual information. The visual input is chunked before the visual backbone. The sequence length of both modalities is C when passed into the cross-attention module. The cross-attention module attends over the chunks instead of the standard sequence lengths of the backbones' outputs.

attention grows quadratically with respect to the sequence length [16], we can use dynamic chunking to control and reduce the effective sequence length without requiring us to discard temporal information.

Let L be the length of the input sequence, m be the fixed length of each chunk, and C be the number of chunks to create. The step size, Δ, between the start of each chunk varies between samples (assuming the samples have different durations) and is given by:

$$\Delta = \left\lfloor \frac{L-m}{C-1} \right\rfloor \tag{1}$$

For the i-th chunk, where $i \in [0, \ldots, C]$, the start index, s_i, which marks the beginning of the i-th chunk, is calculated as:

$$s_i = i \times \Delta \tag{2}$$

Similarly, the end index, e_i, which indicates the end of the i-th chunk, is calculated as:

$$e_i = s_i + m = i \times \Delta + m \tag{3}$$

Since each sequence is split into chunks, the visual modality's effective batch size is increased by a factor of C. We then pass the inputs through the respective unimodal backbones. Specifically, we use MARLIN [23] to handle the visual input, which is an encoder for facial video representations. We also use WavLM [24] to process the raw audio waveforms, which is a speech *self-supervised learning* (SSL) encoder. The output of the visual encoder is mean-pooled over the sequence dimension and then reshaped to make the chunks

the new sequence dimension, with a sequence length of C, to which cross-attention will be applied, as shown in Fig. 1. This approach reduces the visual modality's batch size to its original size. We use adaptive average pooling to adjust the sequence length of the output of the speech encoder to C to match the visual modality.

We utilize a cross-attention architecture [25] to facilitate information transfer between the modalities. We share the query vectors (Q) from one modality (M_0) with the *multi-head attention* (MHA) [16] layer of the other modality (M_1). The cross-attention mechanism is formalized as:

$$\text{Cross-Attn}\left(\Gamma\right) = \text{softmax}\left(\frac{Q_{M_1}{K_{M_0}}^\top}{\sqrt{d_0}}\right)V_{M_0} \qquad (4)$$

$$\text{where,} \quad \Gamma = \{Q_{M_1}, K_{M_0}, V_{M_0}\}$$

In (4), K_{M_0} and V_{M_0} are the key and value matrices of modality M_0, Q_{M_1} is the query matrix of modality M_1, and d_0 is the dimensionality of the key vectors. This mechanism effectively enables the fusion of information from both modalities, leveraging the temporal and contextual nuances captured in their respective segments.

The cross-attention process is repeated for five iterations. Then, the outputs go through five self-attention layers where the query matrix is not shared between modalities, allowing each modality's sequence to attend to itself. We apply mean pooling to combine the sequences for each modality, where each position's representation has been updated based on its relationship with all the other positions in the sequence by the self-attention layers, effectively aggregating the chunks. For final classification, the pooled outputs are concatenated and then passed through a *multilayer perceptron* (MLP).

IV. Experimental Settings

A. Database

The database we utilized in our experiments is the CREMA-D corpus [6]. CREMA-D is an audio-visual emotional dataset comprising data from 91 multi-ethnic actors aged 20 to 74. The actors were instructed to perform sentences expressing different emotions (disgust, anger, fear, happiness, sadness, or neutrality). The database contains 7,442 unique videos. A total of 2,443 annotators analyzed the videos and rated the emotions of each clip. In our experiments, we used the ratings provided by the annotators when they were given both the visual and the audio components of each clip. We utilized the plurality rule to determine the emotion of a clip. The data splits used during experimentation were speaker-independent.

B. Backbones

We used the pre-trained version of the base model WavLM [24] from Hugging Face [26] as the audio encoder and the base version of MARLIN [23] for the visual encoder. For each trial we ran on the multimodal models, we fine-tuned the backbones separately on each trial's training set, ensuring that our multimodal models were not influenced by the evaluation or testing sets (the trial setups will be further discussed in Section IV-E). We fine-tuned both backbones using Low-Rank Adaptation (LoRA) [27] deployed with the *parameter-efficient fine-tuning* (PEFT) library [28]. For the LoRA parameters, we used a rank of 32, an alpha of 64, and a dropout of 0.05. For WavLM, we injected the LoRA layers into the query, key, value, and output projections, as well as the intermediate dense and output dense layers. We trained WavLM on 30 epochs with a batch size of 32 using the AdamW optimizer [29] with a learning rate of 0.00005. For MARLIN, we injected the LoRA layers into the query, key, value, and output projections. We trained MARLIN on 30 epochs with a batch size of 16 using the Adam [30] optimizer with a learning rate of 0.0001.

