AUDIO FEATURE GENERATION FOR MISSING MODALITY PROBLEM IN VIDEO ACTION RECOGNITION

Hu-Cheng Lee, Chih-Yu Lin, Pin-Chun Hsu, Winston H. Hsu

National Tawian University, Taiwan

ABSTRACT

Despite the recent success of multi-modal action recognition in videos, in reality, we usually confront the situation that some data are not available beforehand, especially for multi-modal data. For example, while vision and audio data are required to address the multi-modal action recognition, audio tracks in videos are easily lost due to the broken files or the limitation of devices. To cope with this sound-missing problem, we present an approach to simulating deep audio feature from merely spatial-temporal vision data. We demonstrate that adding the simulating sound feature can significantly assist the multi-modal action recognition task. Evaluating our method on the *Moments in Time (MIT) Dataset* , we show that our proposed method performs favorably against the two-stream architecture, enabling a richer understanding of multi-modal action recognition in video.

Index Terms— missing modality problem, action recognition, audio feature simulation, neural network, deep learning

1. INTRODUCTION

Thousands of activities accompanied with different sounds are occurring around us in our daily life. Recently, multi-modal action recognition in videos has been intensively investigated due to its wide applications [1, 2, 3, 4, 5]. In some situations, certain actions share similar visual appearances and are difficult to be discriminated. However, with the help of multi-modal data, we can understand the action in videos more comprehensively. As shown in Fig.1, for example, (a) shows that marchers parade on the avenue while (b) shows that audiences watch the baseball game, and they share the similar scene - crowded people. With the help of sound, we can easily tell the difference between parade and baseball game. Same situation for (c) and (d), the sounds of howling and barking are completely different though they have similar scene. Although we can benefit the action recognition from multi-modal data, in reality, we usually confront the situation that some modalities of data are not accessible beforehand.

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2634-F-002-004. We also benefit from the NVIDIA grants and the DGX-1 AI Supercomputer.

(a) parade (b) baseball game (c) barking dog (d) howling dog

Fig. 1. **Confusing scene.** In multi-modal action recognition task, some actions share similar visual appearances. Obviously, we can tell the difference between the similar appearances with the help of sound. However, in reality, we usually confront the *Missing Modality Problem* - especially sound is not available. (eg. roughly 44% videos in MIT dataset are soundless.)

For example, audio tracks in videos are frequently lost due to the broken files or the limitation of devices. We define such situation as *Missing Modality Problem*. Besides, according to our statistics, roughly 44% videos in *Moments in Time (MIT) dataset*[6] are soundless.

Solutions to multi-modal recognition often utilize the knowledge from various modality and mutually aid each other. Aytar *et al.* [7] proposed a network producing a deep aligned representation among text, sound and image. Simonyan *et al.* [4] proposed a two-stream ConvNet architecture incorporating spatial and temporal networks, which is a well-known baseline model for action recognition. Yet, these works only explores the relationship between sounds and images, but not the relationship between sounds and videos. Besides, for video task, most approaches only leverage the spatial and temporal data but not auditory data. As for missing modality problem, Ding *et al.* [8] proposed a dictionary learning to guide the knowledge transfer between and within different face databases. We want to explore the

new opportunity for action recognition.

In this paper, we propose a audio feature generation framework for missing modality problem. Our goal is not only to exploit the relationships between vision and sound, but also to improve our action recognition results by simulating the sound feature for the missing ones. We formulate our problem into cross-modality generation problem. *Our model accepts fused deep features extracted from RGB images as well as optical flows and produces simulated deep audio features.* In order to learn semantic deep audio feature, we also utilize the ground truth of action label and adopt multi-tasking method in training stage. Our experiments show that these simulated deep audio features boost the performance of multi-modal action recognition especially on soundless videos in certain classes.

Specifically, we make the following contributions:

- We design a audio feature generation network to learn the relationship between visual data and audio data from videos.
- To our knowledge, leveraging the simulated audio deep feature generated from spatial-temporal vision data to improve action recognition is novel.
- Our proposed method performs favorably against two-stream method on *MIT Dataset*.

2. PROPOSED METHOD

Our proposed model is intuitive. Since we possess spatial and temporal modality rather than auditory modality, we formulate our problem into cross-modality generation problem. In the following section, we will introduce our two main architectures, and then describe our training and testing process.

