AUDIO-VISUAL SCENE CLASSIFICATION USING TRANSFER LEARNING AND HYBRID FUSION STRATEGY

Technical Report

Meng Wang[1,2], *Chengxin Chen*[1,2], *Yuan Xie*[1,2], *Hangting Chen*[1,2], *Yuzhuo Liu*[1,2], *Pengyuan Zhang*[1,2*]

[1] Key Laboratory of Speech Acoustics & Content Understanding, Institute of Acoustics, CAS, China
[2] University of Chinese Academy of Sciences, Beijing, China

ABSTRACT

In this technical report, we describe the details of our submission for DCASE2021 Task1b. This task focuses on audio-visual scene classification. We use 1D deep convolutional neural network integrated with three different acoustic features in our audio system, and perform a two-stage fine-tuning on some pre-trained models such as ResNet-50 and EfficientNet-b5 in our image system. In model-level fusion, the extracted audio and image embeddings are concatenated as input into a classifier. We also use decision-level fusion to make our system more robust. On the official train/test setup of the development dataset, our best single audio-visual system obtained a 0.159 log loss and 94.1% accuracy compared to 0.623 and 78.5% for the audio-only system and 0.270 and 91.8% for the image-only system. Our final fusion system could achieve a 0.143 log loss and 95.2% accuracy.

***Index Terms*—** Acoustic scene classification, Image scene classification, Convolutional neural network, Wavelet, Transfer learning

1. INTRODUCTION

Acoustic scene classification (ASC) is aimed at classifying a test recording into one of predefined acoustic scene classes. Detection and Classification of Acoustic Scenes and Events (DCASE) challenge has organized ASC task for many years. DCASE2021 Task1b [1] introduces an audio-visual dataset [2] for the first time. The goal of this task turns into audio-visual scene classification (AVSC), using joint modeling of the audio and video content.

We describe our submitted systems for DCASE2021 Task1b in this report. In our audio system, we use three different acoustic features, Mel filter bank feature, scalogram extracted by wavelets and bark filter bank feature. We feed these features into 1D deep convolutional neural networks (DCNN) to extract audio embedding. We use the center image frame to represent the 1-second long video, as the baseline did. In our image system, we perform a two-stage fine-tuning on some pre-trained models. The models pre-trained on ImageNet [3] are first fine-tuned on Places365 [4], and then fine-tuned on this task's dataset. These models are used to extract image embeddings. Then in model-level fusion, we concatenated audio and image embeddings as input into a classifier. We also use decision-level fusion for model ensemble. Our final fusion system can achieve a 0.143 log loss and 95.2% accuracy using official train/test split on the development dataset.

*Pengyuan Zhang is the corresponding author.

The remainder of this report is organized as follows. Section 2 describes our classification systems. Section 3 shows our experimental setup. Section 4 covers the results of our systems and makes some discussion. Section 5 concludes this report.

2. CLASSIFICATION SYSTEMS

2.1. Audio Model

Our audio classifier is based on [5]. It's a 1D deep convolutional neural network. We make some improvements on this model. We introduce residual learning [6] and change the model to be 1D-ResNet. In each block, we use one 3×3 convolutional layer to extract features and an extra 1×1 convolutional layer to match input and output's dimensions. Table 1 shows our network architecture. We use FC3's output as audio embedding.

We chooses three different acoustic features as input of the classifier for getting robust results. We use Mel filter bank feature, scalogram extracted by wavelets and bark filter bank feature which are all transformed from raw waveform.

Table 1: The 1D-ResNet Classifier. The size of input is frames(B) × channels(c) × filters(n). The notation "c-3 Conv(pad=1,stride=1)-2c-BN-ReLU" denotes a convolutional kernel with c input channels, 2c output channels and a size of 3, followed by batch normalization and ReLU activation.

Layer Name	Settings
Input	Acoustic feature $B \times c \times n$
Block1	c-3 Conv(pad=0,stride=1)-2c-BN-ReLU 2 Pooling(pad=1,stride=2)
Block2	2c-3 Conv(pad=0,stride=1)-4c-BN-ReLU 2 Pooling(pad=1,stride=2)
Block3	4c-3 Conv(pad=0,stride=1)-8c-BN-ReLU 2 Pooling(pad=1,stride=2)
Block4	8c-3 Conv(pad=0,stride=1)-16c-BN-ReLU 2 Pooling(pad=1,stride=2)
Flatten and concatenate input as well as Block's output	
FC1	Linear (2048 units)-BN-ReLU-Dropout
FC2	Linear (1024 units)-BN-ReLU-Dropout
FC3	Linear (1024 units)-BN-ReLU
Output	Linear (10 units)-Softmax

2.2. Image Model

In this report, we use four different image models (ResNet[6] , EfficientNet[7], EfficientNetV2[8], Swin Transformer[9]) to extract image embedding after pre-training and fine-tuning.