TABLE I
Summary of Chunking Methods and Their Respective Gaps or Overlaps

Method	# of Chunks	Gap/Overlap (%)
Chunking 1	4	Gap: 50
Chunking 2	5	Gap: 12.5
Chunking 3	6	Overlap: 10
Chunking 4	8	Overlap: 35.7
Chunking 5	10	Overlap: 50

Note: Gap/overlap refers to the size of the gaps/overlaps between chunks as a proportion of the total duration of a single chunk for the average clip.

C. Baseline

For the baselines, we did not apply chunking to the inputs for either modality. We modified MARLIN's [23] encoder to use zero-padded masking to handle videos of different lengths in the absence of chunking. The audio encoder was left unchanged, as the pre-trained WavLM model [24] from Hugging Face [26] is already configured to handle masking by default.

For *Baseline 1*, we replaced the proposed five cross-attention layers with five self-attention layers. For *Baseline 2*, we retained the five cross-attention layers but attended to both backbones' original sequence lengths in the cross-attention layers instead of using chunks. As a result, the baselines have the same number of parameters as our proposed method.

D. Chunking Methods

In this study, we explored five different chunking strategies, which are summarized in Table I. For the visual modality, we selected a chunk length of 2 frames to align the sequence length with the audio encoder. For the audio modality, we did not utilize chunking due to its lower computational overhead. The audio backbone relies heavily on a more extended temporal range for feature extraction, whereas the visual backbone can effectively utilize spatial information from a single frame. Furthermore, the chunks were evenly distributed across the video, ensuring both modalities are synchronized in time.

We kept the chunk length constant and varied the number of chunks to assess how the number of chunks impacts the

TABLE II
PERFORMANCE METRICS FOR DIFFERENT ARCHITECTURES: MICRO AND MACRO F1 SCORES, MACRO PRECISION, AND MACRO RECALL

Architecture	Micro F1	Macro F1	Prec.	Rec.
Baseline 1	0.763	0.711	0.729	0.717
Baseline 2	0.769	0.720	0.727	0.726
Chunking 1	**0.793***†	**0.752***†	0.758*†	**0.760***†
Chunking 2	**0.793***†	0.750*†	**0.759***†	0.759*†
Chunking 3	**0.793***†	0.750*†	0.757*†	0.758*†
Chunking 4	0.788*†	0.743*†	0.751*†	0.748*†
Chunking 5	0.755	0.715	0.724	0.724

* and † indicate statistically significant improvements over *Baseline 1* and *Baseline 2*, respectively, based on a two-tailed t-test with $p < 0.05$.

cross-attention mechanism's ability to extract crucial temporal information between the two modalities.

E. Training

We trained each multimodal system for 20 trials, with 20 epochs per trial and a batch size of 16. We used the Adam [30] optimizer with a learning rate set at 0.0001. Both backbones were frozen when experimenting with the fused architectures to ensure fairness. We train all the models with the same partitions. The partitions are random splits of the CREMA-D dataset [6] such that all the data of each actor belong to the training, evaluating, or testing sets. Furthermore, we shared seeds across architectures to eliminate variability due to randomness. This approach ensures that any difference in performance between the architectures is due to the architectures themselves and not influenced by random factors such as weight initialization or data shuffling. For reference, the training times and memory usage were analyzed using the same Nvidia RTX 4090. We used PyTorch's built-in `cuda.max_memory_allocated` function to retrieve the GPU memory required for each architecture.