2.1. LSTM-based sound feature generator

Due to the continuity of videos, we want to exploit the context information among a sequence of frames. Therefore, we choose LSTM [9] as our generation structure. As shown in Fig. 2 (a), first we utilize the feature extractor to extract the spatial and temporal features:

$$v_R = cnn_{\phi_r}(I_R), v_F = cnn_{\phi_f}(I_F), \tag{1}$$

where I_R, I_F are a set of RGB images and optical flow images respectively: $I_R = \{I_{r,1}, I_{r,2}, ..., I_{r,T}\}$, $I_F = \{I_{f,1}, I_{f,2}, ..., I_{f,T}\}$. $v_{r,t} \in \mathbb{R}^m$ and $v_{f,t} \in \mathbb{R}^n$ are the RGB and optical flow feature extracted by feature extractor at time t, and t ranges from { 1,2,...,T } for a given video length T. cnn_{ϕ_r} and cnn_{ϕ_r} are feature extractor with parameter ϕ. Then we concatenate the spatial and temporal representations at time t respectively, aggregating features as the input of LSTM:

$$x = LSTM(v_{c,1}, v_{c,2}, ..., v_{c,T}) \tag{2}$$

where $v_{c,t} = v_{r,t} \oplus v_{f,t}$, and $\oplus$ denotes concatenation between spatial and temporal features. x is the last hidden state output of the LSTM, which represents the spatial and temporal information of the entire video sequence.

Finally, we have a two sub-branches and perform the multi-task learning. The first branch is classification branch:

$$p(y) = softmax(W_p(x) + b_p) \tag{3}$$

where $W_p \in \mathbb{R}^{d_p \times d_x}$ and $b_p \in \mathbb{R}^{d_p}$ are learned weights and biases for action recognition. $p(y)$ is prediction results in action recognition.

The second branch is sound feature reconstruction branch. First, we utilize the sound feature extractor to extract the sound feature from the video, and we harness it as the ground truth of sound feature.

$$r(y) = W_r(x) + b_r \tag{4}$$

where $W_r \in \mathbb{R}^{d_r \times d_x}$ and $b_r \in \mathbb{R}^{d_r}$ are learned weights and biases for sound reconstruction. $r(y)$ is the reconstruction of the sound feature. Formally, during training, we define a multi-task loss as $L = L_{cls} + L_{recon}$. The classification loss L_{cls} is identical to those defined in action recognition. We train the network with cross-entropy loss, where the probabilities are obtained through a softmax function. For the reconstruction loss L_{recon}, we have two options: L2 loss and KL-divergence loss.

2.2. Autoencoder-based sound feature generator

Autoencoder [10] is another noted generative model. We replace the main structure with autoencoder. As shown in Fig. 2 (b), the entire process is almost identical to the LSTM one. Autoencoder consists of two parts: encoder and decoder.

$$l = encoder(v_{c,1}, v_{c,2}, ..., v_{c,T}), x = decoder(l) \tag{5}$$

The encoder takes a random frame of video sequence as input and learns the corresponding latent representation. Then we exploit the latent representation to predict the action. The decoder receives the latent representation to reconstruct the sound feature.

$$p(y) = softmax(W_p(l) + b_p), r(y) = W_r(x) + b_r \tag{6}$$

2.3. Training and testing process

We split the dataset into two parts: videos with sound track and videos without sound track. At training stage, considering the sound reconstruction needs the ground truth of sound feature, we utilize the videos with sound tracks to train the model. At testing stage, the sound reconstruction branch can predict the sound feature via the spatial and temporal information without ground truth of sound feature. Therefore, the model simulates the sound features for those videos without sound tracks and assists action recognition task.