2.2.1. ResNet

ResNet[6] (Residual Network) is one of the most widely used CNN feature extraction networks. Deep residual learning is introduced in this network, which enables the network to maintain strong classification performance with increasing depth.

In this report, we choose ResNet-50. The network inputs are 224×224 images. The first layer is 7×7 convolution layer. Then we use a maxpool layer. After that, we have convolution layers consisting of four convolution blocks. The network ends with a global average pooling, a 10-way fully-connected layer, and softmax.

2.2.2. EfficientNet

Improving the depth and width of the network and the resolution of the input image can improve the accuracy of the ConvNets. EfiicientNet[7] uses a simple and efficient compound coefficient to enlarge the network from depth, width and resolution. The optimal set of parameters (compound coefficients) can be obtained based on neural structure search technology. And a new mobile-size baseline was developed to evaluate the scaling approach in EfiicientNet.

In this report, we use EfficientNet-b5 model. In this model, the resolution of the input image is 456×456. Its main building block is mobile inverted bottleneck MBConv[10]. The network ends with a 1×1 convolution layer, a pooling layer, and a 10-way fully-connected layer.

2.2.3. EfficientNetV2

EfficientNetV2[8] is an upgrade to EfficientNet that aims to improve training speed while maintaining efficient use of parameters. EfficientNetV2 introduces Fused-MBConv[11] to the search space based on EfficientNet; At the same time, the adaptive regularization intensity adjustment mechanism is introduced for the progressive learning.

In this report, We use EfficientNetV2-small model. EfficientNetV2 and EfficientNet are similar in structure, the difference is that the first few layers of EfficientNetV2 are replaced by the Fused-MBconv layer.

2.2.4. Swin Transformer

Swin Transformer[9] is a new vision Transformer, the representation is computed with shifted windows. The shifted windowing scheme brings greater efficiency by limiting self-attention computation to non-overlapping local windows while also allowing for cross-window connection. This hierarchical architecture has the flexibility to model at various scales and has linear computational complexity with respect to image size.

In this report, We use Swin Transformer tiny model. The network inputs are 224×224 images, we use a patch size of 4×4 and thus the feature dimension of each patch is 48. A linear embedding layer is applied on this raw-valued feature to project it to an arbitrary dimension. Four Transformer blocks with modified self-attention computation (Swin Transformer blocks) are applied on these patch tokens. A Swin Transformer block consists of a shifted window based MSA module, followed by a 2-layer MLP with GELU nonlinearity in between. A LayerNorm (LN) layer is applied before each MSA module and each MLP, and a residual connection is applied after each module. The window size is set to 7 and the embedding dimension is 96.

2.3. Audio-Visual Model

The target of multi-modal fusion is to integrate information from different modalities. There are mainly three strategies for multi-modal fusion, namely feature-level fusion, decision-level fusion and model-level fusion. Feature-level fusion simply concatenates multi-modal features into a joint feature vector at the input level. However, high-dimensional feature set may easily suffer from the problem of data sparseness[12]. In this report, we adopt a hybrid fusion strategy comprised of model-level and decision-level fusion.

2.3.1. Model-level Fusion

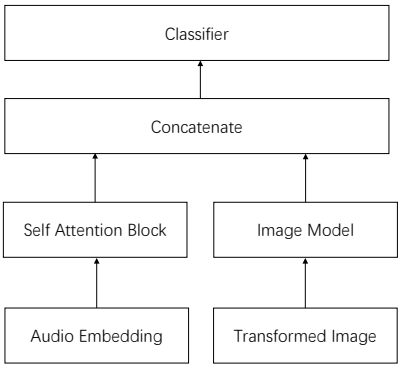

Figure 1: Architecture of the model-level fusion.

In the model-level fusion, we retrieve the inputs before the FC layer(also named embeddings) from audio and visual models. As shown in Figure 1, we firstly use self-attention mechanism[13] along time axis to transform audio embedding into a vector, and concatenate it with the visual embedding. The classifier contains two FC layers, followed by batch normalization and ReLU activation.