V. EXPERIMENTAL RESULTS

We compare the classification performance of our proposed chunking strategies with *Baseline 1*'s and *Baseline 2*'s results. Table II shows that most of the chunking settings evaluated in this study outperform the baselines. Notably, The strategies for *Chunking 1*, *Chunking 2*, *Chunking 3*, and *Chunking 4* show statistically significant improvements over the baselines, demonstrating the effectiveness of these approaches. For our best performing chunking method, *Chunking 1*, we see a 3.9% improvement in micro F1 score, 5.8% in macro F1 score, 4.0% in macro precision, and 6.0% in macro recall when compared to *Baseline 1*. Compared to *Baseline 2*, we see a 3.1% improvement in micro F1 score, 4.4% in macro F1 score, 4.2% in macro precision, and 4.7% in macro recall.

As can be seen by *Chunking 5* in Table II, excessive overlap leads to a noticeable decrease in performance compared to *Baseline 1* and *Baseline 2*. We hypothesize that the redundancy introduced by overlap creates noise in the fused representation, as the model encounters the same features multiple times. This repetition effectively amplifies their perceived importance during the attention mechanisms of both the cross-attention and self-attention layers. Consequently, the model may overfocus on the repetitive parts of the input sequence while neglecting potentially more informative sections.

TABLE III
GPU MEMORY REQUIRED AND TRAINING TIMES FOR DIFFERENT ARCHITECTURES

Architecture	Memory	Training Time
Baseline 1	18.887 GB	122 sec/epoch
Baseline 2	18.809 GB	120 sec/epoch
Chunking 1	7.848 GB	75 sec/epoch
Chunking 2	7.865 GB	88 sec/epoch
Chunking 3	7.889 GB	92 sec/epoch
Chunking 4	7.923 GB	108 sec/epoch
Chunking 5	7.958 GB	122 sec/epoch

We also compared the efficiency of our proposed chunking strategies with *Baseline 1* and *Baseline 2*, as shown in Table III. It is important to note that the number of trainable and non-trainable parameters is the same for all architectures tested. For our best performing chunking method, *Chunking 1*, we see a 58.4% reduction in GPU memory required when compared to *Baseline 1* and a 58.3% reduction in GPU memory needed when compared to *Baseline 2*. Furthermore, we see a 38.5% reduction in training time compared to *Baseline 1* and a 37.5% reduction in training time compared to *Baseline 2*. This analysis confirms our hypothesis that chunking the inputs of the computationally intensive modality and cross-attending over these chunks instead of the entire sequence length of the backbones significantly decreases the computational overhead and resources required by our proposed approach.

VI. CONCLUSIONS

This paper proposes a chunking method specifically tailored for cross-attention-based multimodal transformer architectures. The approach focuses on chunking the visual modality, which is the more computationally intensive component in audio-visual emotion recognition tasks. The video is split into segments, enabling cross-attention with audio over these chunks rather than relying on the standard sequence lengths of the outputs of the unimodal backbones. Our approach demonstrates significant improvements over conventional cross-attention methods in the audio-visual domain in the context of emotion recognition using the CREMA-D database. By segmenting video, our method leverages greater temporal information from the visual modality, thereby enhancing the effectiveness of the cross-attention mechanism and reducing computational overhead.

We can extend this method to other multimodal systems. Our strategy is more effective in scenarios where we need to accommodate modalities with a significant disparity in computational overhead for the same amount of temporal information.

REFERENCES

[1] Y.-H. Tsai, S. Bai, P. Liang, J. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," in *Association for Computational Linguistics (ACL 2019)*, vol. 1, Florence, Italy, July 2019, pp. 6558–6569.

[2] L. Goncalves, S.-G. Leem, W.-C. Lin, B. Sisman, and C. Busso, "Versatile audio-visual learning for emotion recognition," *IEEE Transactions on Affective Computing*, vol. Early Access, 2024.

[3] L. Goncalves and C. Busso, "Learning cross-modal audiovisual representations with ladder networks for emotion recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023)*, Rhodes island, Greece, June 2023, pp. 1–5.

[4] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" *CoRR*, vol. abs/2102.05095, 2021. [Online]. Available: https://arxiv.org/abs/2102.05095

[5] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," 2021. [Online]. Available: https://arxiv.org/abs/2103.15691

[6] H. Cao, D. Cooper, M. Keutmann, R. Gur, A. Nenkova, and R. Verma, "CREMA-D: Crowd-sourced emotional multimodal actors dataset," *IEEE Transactions on Affective Computing*, vol. 5, no. 4, pp. 377–390, October-December 2014.

