ASDNet at ActivityNet Challenge 2021 - Active Speaker Detection (AVA)

Okan Köpüklü[1], Maja Taseska[2], Gerhard Rigoll[1]

[1] Technical University of Munich

[2] Microsoft

1. Audio-Visual Active Speaker Detection

Audio-visual active speaker detection is a specific case of source separation, where audio and visual signals are leveraged jointly to assign a speech segment to its speaker. In this report, we propose a three-stage pipeline for audio-visual active speaker detection containing (i) audio-visual encoder, (ii) inter-speaker relation modeling and (iii) temporal modeling components. This is a brief explanation of the applied methodology. Please refer to [7] for detailed analysis of the applied methodology.

1.1. Notation and Overview

Let K denote the total number of speakers in a given clip. The data available to the active speaker detection system at time t is a set $\mathcal{X}_t = \{\boldsymbol{X}_{t,1}, \boldsymbol{X}_{t,2}, \dots \boldsymbol{X}_{t,K}, \boldsymbol{x}_t\}$, where $\boldsymbol{X}_{t,k} \in \mathbb{R}^{n \times 3 \times d_h \times d_w}$ is a tensor of face crops corresponding to the k-th speaker. The height and width of the face crops are denoted by d_h and d_w, 3 is the RGB channels and n is the number of consecutive face crops centering time t. The vector $\boldsymbol{x}_t$ contains the samples of the audio track corresponding to the duration of the video input. Given the input data, the objective of an active speaker detection system is to produce a binary vector $\boldsymbol{z}_t$, where $\boldsymbol{z}_t[k] = 1$ if the k-th speaker is detected as *speaking* at time frame t, and $\boldsymbol{z}_t[k] = 0$ otherwise. A high-level overview of our pipeline is illustrated in Fig. 1.

1.2. Audio-Visual Encoder Architecture

Our audio-visual encoder is illustrated in Fig. 2. The stack of face thumbnails $\boldsymbol{X}_{t,k}$ consists of n frames, $X_{t-\frac{n}{2},k}, \dots, X_{t,k}, \dots, X_{t+\frac{n}{2}-1,k}$, and the size of the audio input vector $\boldsymbol{x}_t$ is determined by the number of video frames, the frame rate of the video signal, and the sampling rate of the audio signal. The encoder produces an embedding vector by concatenating the following modality-specific embeddings

$$\mathbf{v}_{t,k} = f_v(\boldsymbol{X}_{t,k}; w_v), \quad \mathbf{a}_t = f_a(\boldsymbol{x}_t; w_a). \tag{1}$$

The embedding functions f_v and f_a are neural networks with trainable parameters w_v and w_a, respectively, which will be discussed shortly.

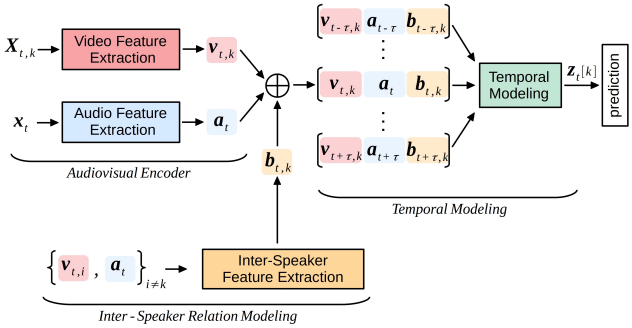

Figure 1. Overview of the proposed three-stage pipeline.

The concatenated features $\mathbf{v}_{t,k} \bigoplus \mathbf{a}_t$ are fed into a fully connected layer to get final predictions.Similar to previous works [1, 9, 11], we apply auxiliary classification networks after each backbone. After training is completed, supervision heads are discarded and only the audio-visual backbone is used to extract features $\mathbf{v}_{t,k}$ and $\mathbf{a}_t$ for all speakers and time instants.

Video backbone. We apply a 3D-Convolutional Neural Network (CNN) as our f_v. Our objective when choosing 3D-CNN was to benefit from its ability to capture motion patterns in face crops. Capturing motion patterns is crucial since movements of facial muscles and mouth are indicative of active speaking. We experimented with several high-performance and resource-efficient 3D-CNN architectures [6]. 3D-ResNeXt-101 performs best and becomes our final choice as video backbone and 16-frames clips are used as input to the video backbone. For all architectures, features before the fully connected layer are extracted as the video features $\mathbf{v}_{t,k}$.