2.3.2. Decision-level Fusion

In the decision-level fusion, we retrieve the output probabilistic distribution of different model-level fusion models. Inspired by [5], we perform unweighted fusion and weighted fusion on the outputs. Different models are considered equally important in unweighted fusion, while in weighted fusion, we perform grid search to select the best weights of different models.

3. EXPERIMENTAL SETUP

3.1. Audio Experiments

3.1.1. Feature Extraction

The audio files in the development dataset are recorded in binaural way using 48kHz sampling rate and have a fixed-length of 10

seconds. For all features, we use average and difference channel instead of left and right channel. To extract features, STFT was all applied on the raw signal every 171*ms* over 512*ms* windows. For log-mel energies and bark filter bank feature, we all set the number of filters to 256 so that their dimension was 59×2×256. For scalogram, the total number of wavelet filters was set to 290 so its dimension was 59×2×290. For 1-second audio, the demension of features would be 6×2×256 or 6×2×290.

3.1.2. Model Training

We provided the same label as original audio to every frame of features and trained the model at the frame level. When we predicted a 1-second audio's scene, we calculated the average of the frame-wise output from our classifier.

We used the official train/test split of the development dataset to train and test our models. We used stochastic gradient descent (SGD) with a cosine-decay-restart learning rate scheduler. The maximun and mininum learning rates are 0.1 and 1e-5, respectively. Before submitting our final system, we used all development data to retrain our models.

Table 2: Results of experiments of our audio systems

Feature	Model	Log Loss	Accuracy
log-mel	DCNN	0.703	75.0%
	1D-ResNet	**0.652**	**77.5%**
scalogram	DCNN	0.661	76.6%
	1D-ResNet	**0.623**	**78.5%**
bark	DCNN	**0.764**	**71.5%**
	1D-ResNet	0.794	71.5%

3.2. Image Experiments

We evaluate our models on the DCASE2021 development dataset that consists of 10 classes, and we use ImageNet[3] for pre-training, use Places365[4] for fine-tuning. This experiment does not build a novel image scene classification model. We choose to use the model that is most effective in the field of image classification and has the appropriate number of model parameters. More complex models may lead to overfitting.

For ResNet-50 and EfficientNet-b5 model, we directly use the pre-trained weights which are pre-trained on ImageNet, then fine-tune the weights on Places365, and fine-tune the weights on DCASE2021 dataset. For the other two models, we firstly train them on ImageNet to get the pre-trained weights by ourselves, then fine-tune the weights on Places365, and fine-tune the weights on DCASE2021 dataset. Finally, we use the models and weights to extract the image embedding.

During the fine-tuning, we normalize all the images, and use RandomResizedCrop, RandomHorizontalFlip, random adjustment of image brightness and contrast and other data augmentation tricks. In addition, we use Mixup[14] data augmentation for the images. For all models, we use SGD as optimizer, and use cosine annealing and warm up to adjust learning rate. This strategy of adjusting the learning rate can keep the deep stability of the model. We use a weight decay of 1e-4 and a momentum of 0.9. We do not use dropout.

Table 3: Results of experiments of our image systems

Model	Log loss	Accuracy
ResNet-50	0.346	90.7%
EffNet-b5	0.274	90.9%
EffNetV2-S	**0.270**	**91.8%**
SwinT	0.371	88.8%

3.3. Audio-Visual Experiments

In the model-level fusion, the audio embeddings and the weights of trained image models are fixed, while the classifier is trainable. Since the ten segments in the same 10-second video is similar, we randomly choose one of them in every training epoch. We adopt Adam as optimizer, with a fixed learning rate of 1e-5. Note that we also use data augmentation for images in this stage to prevent overfitting. Combining 4 image models and 3 acoustic features in pair, we can derive 12 fusion models. In the decision-level fusion, we firstly get the outputs of the 12 fusion models, then use grid search of an interval of 0.05 in the weighted fusion.

3.4. External Data

This section is the list of external data sources used in our training. It contains ImageNet, Places365, EfficientNet and Pytorch pretrained Models on ImageNet, which are all included in the official external data resources.

4. RESULTS AND DISCUSSION

In this section, we report and discuss the evaluation results collected on the development dataset. Table 2 shows our audio systems' results. Our best audio-only system is 1D-ResNet using scalogram as input, achieving a 0.623 log loss and 78.5% accuracy. We also discover that our 1D-ResNet performs better than DCNN on the log-mel and scalogram features, but its log loss on bark is higher than DCNN. So when we extracted audio embedding, we used 1D-ResNet on log-mel and scalogram and DCNN on bark.