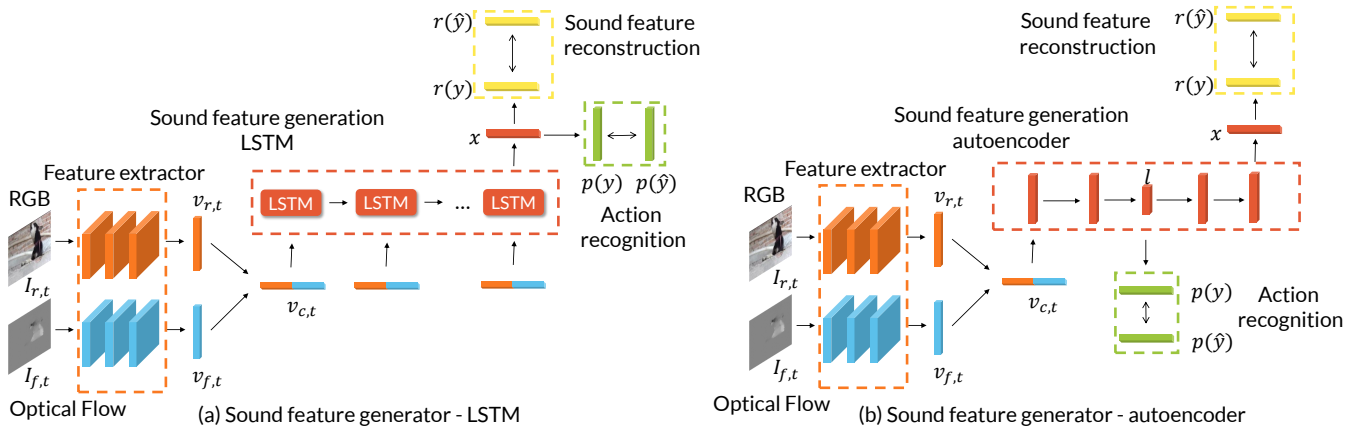

Fig. 2. **Overview of our proposed architecture:** Since we possess spatial and temporal modality rather than auditory modality, we intuitively formulate our problem into cross-modality generation problem. We have two main architecture: (a) LSTM-based sound generator and (b) autoencoder-based sound generator. More details can be found in section 2.1 and 2.2.

3. EXPERIMENTS AND RESULTS

Table 1. The number of videos with sound and without sound.

the number of videos	Training	Validation
with sound	55,196	6,286
without sound	44,804	3,714
total	100,000	10,000

3.1. Datasets: Moments in Time Dataset (MIT Dataset)

MIT Dataset is a large-scale human-annotated collection of short videos corresponding to dynamic events and has a significant intra-class variation among the categories. We choose the mini-track of the dataset. The dataset consists of over a hundred thousand 3-second videos corresponding to 200 different verbs. Each verb is associated with 650 videos resulting in a large balanced dataset for learning a basis of dynamical events from videos. According to our statistics, as shown in Table. 1, nearly 44% videos have no auditory signals.

3.2. Implementation details

We will discuss how to extract the image and sound features, and how to fuse our prediction of each modality.

Action recognition: We fine-tune a pre-trained Temporal Segment Network (TSN)[5] on *MIT Dataset* sampled at 6 FPS. The detailed settings are identical to TSN. After fine-tuning the network, we utilize TSN as feature extractor to extract image and optical flow feature.

Sound feature extraction: We use two pretrained models for audio feature extraction: *Audio Event Net* (AENet) [11] and *VGGish* pretrained on AudioSet[12]. We use wav file format with 16kHz sampling rate, 16bit, mono channel; the codec is PCM S16 LE. In AENet, the dimensions of extracted features are $(N, 1024)$, where N equals to the total length in seconds. For VGGish, we extract the features into $(N, 3, 128)$ embeddings.

Fusion method: For sound feature prediction, we train 200 linear SVM binary classifiers for each class using the extracted AENet and VGGish features respectively. Besides, we do not perform any preprocessing on the extracted AENet features while we flatten the extracted VGGish features to dimension $(N, 384)$ before we feed them into the SVM classifiers for training and validation. For the final prediction, we directly add the probabilities of TSN and sound feature.

3.3. Results

In the following section we conduct an ablation study of our audio feature generation model. First, we want to discuss the effectiveness of the sound feature for action recognition. We compare the different combination of sound generation strategies in Table. 2 and Table. 3. Afterwards, we aim to explore the best combination of our audio feature generation framework. Note that for the baseline row in the table, since we use the audio feature to acquire our action prediction, we assume that the prediction of the missing-sound videos will be definitely wrong. Besides, we want to discuss how auditory modality will affect the performance of multi-modal action recognition. As shown in Table. 4, it shows that our simulated audio feature significantly improve the multi-modal action recognition. Last but not least, we want to observe that the accuracy of which action class will be enhanced the most by the effectiveness of sound. We also list the top-3 improved classes results in Table. 5.

Architecture comparison: We observe that for AENet sound feature in Table. 2, LSTM-based sound feature gen-

Table 2. **Simulated AENet sound feature for action recognition accuracy on the *MIT Dataset*.**

AENet		Top-1 (%)	Top-5 (%)
w/o generation (baseline)		4.41	11.78
LSTM			
L2 Loss	w/ classifier	4.53	11.69
	w/o classifier	**5.19**	**13.44**
KL Div.	w/ classifier	4.47	11.50
	w/o classifier	4.45	11.40
Autoencoder			
L2 Loss	w/ classifier	**4.70**	**11.70**
	w/o classifier	**4.70**	**11.70**
KL Div.	w/ classifier	4.52	11.48
	w/o classifier	4.55	11.60

Table 3. **Simulated VGGish sound feature for action recognition accuracy on the *MIT Dataset*.**

VGGish		Top-1 (%)	Top-5 (%)
w/o generation (baseline)		1.57	7.29
LSTM			
L2 Loss	w/ classifier	1.54	6.91
	w/o classifier	**1.95**	**7.59**
KL Div.	w/ classifier	1.59	6.85
	w/o classifier	1.59	6.83
Autoencoder			
L2 Loss	w/ classifier	**2.19**	**7.86**
	w/o classifier	2.11	7.84
KL Div.	w/ classifier	1.71	7.23
	w/o classifier	1.72	6.90

erator performs better than autoencoder-based one. As for VGGish sound feature in Table 3, autoencoder-based sound feature generator is better. The experiment shows that the choices of sound feature generator will depend on the sound feature.