Audio backbone. We propose an audio backbone architecture that directly operates on raw audio signal via sinc convolutions [10]. After sinc convolutions, we apply log-compression, i.e., $y = log(abs(x) + 1)$. This non-linearity has been effective in other raw audio processing tasks as well [8, 14]. The features extracted by the sinc-convolution block are used as input to Depthwise Separable Convolutional (DSConv) blocks with Leaky-ReLU nonlin-

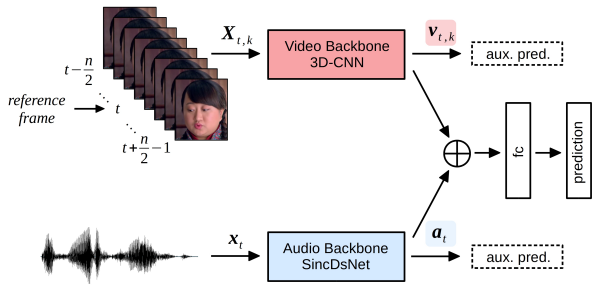

Figure 2. Audio-visual encoder architecture. Visual input $\boldsymbol{X}_{t,k}$ and audio input $\boldsymbol{x}_t$ are fed to the respective backbones to produce features $\mathbf{v}_{t,k}$ and $\mathbf{a}_t$. A concatenated feature vector $\mathbf{v}_{t,k} \bigoplus \mathbf{a}_t$ is fed to a fully connected layer which produces a prediction if speaker k is speaking at time t. Prediction heads are removed after training and are not part of the global picture in Fig. 1.

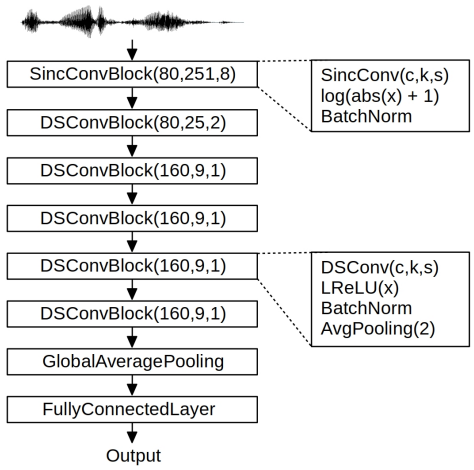

Figure 3. Audio feature encoding backbone utilizing Sinc Convolutions (SincConv) and Depthwise Separable Convolutions (DSConv). Convolution parameters are given as *c, k, s* representing the number of output channels, kernel size and applied stride, respectively.

earity [13]. The full audio encoder architecture, referred to as SincDSNet in the rest of the paper, is shown in Fig. 3. Features after the global average pooling are extracted as the audio features $\mathbf{a}_t$.

1.3. Inter-Speaker Relation Modeling (ISRM)

Consider a reference speaker k and m background speakers in the scene at time t. The output of the audio-visual encoder for the reference speaker is $[\mathbf{v}_{t,k}, \mathbf{a}_t]$. To incorporate information from background speakers, $[\mathbf{v}_{t,k}, \mathbf{a}_t]$ is concatenated with an additional feature vector $\mathbf{b}_{t,k}$, as illustrated in Fig. 4. The vector $\mathbf{b}_{t,k}$ is the output of a single-layer MLP whose inputs are the concatenated audio-visual embeddings from all background speakers at time t. Note that the number m is fixed from the system's perspective: if there are less than m background speakers at time t, the encoder features are populated with zero vectors. If there are more than m speakers, only m are randomly selected. In this manner, the input dimension of the MLP is fixed, and the feature vector $[\mathbf{v}_{t,k}, \mathbf{a}_t, \mathbf{b}_{t,k}]$ is fed to the temporal modeling mechanism, described next.

1.4. Temporal Modeling

We use 2 layer bidirectional Gated Recurrent Unit (GRU) [3] for temporal modeling. The reference frame is selected at the center of the input. The hidden state vector of the recurrent block at the reference frame is fed to a fully connected layer to produce a binary output $\mathbf{z}_t[k] \in \{0, 1\}$ (i.e. active speaker or not). In case speakers' features are not available for the selected time window, similar to [1] we apply same padding to the beginning or to the end. Out of all methods, Bidirectional-GRU performs best and becomes our final choice in temporal modeling stage.

1.5. Training Details

Training Audio-Visual Encoding Backbones. We train our audio-visual encoder using ADAM optimizer [5] for 70 epochs. Batch size is selected as highest possible number that fits to a single NVIDIA Titan XP GPU for different backbones. However, gradients are accumulated reaching to effective batch size of 192 before doing backward propagation. The learning rate is initialized with 3×10^{-4} and dropped by a factor of 10 at every 30 epochs. For video input, we apply random cropping, random horizontal flipping and color transformations as data augmentation at the training time. Finally, video input is reshaped to the resolution of 160×160. Audio data is extracted with sampling rate of 16 kHz. All 3D CNNs are pretrained on Kinetics [2] and SincDSNet is trained from scratch. Once the training is finished, prediction heads are discarded and the features $\mathbf{v}_{t,k} \in \mathbb{R}^{512}$ and $\mathbf{a}_t \in \mathbb{R}^{160}$ are used for the training of ISRM and temporal modeling stages.