Table 4: Results of experiments of our audio-visual systems

ID	V-Model	A-Feature	Log Loss	Accuracy
1	ResNet-50	log-mel	0.189	93.7%
2		scalogram	0.187	94.1%
3		bark	0.211	93.0%
4	EffNet-b5	log-mel	**0.159**	**94.1%**
5		scalogram	0.170	93.5%
6		bark	0.197	93.3%
7	EffNetV2-S	log-mel	0.169	94.4%
8		scalogram	0.177	94.2%
9		bark	0.204	92.9%
10	SwinT	log-mel	0.210	91.8%
11		scalogram	0.217	91.7%
12		bark	0.244	91.1%

The results of our image systems are all listed in Table 3. EffNetV2-S has the best results which obtains a 0.270 log loss and 91.8% accuracy. The results of EffNet-b5 are close to EffNetV2-S.

Both ResNet-50 and SwinT achieve a log loss greater than 0.3 but they are also smaller than audio systems' results.

Table 4 shows the results of our audio-visual systems. The best performance is achieved from the combination of EfficientNet-b5 and log-mel, with a 0.159 log loss and 94.1% accuracy. When we fix the image model, we discover that log-mel and scalogram are much better than bark. When we fix the acoustic feature, we discover that although EfficientNetV2-S is the best model in image systems, EfficientNet-b5 is the best image model in audio-visual systems.

In Table 5, we present the results of our final fusion systems. In the grid search procedure, we discover that the weights of audio-visual models comprised of ResNet-50 are all zeros, so we abandon model 1,2,3. The weights distribution of No.1 fusion systems is [0.5, 0, 0.25, 0.15, 0.1, 0], and the weights distribution of No.3 fusion systems is [0.4, 0, 0.15, 0.2, 0.15, 0, 0.1, 0, 0]. Our best fusion system obtains a 0.143 log loss and 95.2% accuracy.

Table 5: Results of experiments of our fusion systems

No.	Model ID	Vote Method	Log loss	Accuracy
I	4,5,7,8,10,11	Weighted	0.144	95.0%
II	4,5,7,8,10,11	Unweighted	0.150	95.1%
III	4,5,6,7,8, 9,10,11,12	Weighted	**0.143**	**95.2%**
IV	4,5,6,7,8, 9,10,11,12	Unweighted	0.151	95.1%

5. CONCLUSION

This report describes our submission for DCASE2021 Task1b. Our audio system uses 1D-ResNet and three different acoustic features. A two-stage fine-tuning strategy is applied in our image system. We concatenate the extracted audio and image embedding and feed them into a classifier. Decision-level fusion is also used to make our system more robust. From our evaluation of development data, our final fusion system could achieve a 0.143 log loss and 95.2% accuracy.

6. REFERENCES

[1] S. Wang, T. Heittola, A. Mesaros, and T. Virtanen, "Audio-visual scene classification: analysis of dcase 2021 challenge submissions," 2021.

[2] S. Wang, A. Mesaros, T. Heittola, and T. Virtanen, "A curated dataset of urban scenes for audio-visual scene analysis," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, accepted. [Online]. Available: https://arxiv.org/abs/2011.00030

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[4] B. Zhou, À. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 1452–1464, 2018.

[5] H. Chen, Z. Liu, Z. Liu, P. Zhang, and Y. Yan, "Integrating the data augmentation scheme with various classifiers for acoustic scene modeling," *arXiv preprint arXiv:1907.06639*, 2019.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[7] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.

[8] M. Tan and Q. V. Le, "Efficientnetv2: Smaller models and faster training," *arXiv preprint arXiv:2104.00298*, 2021.

[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *ArXiv*, vol. abs/2103.14030, 2021.

[10] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[11] S. Gupta and B. Akin, "Accelerator-aware neural network design using automl," *ArXiv*, vol. abs/2003.02838, 2020.

[12] C.-H. Wu, J.-C. Lin, and W.-L. Wei, "Survey on audiovisual emotion recognition: Databases, features, and data fusion strategies," *APSIPA Trans. Signal Inf. Process.*, vol. 3, 2014.

[13] L. Tarantino, P. Garner, and A. Lazaridis, "Self-attention for speech emotion recognition," in *Proc. Interspeech*, 2019, pp. 2578–2582.

[14] H. Zhang, M. Cissé, Y. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *ArXiv*, vol. abs/1710.09412, 2018.