[7] J.-K. H. Hyun-Sam Shin, "Performance analysis of a chunk-based speech emotion recognition model using rnn," *Intelligent Automation & Soft Computing*, vol. 36, no. 1, pp. 235–248, 2023. [Online]. Available: http://www.techscience.com/iasc/v36n1/50034

[8] W.-C. Lin and C. Busso, "Chunk-level speech emotion recognition: A general framework of sequence-to-one dynamic temporal modeling," *IEEE Transactions on Affective Computing*, vol. 14, no. 2, pp. 1215–1227, April-June 2023.

[9] R. Lin and H. Hu, "MissModal: Increasing Robustness to Missing Modality in Multimodal Sentiment Analysis," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1686–1702, 12 2023. [Online]. Available: https://doi.org/10.1162/tacl_a_00628

[10] W.-C. Lin and C. Busso, "An efficient temporal modeling approach for speech emotion recognition by mapping varied duration sentences into fixed number of chunks," in *Interspeech 2020*, Shanghai, China, October 2020, pp. 2322–2326.

[11] W.-C. Lin, L. Goncalves, and C. Busso, "Enhancing resilience to missing data in audio-text emotion recognition with multi-scale chunk regularization," in *ACM International Conference on Multimodal Interaction (ICMI 2023)*, Paris, France, October 2023, pp. 207–215.

[12] F. Tao and C. Busso, "Aligning audiovisual features for audiovisual speech recognition," in *IEEE International Conference on Multimedia and Expo (ICME 2018)*, San Diego, CA, USA, July 2018, pp. 1–6.

[13] T. Halperin, A. Ephrat, and S. Peleg, "Dynamic temporal alignment of speech to lips," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3980–3984.

[14] J. Wang, Z. Fang, and H. Zhao, "Alignnet: A unifying approach to audio-visual alignment," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 3309–3317.

[15] S. Latif, A. Zaidi, H. Cuayahuitl, F. Shamshad, M. Shoukat, and J. Qadir, "Transformers in speech processing: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2303.11607

[16] A. Vaswani *et al.*, "Attention is all you need," in *In Advances in Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, December 2017, pp. 5998–6008.

[17] G. Paraskevopoulos, S. Parthasarathy, A. Khare, and S. Sundaram, "Multiresolution and multimodal speech recognition with transformers," 2020. [Online]. Available: https://arxiv.org/abs/2004.14840

[18] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020. [Online]. Available: https://arxiv.org/abs/2006.04768

[19] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, D. Belanger, L. Colwell, and A. Weller, "Rethinking attention with performers," 2022. [Online]. Available: https://arxiv.org/abs/2009.14794

[20] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are rnns: Fast autoregressive transformers with linear attention," 2020. [Online]. Available: https://arxiv.org/abs/2006.16236

[21] N. Kitaev, Łukasz Kaiser, and A. Levskaya, "Reformer: The efficient transformer," 2020. [Online]. Available: https://arxiv.org/abs/2001.04451

[22] J. Xie, P. Cheng, X. Liang, Y. Dai, and N. Du, "Chunk, align, select: A simple long-sequence processing method for transformers," 2024. [Online]. Available: https://arxiv.org/abs/2308.13191

[23] Z. Cai, S. Ghosh, K. Stefanov, A. Dhall, J. Cai, H. Rezatofighi, R. Haffari, and M. Hayat, "Marlin: Masked autoencoder for facial video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2023, pp. 1493–1504.

[24] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, October 2022.

[25] L. Goncalves and C. Busso, "Robust audiovisual emotion recognition: Aligning modalities, capturing temporal information, and handling missing features," *IEEE Transactions on Affective Computing*, vol. 13, no. 4, pp. 2156–2170, October-December 2022.

[26] T. Wolf *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," 2020.

[27] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[28] S. Mangrulkar, S. Gugger, L. Debut, Y. Belkada, S. Paul, and B. Bossan, "Peft: State-of-the-art parameter-efficient fine-tuning methods," https://github.com/huggingface/peft, 2022.

[29] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2019. [Online]. Available: https://arxiv.org/abs/1711.05101

[30] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.