Training ISRM and Temporal Modeling. We have again used ADAM optimizer with cross-entropy loss to train ISRM and temporal modeling stages. We train with batch size of 256 and for 10 epochs. Learning rate is initialized with 3×10^{-6} and dropped by 10 at 5th epoch. For

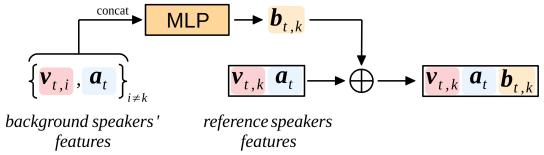

Figure 4. Inter-speaker relation modeling architecture. For reference speaker k at time instant t, we extract background features $\mathbf{b_{t,k}}$ by passing the concatenated features of background speakers through one layer MLP. Extracted features are then concatenated to reference speakers video features and audio features.

	Method	mAP
validation set	ASDNet (ours)	**93.5**
	MAAS-TAN [9]	88.8
	Chung et al. [4]	87.8
	ASC [1]	87.1
	Zhang et al. [15]	84.0
	Sharma et al. [12]	82.0
	Roth et al. [11]	79.2
test set	ASDNet (ours) *	**91.9**
	ASDNet (ours)	91.7
	Chung et al. [4]	87.8
	ASC [1]	86.7
	Zhang et al. [15]	83.5
	Roth et al. [11]	82.1

Table 1. Comparison with state-of-the-art methods on the AVA-ActiveSpeaker dataset. For model denoted with *, we have included validation set to the training.

ISRM, MLP creates the feature $\mathbf{b}_{t,k} \in \mathbb{R}^{128}$ independent from the number of background speaker features used. For RNN blocks, we always used two recurrent layers with hidden state dimension of 128, which experimentally proved to be working best.

2. Results

We compare the performance of ASDNet with several state-of-the-art methods in Table 1. For the final ASDNet, we used 16-frame clips at the audio-visual encoding stage, 3 background speakers with 9 neighbouring window at the ISRM stage, and bidirectional-GRU with 64-frame clips at the temporal modeling stage. ASDNet achieves 93.5 mAP at the validation set outperforming the second best approach by 4.7 mAP. On the test set, ASDNet achieves 91.9 mAP if validation set is used at the training and 91.7 mAP if only training set is used. Consequently, ASDNet outperforms the second best approach by 4.1 mAP on the test set of AVA-ActiveSpeaker dataset.

References

[1] Juan León Alcázar, Fabian Caba, Long Mai, Federico Perazzi, Joon-Young Lee, Pablo Arbeláez, and Bernard Ghanem. Active speakers in context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12465–12474, 2020. 1, 2, 3

[2] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2

[3] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014. 2

[4] Joon Son Chung. Naver at activitynet challenge 2019–task b active speaker detection (ava). *arXiv preprint arXiv:1906.10555*, 2019. 3

[5] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[6] Okan Kopuklu, Neslihan Kose, Ahmet Gunduz, and Gerhard Rigoll. Resource efficient 3d convolutional neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1

[7] Okan Köpüklü, Maja Taseska, and Gerhard Rigoll. How to design a three-stage architecture for audio-visual active speaker detection in the wild. *arXiv preprint arXiv:2106.03932*, 2021. 1

[8] Ludwig Kürzinger, Nicolas Lindae, Palle Klewitz, and Gerhard Rigoll. Lightweight end-to-end speech recognition from raw audio data using sinc-convolutions. *Proc. Interspeech 2020*, pages 1659–1663, 2020. 1

[9] Juan León-Alcázar, Fabian Caba Heilbron, Ali Thabet, and Bernard Ghanem. Maas: Multi-modal assignation for active speaker detection. *arXiv preprint arXiv:2101.03682*, 2021. 1, 3

[10] Mirco Ravanelli and Yoshua Bengio. Speaker recognition from raw waveform with sincnet. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 1021–1028. IEEE, 2018. 1

[11] Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, et al. Ava active speaker: An audio-visual dataset for active speaker detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4492–4496. IEEE, 2020. 1, 3

[12] Rahul Sharma, Krishna Somandepalli, and Shrikanth Narayanan. Crossmodal learning for audio-visual speech event localization. *arXiv preprint arXiv:2003.04358*, 2020. 3

[13] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015. 2

[14] Neil Zeghidour, Nicolas Usunier, Iasonas Kokkinos, Thomas Schaiz, Gabriel Synnaeve, and Emmanuel Dupoux. Learning filterbanks from raw speech for phone recognition. In *2018 IEEE international conference on acoustics, speech and signal Processing (ICASSP)*, pages 5509–5513. IEEE, 2018. 1

[15] Yuan-Hang Zhang, Jingyun Xiao, Shuang Yang, and Shiguang Shan. Multi-task learning for audio-visual active speaker detection. 3