Loss comparison: In Table. 2, Table. 3, L2 loss defeats KL divergence in both sound feature and both architectures. The result shows that L2 loss is a better choice for sound feature reconstruction.

w/ or w/o classifier comparison: The reason for adding the classifier branch to the generation structure is that we want to see if the class label information helps the sound feature reconstruction or not. Experiment shows that the classifier branch is helpful for autoencoder-based structure while it is not helpful for LSTM-based structure. Although the choices of classifier are case by case, they all surpass the results without generation, which shows that after adding simulating sound feature, it will definitely improve the accuracy of action recognition.

Overall comparison: Finally, after choosing our best audio-simulated framework, we demonstrate the whole action recognition results with our simulated features. As shown in Table. 4, our proposed method outperforms other methods. We can see significant improvements over each modality input modality as well as over the late-fused results.

Table 4. **Overall comparison of action recognition**. With the assistance of sound, we demonstrate that simulated sound feature significantly improves the action recognition.

Method	Top-1 (%)	Top-5 (%)
RGB (spatial)	24.23	50.86
Flow (temporal)	12.28	29.90
RGB + Flow (fusion)	27.02	52.50
RGB + AENet	26.69	52.62
RGB + VGGish	23.54	49.56
RGB + AENet (Gen-LSTM)	**26.91**	**52.93**
RGB + AENet (Gen-AE)	26.78	52.58
RGB + VGGish (Gen-LSTM)	25.96	**51.77**
RGB + VGGish (Gen-AE)	**26.02**	51.75
fusion + AENet (Gen-LSTM)	**28.19**	**53.73**
fusion + VGGish (Gen-AE)	27.58	53.29

Table 5. **Improvement ranking for action recognition**. We list the top-3 improved action class. *R* means RGB feature. *S* means original sound feature. *S(gen)* means simulated sound feature.

(R+S) vs. R	*(R+S(gen)) vs. R*	*(R+S) vs. (R+S(gen))*
whistling	whistling	boiling
howling	howling	wrapping
chewing	mowing	swimming

Improvement of action class comparison: We compare the performance of each action class after the addition of sound and list the top-3 improved classes. As shown in Table 5, we observe that the most-improved classes of original sound feature (*(R+S) vs.R*) are nearly identical to those of simulated sound feature (*(R+S(gen)) vs.R*). However, Table. 2, 3 demonstrate that the accuracy of simulated features perform better than that of original features. Hence, we compare the scenario with and without generation and list the most-improved classes which are also sound-related classes such as boiling and swimming. The reason for the different improved classes is that for those sound-related actions such as whistling, nearly every video has sound track in our dataset, which means after our generation, they will not be improved so much. However, we can still improve the performance by those soundless yet sound-related videos such as boiling and swimming.

4. CONCLUSION

To cope with the missing modality problem, we propose an approach to simulating deep audio feature from merely spatial-temporal data. We demonstrate that simulating sound feature significantly assists our action recognition. Evaluating our method on the *Moments in Time Dataset*, we show that our proposed method performs favorably against the two-stream architecture.

5. REFERENCES

[1] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," 2015.

[2] Christoph Feichtenhofer, Axel Pinz, and Richard Wildes, "Spatiotemporal residual networks for video action recognition," 2016.

[3] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[4] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.

[5] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Val Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *ECCV*, 2016.

[6] Mathew Monfort, Bolei Zhou, Sarah Adel Bargal, Alex Andonian, Tom Yan, Kandan Ramakrishnan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al., "Moments in time dataset: one million videos for event understanding," *arXiv preprint arXiv:1801.03150*, 2018.

[7] Yusuf Aytar, Carl Vondrick, and Antonio Torralba, "See, hear, and read: Deep aligned representations," *arXiv preprint arXiv:1706.00932*, 2017.

[8] Zhengming Ding, Shao Ming, and Yun Fu, "Latent low-rank transfer subspace learning for missing modality recognition," 2014.

[9] Martin Sundermeyer, Ralf Schlüter, and Hermann Ney, "Lstm neural networks for language modeling," in *Thirteenth annual conference of the international speech communication association*, 2012.

[10] Pierre Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 37–49.

[11] Naoya Takahashi, Michael Gygli, and Luc Van Gool, "Aenet: Learning deep audio features for video analysis," *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 513–524, 2018.

[12] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson, "Cnn architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017.