End-to-End Active Speaker Detection (Supplementary Material)

Juan León Alcázar[1], Moritz Cordes[1,2], Chen Zhao[1], and Bernard Ghanem[1]

[1] King Abdullah University of Science and Technology, KAUST
{juancarlo.alcazar,chen.zhao,bernard.ghanem}@kaust.edu.sa
[2] Leuphana University Lüneburg moritz.cordes@stud.leuphana.de

1 Training details

We implement the audio encoder f_a with the Resnet18 convolutional encoder [4] pre-trained on ImageNet [2]. We adapt the raw 1D audio signal to fit the input of a 2D encoder by generating Mel-frequency cepstral coefficients (MFCCs) of the original audio clip, and then averaging the filters of the network's first convolutional layer to adapt for a single channel input [11]. We create the MFCCs with a sampling rate of 16 kHz and an analysis window of 0.025 ms. Our filter bank consists of 26 filters and a fast Fourier transform of size 256 is applied, resulting in 13 cepstrums. The visual encoder f_v is based on the R3D architecture, pre-trained on Kinetics-400 dataset [6]. For fair comparison with other methods, we also implement f_v as a 2D encoder by stacking the temporal and channel dimensions into a single one, then we replicate the filters on the encoder's first layer to accommodate for the input of dimension (B, CT, H, W) [12,11]. We also rely on ImageNet pre-training [2] for this encoder.

Φ Embedding We assemble Φ on-the-fly with parallel forward passes of f_a, f_v, and then map Φ into nodes of the Graph Convolutional Network and continue with the GCN in a single forward pass. We design the GCN module using the pytorch-geometric library [3] and use the EdgeConvolution operator [13] with filters of size 128. Each layers on the spatio-temporal module contains a single iGNN block. EdgeConvolution allows to build a sub-network that performs the message passing between nodes, where every layer (spatial or temporal) in the iGNN is built by a sub-network of two linear layers with ReLu [9] and batch normalization [5]. Therefore, a single iGNN block contains 4 linear layers in total.

Training EASEE. We Train EASEE for a total of 12 epochs[3] using the ADAM optimizer [7], and supervise every node in the final layer with the Cross-Entropy Loss. We also apply intermediate supervision at the end of f_a and f_v encoders [11].

[3] Similar to [1], we find that sampling every element in the tracklet leads to overfit. For every training epoch, we randomly sample only 4 training examples inside every tracklet.

We empirically observe that this favors faster learning and provides a small performance boost. The learning rate is set to 3×10^{-4} and is decreased with annealing $\gamma = 0.1$ at epochs 6 and 8. This very same procedure is applied regardless of the backbone. For every experiment we use a crop size of 160×160.

2 Challenging Scenarios Analysis

We complement the analysis of EASEE, and assess its performance in known challenging scenarios. We follow the procedure of [11], and evaluate EASEE in the AVA-ActiveSpeakers dataset according to: i) number of visible faces, and ii) the size of the face.

Table 1 shows the ablation of the performance of EASEE according to the face size. Overall, EASEE shows a similar behavior to state-of-the-art methods, where smaller faces (less than 64×64) are harder to classify (79.3 mAP). Medium images (between 64×64 and 128×128) show an improvement in performance over small images, and large faces report a the highest mAP at 97.7 mAP.

Faces Size	**EASEE-50**	**ASD** [8]	**MAAS** [10]	**ASC** [1]	**AVA Baseline** [11]
Small	**79.3**	74.3	55.2	44.9	56.2
Medium	**93.2**	89.8	79.4	68.3	79.0
Large	**97.7**	96.3	93.0	86.4	92.2

Table 1: **AVA-ActiveSpeaker Face Size.** We evaluate EASEE in the AVA-ActiveSpeaker dataset according to the size of the faces. As observed in previous works smaller faces are harder to classify. EASEE outperforms the state-of-the art in every scenario

Table 2 evaluates the performance of EASEE according to the number of simultaneous faces. Just like other ensemble methods, EASEE shows an improved performance in the mutli-speaker scenario when compared to the single speaker baseline [11] (20.8 mAP improvement for two speakers, 29.5 mAP improvement for 3 speakers).

Number of Faces	**EASEE-50**	**ASD** [8]	**MAAS** [10]	**ASC** [1]	**AVA Baseline** [11]
1	**96.5**	95.7	93.3	91.8	87.9
2	**92.4**	92.4	85.8	83.8	71.6
3	**83.9**	83.7	68.2	67.6	54.4

Table 2: **Performance evaluation by number of faces.** We evaluate EASEE in the AVA-ActiveSpeaker according to the number of visible faces (tracklets) in the scene. Multi-speaker scenes are far more challenging, our method outperforms the current state-of-the-art in any scneario.

3 Additional Ablation Experiments

We complement the ablation analysis of Section 4, and proceed to analyze two extra architectural decisions in EASEE: i) The effect of the number of iGNN modules, and ii) the size (number of neurons) in the linear layers in the iGNN blocks.

We first analyze the effect of the number of iGNN blocks. We control this hyper-parameter for the Resnet50 Backbone and the Resnet18 Backbone, and evaluate from 2 to 7 iGNN modules. Table 3 summarizes the results. Deeper GNN networks lead to higher performance, but this improvement stalls at 4 iGNN blocks for the Resnet50 backbone and 6 iGNN blocks for the Resnet18.

Backbone	2 iGNN	3 iGNN	4 iGNN	5 iGNN	6 iGNN	7 iGNN
EASEE-18	92.8	93.0	93.2	93.2	**93.3**	93.2
EASEE-50	93.6	93.8	**94.1**	94.0	93.8	93.8

Table 3: **EASEE Performance By iGNN Depth.** We analyze the effect of the number of iGNN blocks in EASEE. Stacking blocks improves the performance util 4 blocks are stacked (Resnet50) or 6 blocks are stacked (Resnet18)

We conclude by analyzing the effect of the size of the linear layers used in iGNN. Our best models (EASEE-50 & EASEE-18) use linear layers of size 128. In table 4 we ablate the size of this layer in the EASE50 architecture. We see a smaller impact on this hyper-parameter, where a smaller net only losses 0.3 mAP, and iGNN blocvks with double the number of neurons only loose 0.2 mAP.

Backbone	64	128	224	256
EASEE-50	93.8	**94.1**	93.9	93.9

Table 4: **Linear layer size.** We assess the effect of the layer size in the iGNN module. We find slightly reduced performance by altering the size of the iGNN module.

References

1. Alcázar, J.L., Caba, F., Mai, L., Perazzi, F., Lee, J.Y., Arbeláez, P., Ghanem, B.: Active speakers in context. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12465–12474 (2020)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
3. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
5. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456. PMLR (2015)
6. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
8. Köpüklü, O., Taseska, M., Rigoll, G.: How to design a three-stage architecture for audio-visual active speaker detection in the wild. arXiv preprint arXiv:2106.03932 (2021)
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**, 1097–1105 (2012)
10. León-Alcázar, J., Heilbron, F.C., Thabet, A., Ghanem, B.: Maas: Multi-modal assignation for active speaker detection. arXiv preprint arXiv:2101.03682 (2021)
11. Roth, J., Chaudhuri, S., Klejch, O., Marvin, R., Gallagher, A., Kaver, L., Ramaswamy, S., Stopczynski, A., Schmid, C., Xi, Z., 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). pp. 4492–4496. IEEE (2020)
12. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. Advances in neural information processing systems **27** (2014)
13. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. Acm Transactions On Graphics (tog) **38**(5), 1–12 (2019)